

Capacitive Touch Panel with 8 Buttons and Paged Graphical Display

ZVI-TMDV

Application Programme Version: [1.1]
User Manual Version: [1.1]_a

CONTENTS

Contents	2
Document updates.....	4
1 Introduction	5
1.1 TMD-Display View	5
1.2 Installation.....	7
2 Configuration	9
2.1 Default Configuration.....	9
2.2 General	11
2.2.1 CONFIGURATION	11
2.2.2 Temperature Sensor	13
2.2.3 Screensaver.....	13
2.2.4 LEDs & Display	14
2.2.5 Sounds	18
2.2.6 Touch Locking	20
2.2.7 Welcome Back Object.....	21
2.3 Pages	24
2.3.1 CONFIGURATION	24
2.4 Page <i>n</i>	26
2.4.1 CONFIGURATION	26
2.4.2 Disabled	29
2.4.3 Pair	29
2.4.4 Individual	53
2.5 Configuration pages	67

2.5.1	General Configuration	67
2.5.2	Display.....	68
2.6	Inputs.....	69
2.6.1	Binary Input	69
2.6.2	Temperature Probe	69
2.6.3	Motion Detector	70
2.7	Thermostat	71
ANNEX I: LED Illumination Modes.....		72
ANNEX II. Button Icons		75
ANNEX II: Communication Objects		76

DOCUMENT UPDATES

Version	Changes	Page(s)
[1.1]_a	Changes in the application programme: <ul style="list-style-type: none">• Optimisation of the start-up process.• Minor optimisation in character representations.• Update of the motion detector function.	-

1 INTRODUCTION

1.1 TMD-DISPLAY VIEW

TMD-Display View is an evolution of TMD-Display One, the KNX **multifunction capacitive touch switch** with graphical display from Zennio.

TMD-Display View provides **eight capacitive buttons** aligned in 2 columns of 4 buttons each, **LEDs** to indicate user presses or states, and a **wide display** on the top of the front panel, as in ZAS and Roll-ZAS from Zennio.

TMD-Display View is a **fully customisable** solution for the room control, including hotel rooms, offices or any other environment where the user needs to control climate systems, lighting, blinds, scenes, etc.

The versatility of the above functions is enhanced by the built-in **analogue/digital inputs**, the **internal temperature sensor** and the **thermostat** function, as well as an elegant and **fully customisable design of the front glass** – customers can choose their button icons, texts and colours and even personalise the background with their pictures, logos, etc.

Moreover, the **display** offers the possibility to show useful information to the user about the control states and object values, being possible to configure **up to 4 different pages**. Unlike those in TMD-Display One, the capacitive buttons and the display in TMD-Display View can perform different functions depending on the page selected, thus providing the user a versatile interface with multitude of controls available.

The most outstanding features of TMD-Display View are:

- **1.8"** back-lit display with a resolution of **128x64** pixels.
- **Fully customisable** design of the front glass.
- **8 touch buttons** which can operate as individual or pair controls.
- Up to **4 different pages** available.
- **Light indicator (LED)** for every button, the state of which (on or off) can depend on different factors according to the user's needs.

- **Buzzer** for an audible acknowledgement of user actions (with the possibility of disabling it either by parameter or by object).
- Possibility of **locking / unlocking the touch panel** through binary orders or scenes, and of setting a timed/automatic locking of the device (**cleaning function**).
- Welcome greeting on the display and Welcome Back object (binary or scene).
- **Central touch area** which allows browsing through the configured pages or confirming certain actions.
- Screensaver function.
- **Two analogue/digital inputs** (temperature probes, additional switches, etc.).
- **Thermostat** function.
- Built-in temperature sensor.
- **Motion detection function**, by connecting the corresponding accessory.



Figure 1 TMD-Display View.

1.2 INSTALLATION

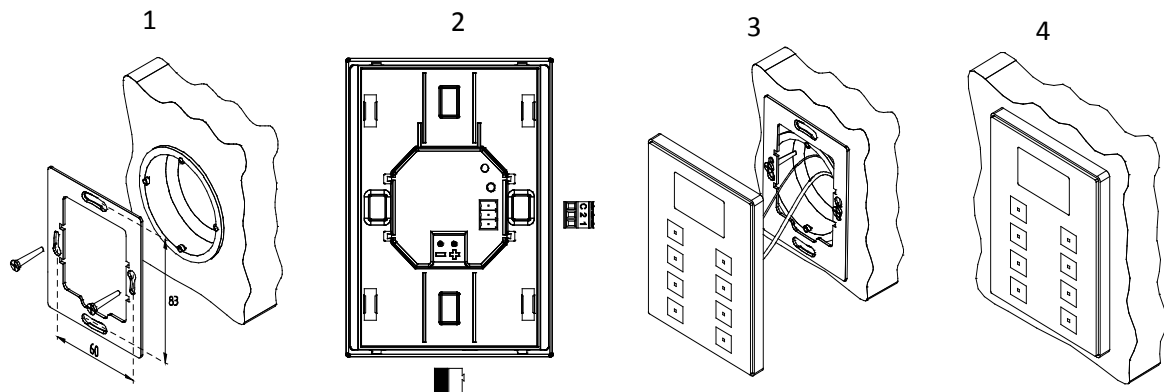


Figure 2 TMD-Display View. Installation Process.

To install the device, it is first necessary to mount the metallic plate into a square/round standard appliance box through the suitable screws. Next, TMD-Display View is connected to the KNX bus through the corresponding terminal on the rear side of the device, and then the input terminal is as well connected to the rear of the device.

Once the input terminal and the KNX terminal are connected, the device can be easily mounted on the metallic plate by the action of the built-in magnets. After that, it is necessary to slide it downwards to fix it through the security anchorage system.

Finally, it is advisable to check that the device is properly installed, and that only its profile becomes visible from above, from below and from both sides (the metallic plate should be completely hidden).

This device does not need any external supply, as it is powered through the KNX bus.

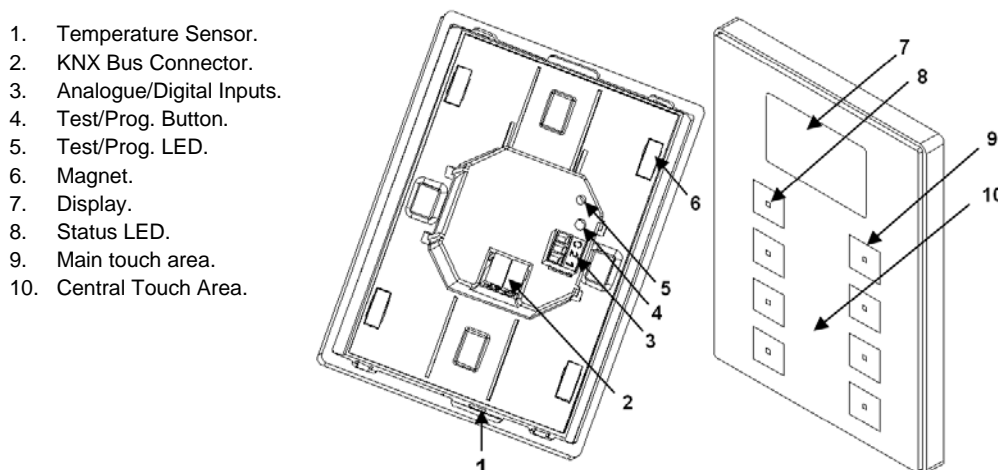


Figure 3 Element diagram.

The programming button (4) can be pressed with the help of a thin screw to set the device into the **Programming Mode**. After a short press, the programming LED (5) will light in red. Note that if this button is held while plugging the device into the KNX bus, the device will enter the **Safe Mode**. The LED will then blink in red.

Note: *whenever the device recovers the bus power, an immediate self-calibration process of the touch panel takes place. Please ensure to avoid touching the front glass while powering the device. If undesired effects arise during normal use, please disconnect the device from the bus and connect it again, making sure that the front glass is not touched during this process.*

For detailed information about the technical features of the device and for safety instructions and on the installation process, please refer to the **Datasheet** bundled with the original packaging of the device and also available at www.zennio.com.

2 CONFIGURATION

2.1 DEFAULT CONFIGURATION

To make the device perform the desired functions, several options need to be parameterised related to its **general behaviour** (screensaver, sounds, LEDs and display brightness levels, lock procedure of the touch panel, welcome back object, etc.).

Some of the above options are common to other Zennio devices such as Touch-MyDesign, Roll-ZAS or Z41, while others are specific for TMD-Display View.

On the other hand, TMD-Display View features two opto-coupled inputs, each of which may be independently configured as a **switch/sensor**, a **push-button** or a **temperature probe** or **motion detector**, so that external accessories can be connected to the device.

In the particular case of an external temperature probe (such as model **ZN1AC-NTC68** from Zennio), it will be possible to use it with independence of the built-in temperature sensor of the device, which implements its own communication objects and can be enabled and disabled in parameters.

ETS PARAMETERISATION

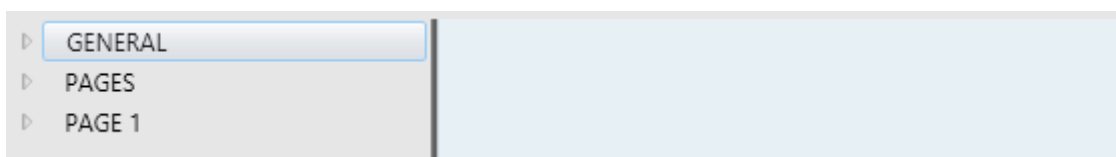


Figure 4 General.

After importing the corresponding database in ETS and adding the device into the topology of the project, the configuration process begins by right-clicking into the device and selecting *Edit parameters*.

This will bring the window shown in Figure 4, which contains the following tabs:

- **General:** contains the parameters that define the general behaviour of the device and that enable/disable additional features such as the thermostat or the external inputs. See section 2.2 for details.

- **Pages:** contains the parameters for enabling/disabling the different control pages and some general parameters related to the touch buttons and the navigation control. See section 2.3 for details.
- **Page 1:** contains the parameters for enabling/disabling the controls contained in page 1 (enabled by default). See section 2.4 for details.

Depending on the configuration, other tabs may show in the tab tree on the left.

These tabs will be explained in the next sections.

2.2 GENERAL

As shown in Figure 5, the “General” tab only contains one sub-tab by default: “Configuration”. However, it may include other tabs depending on the parameterisation.

2.2.1 CONFIGURATION

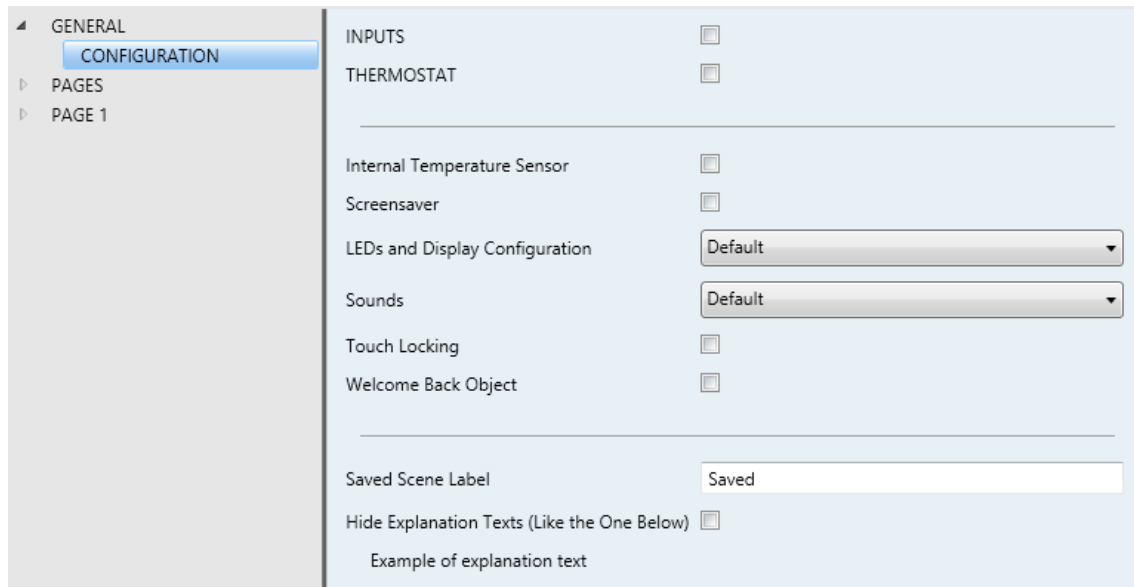


Figure 5. General - Configuration

This tab shows the following parameters:

- **Inputs:** enables or disables the “Inputs” tab in the tree on the left, depending on whether the device will or will not be connected any external accessories. See section 2.6 for details.
- **Thermostat:** enables or disables the “Thermostat” tab in the tree on the left. See section 2.7 for details.
- **Internal temperature sensor:** enables or disables the “Temperature Sensor” tab in the tree on the left. See section 2.2.2 for details.
- **Screensaver:** enables or disables the “Screensaver” tab in the tree on the left. See section 2.2.3 for details.
- **LEDs and Display config.:** sets whether the LEDs and display should make use of the pre-defined brightness levels (“Default”) or of a user-defined configuration (“Custom”).

- **“Default”**: the LEDs will remain off while in the “off” state, and at the maximum light level while in the “on” state. The display contrast level is set to 7.
- **“Custom”**: a specific tab will be included in the tab tree on the left so the integrator can set the desired light levels for the “off” and “on” states and whether to use the Night Mode or not. The display contrast level will also be configurable.

See section 2.2.4 for details.

- **Sounds**: sets whether the sound functions (button beeps, alarm and doorbell) should work according to the pre-defined configuration (“**Default**”) or to a user-defined configuration (“**Custom**”). See section 2.2.5 for details.
- **Touch Locking**: enables or disables the “Touch Locking” tab in the tree on the left. See section 2.2.6 for details.
- **Welcome Back Object**: enables or disables the “Welcome Back” tab in the tree on the left. See section 2.2.7 for details.
- **Saved Scene Label**: textbox to enter the string displayed when a scene is being saved.
- **Hide explanation texts**: displays or hides the explanation texts on the top of the parameter screen of some functions. Experienced users may prefer to hide these texts to have simpler parameter screens.

The project topology shows the following objects by default:

- **[General] Time of day**: object for updating the time shown on the display.

Important: *the time of day must be set through the bus by means of an external reference. Note that in the absence of power, the clock will get outdated.*
- **[General] Scene: Receive** and **[General] Scene: Send**: objects for respectively receiving and sending scene values from/to the KNX bus whenever it is necessary (e.g., when the user touches a button that has been configured to send scene commands; see section 2.4.3 and section 2.4.4).

2.2.2 TEMPERATURE SENSOR

TMD-Display View is equipped with **one internal temperature probe** which can monitor the ambient temperature of the room, thus making the device capable of reporting it to the KNX bus and of triggering certain actions when the temperature reaches specific values.

Please refer to the specific manual “**Temperature Sensor**” (available in the TMD-Display View section at the Zennio homepage, www.zennio.com) for detailed information about the functionality and the configuration of the related parameters.

2.2.3 SCREENSAVER

The screensaver is a special page that will only be shown after a period of inactivity, configurable by parameter.

It is possible to configure the screensaver to only show the current **Time**, the current **Temperature** (selecting the desired measurement source: the internal temperature sensor or an external value) or **both** (alternating every five seconds).

Notes:

- *The temperature shown will be truncated to the range [-99°C, 199°C].*
- *The time of day must be set through the bus by an external reference.
Note that in the absence of power the clock will get outdated.*

The screensaver disappears when any button is touched. It can be set by parameter whether, while the screensaver is active, this press should simply cause the screensaver interruption or, moreover –apart from hiding the screensaver–, the execution of the action corresponding to the pressed button.

ETS PARAMETERISATION

This screen is divided into three parts: Activation, Content and Exit, containing the following parameters:

- **Activation timeout:** time to automatically show the screensaver after the last button touch. Range: 1 – 255 seconds, minutes or hours.
- **Time of Day:** sets whether to show the current time or not.
- **Temperature:** sets whether to show the current temperature or not. When enabled, the following parameter will show up:
 - **Taken from:** defines the source of the temperature value: “Internal sensor” or “External reference”. The latter will bring a new two-byte object to the project topology (“**[General] Screensaver - External temperature**”) so that it the device can receive the required values from the bus.
- **Run button action on screensaver exit:** sets whether touching a button while the screensaver is active should cause the execution of the button action, apart from hiding the screensaver.

Figure 6 General – Screensaver.

2.2.4 LEDs & DISPLAY

As anticipated in section 2.2, the integrator can configure the **LEDs of the buttons** to make use of the pre-defined brightness levels or of user-defined levels.

To begin with, each LED can commute between two states: **off** (which not necessarily means “no light”) and **on** (which not necessarily means “light on”). They will switch their states according to the options explained in [ANNEX I: LED Illumination Modes](#).

Moreover, the LEDs and the display can also commute between two operation modes: the **normal mode** and the **night mode**. The second one is optional. It is provided for temporary situations and environments where an excess of brightness may disturb the user. In such cases, it will be possible to switch the mode by means of a one-bit object and/or a scene object.

Note: a general parameter is provided to make the ETS parameterisation easier in case the night mode is not required and the integrator feels comfortable with the default on/off brightness levels.

Regarding the display, it is possible to parameterise a contrast level in parameters. The integrator can also enable an object to change the display contrast in real time.

ETS PARAMETERISATION

The screenshot shows the 'LEDs and Display' configuration window. On the left is a tree view with 'GENERAL' expanded, containing 'CONFIGURATION' (with 'LEDs and Display' selected), 'PAGES', and 'PAGE 1'. The main area is divided into two sections: 'Normal Mode' and 'Night Mode'. 'Normal Mode' is active, indicated by a checked checkbox. It contains three parameters: 'LEDs: OFF Level Brightness' (value 0), 'LEDs: ON Level Brightness' (value 255), and 'Display: Brightness' (value 10). 'Night Mode' is inactive, indicated by an unchecked checkbox. Below it, 'Display Contrast Level' is set to 7, and 'Object for Changing the Contrast Level' is an unchecked checkbox.

Parameter	Value
Normal Mode	<input checked="" type="checkbox"/>
LEDs: OFF Level Brightness	0
LEDs: ON Level Brightness	255
Display: Brightness	10
Night Mode	<input type="checkbox"/>
Display Contrast Level	7
Object for Changing the Contrast Level	<input type="checkbox"/>

Figure 7 General – LEDs & Display config.

After selecting “Custom” for “**LEDs and Display config.**” (see section 2.2.1), a new tab will be incorporated into the tree on the left.

The parameters shown in Figure 7 are equivalent to those for the “Default” configuration of parameter “**LEDs and Display config.**”.

The upper section of this new screen is for the LED brightness configuration while the lower is for the display brightness configuration.

LED Brightness Levels

Normal Mode	<input checked="" type="checkbox"/>
LEDs: OFF level brightness	0
LEDs: ON level brightness	255
Display: Brightness	10
Night Mode	<input checked="" type="checkbox"/>
LEDs: OFF level brightness	1
LEDs: ON level brightness	8
Display: Brightness	10
Brightness Mode Control:	
Control object: 1-bit	<input checked="" type="checkbox"/>
Value	0 = Normal; 1 = Night
Control object: Scene	<input type="checkbox"/>
Brightness Mode after ETS download	Normal
Light up when any button is pressed	<input checked="" type="checkbox"/>
Length of lighting	10 x 1 s.

Figure 8 General – LEDs & Display – LED Brightness.

- Normal Mode:
 - **LEDs: OFF level brightness:** values between 0 (default) and 255.
 - **LEDs: ON level brightness:** values between 0 and 255 (default).
 - **Display: Brightness:** values between 0 and 10 (default).
- **Night Mode:** in case of being this mode necessary, this checkbox needs to be marked, which will incorporate some new parameters:
 - **LEDs: OFF level brightness:** values between 0 and 255 (1 by default).
 - **LEDs: ON level brightness:** values between 0 and 255 (8 by default).
 - **Display: Brightness:** values between 0 and 10 (default).


In case of enabling the night mode, some more options can be configured:

- **Control object: 1 bit:** when marked, it will be possible to switch the mode by writing to a binary object (“**[General] LEDs & Disp - Brightness Mode**”). A specific drop-down list will show up to select which value should trigger which mode (“0 = Normal; 1 = Night” or “0 = Night; 1 = Normal”).
- **Control object: Scene:** when marked, it will be possible to switch the mode by writing a certain scene value to “**[General] Scene: Receive**”. Two specific textboxes will show up to enter what scenes (1 through 64) will trigger each mode.
- **Idle time to change to Night Mode (0=disabled):** when the time set in this parameter elapses, TMD-Display View will switch to Night Mode (provided that it was not already in this mode). The range is 0 to 65535 seconds (0 by default).
- **Brightness Mode after ETS download:** sets which of the two modes (“Normal” or “Night”) will be active after an ETS Download.

Once either **Control object: 1-bit** or **Control object: scene** have been activated, a new parameter will be shown to allow a temporary transition to normal mode when a button is pressed during the night mode:

- **Light up when any button is pressed:** sets whether pressing on one of the enabled buttons during the night mode should imply a temporary transition to normal mode or not.
- **Length of lighting:** sets the time (1 to 65535 seconds) the device will remain in normal mode before switching back to night mode. The default value is 10 seconds.

Display



Display contrast level 7

Object for changing the contrast level ☐

Figure 9 General – LEDs & Display – Display configuration.

- **Display contrast level:** value to set the display contrast level, from 1 to 10 (7 by default).
- **Object for changing the contrast level:** enables a one-byte object (“[General] Display - Contrast”) to change the display contrast level in real time.

2.2.5 SOUNDS

Apart from the behaviour of the LEDs, it is possible to make TMD-Display View emit a **brief beep** as an acoustic feedback in two situations:

- **When a button is pressed:** short beep indicating that the user has pressed a button. This only applies to step controls, i.e., controls that walk through a certain range of values and that do not send a value after every touch, but only the final value after the last press.
- **When an action has been triggered:** sharper beep than the above one, indicating that a value is being sent to the KNX bus.

Enabling and disabling the button sounds can be done in parameters or through an object, being also possible to define in parameters whether the button sounds should be initially enabled or not.

On the other hand, TMD-Display View can also emit the following sounds on request (through the corresponding communication objects):

- **Doorbell sounds** (a single beep),
- **Alarm sounds** (a sequence of brief beeps with a higher pitch),

The alarm sequence will only stop when the alarm object gets deactivated or when the user touches any of the buttons (this will not trigger any action, only the alarm deactivation). Note that doorbell requests are ignored while the alarm is active.

A general parameter is provided to make the ETS parameterisation easier in case the doorbell and alarm functions are not required and the user feels comfortable with the default button beeps. On the contrary, customising the button, doorbell and alarm sounds involves a set of parameters, as explained next.

ETS PARAMETERISATION

In case the default button beep sound matches the requirements of the installation and the doorbell and alarm functions are not necessary, the “**Sounds**” parameter in the general **Configuration** tab (see section 2.2.1) can be set to “Default”. This will also imply that the button beeps will be unconditional, as it will not be possible to disable this function through an object.

On the other hand, if set to “Custom”, a specific tab named “**Sounds**” will show up in the tab tree on the left. The initial configuration of this screen is equivalent to the aforementioned default option. However, the following parameters will be configurable.

Figure 10 General – Sounds.

- **Disable button sound:** enables or disables the button and action beeps. If enabled (default option) the following parameters will also be available:

- **Enable / Disable button sounds through a 1-bit object:** makes it possible to disable / resume the button beeping function in runtime by writing to a specific object (“**[General] Sounds – Disabling button sound**”).

The values (0 or 1) that will disable / resume it are parameterisable through “**Value**”.

- **Button sound after ETS Download:** sets whether the button beeping function should start up enabled (default option) or disabled after an ETS download. This makes sense if the “**Enable/Disable Button Sounds Using 1-Bit Object**” checkbox (see below) is enabled too.

- **Object for doorbell:** enables or disables the doorbell function. If enabled, a specific object (“**[General] Sounds: Doorbell**”) will be included into the project topology. The value that will trigger the sound (1 or 0) must be set in “**Value**”.
- **Object for alarm bell:** enables or disables the alarm function. If enabled, a specific object (“**[General] Sounds: Alarm**”) will be included into the project topology. The values (1 or 0) that will trigger the alarm sequence and then stop it must be entered in “**Value**”.

2.2.6 TOUCH LOCKING

The touch panel of TMD-Display View can be optionally locked and unlocked anytime by writing a configurable one-bit value to a specific object provided for this purpose. It can also be done through scene values.

While locked, user presses on the touch buttons will be ignored: no actions will be performed (and no LEDs will change their states) when the user presses on any of the controls. However, if configured, a message will be shown on the display for three seconds if the user touches a button during the lock state.

ETS PARAMETERISATION

After enabling “**Touch Locking**” (see section **¡Error! No se encuentra el origen de la referencia.**), a new tab will be incorporated into the tree on the left.

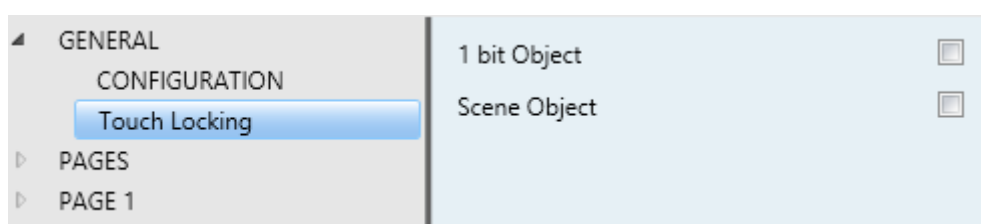


Figure 11 General - Touch Locking.

The screen contains two non-exclusive checkboxes to select how the touch panel lock/unlock should be performed:

1 bit object	<input checked="" type="checkbox"/>
Value	0 = Unlock; 1 = Lock
Scene object	<input checked="" type="checkbox"/>
Lock: Scene number	1
Unlock: Scene number	1

Figure 12 General - Touch Locking: Control.

Upon the reception of a one-bit value ("**1 bit object**").

When marked, a specific drop-down list will show up to select which value should trigger which action ("0 = Unlock; 1 = Lock" or "0 = Lock; 1 = Unlock"). These values are to be received through object "**[General] Lock Touch**".

Upon the reception of a scene value ("**Scene object**").

When marked, two specific textboxes will show up to enter the scene numbers (1 - 64) that should trigger each action. These values are to be received through the general "**[General] Scene: Receive**" object.

2.2.7 WELCOME BACK OBJECT

TMD-Display View can send a specific object (the *welcome back object*) to the KNX bus when the user presses a touch button after a significant amount of time since the last press. Sending it or not can also depend on an **additional, configurable condition** consisting in the evaluation of up to five binary objects.

Any actions that in normal operation may be executed will not be if the welcome back object is sent to the bus. Thus, if the user presses a button and this causes the welcome back object to be sent, the normal action of that button will not be triggered. On the other hand, if the additional condition is not evaluated to true, the device will react normally. Hence, the action corresponding to the button touch will be executed.

The welcome back object can consist in a **one-bit** value or a **scene** value (or both), depending on the parameterisation.

ETS PARAMETERISATION

After enabling “**Welcome Back Object**” (see section **¡Error! No se encuentra el origen de la referencia.**), a new tab will be incorporated into the tree on the left.

This screen contains the following parameters:

Figure 13 General - Welcome Back Object.

- **Timeout (since last touch):** sets the minimum time (1 to 255 seconds, 1 to 255 minutes, or 1 to 255 hours) that should elapse after the last button touch before the next one triggers the execution of the welcome back function.
- **Additional Condition:** sets if sending the welcome back object should also depend on an external condition. The option by default is “No Additional Condition”. The following are available too:
 - Do not send unless all additional conditions are 0: the welcome back object will only be sent if all the condition objects are found to have the value “0”.
 - Do not send unless all additional conditions are 1: the welcome back object will only be sent if all the condition objects are found to have the value “1”.
 - Do not send unless at least one of the additional conditions is 0: the welcome back object will only be sent if at least one of the condition objects is found to have the value “0”.

- Do not send unless at least one of the additional conditions is 1: the welcome back object will only be sent if at least one of the condition objects is found to have the value “1”.
- **Welcome Back Object (1 Bit)**: checkbox to enable the sending of a 1-bit value (through “[General] Welcome back”) when the welcome back function is triggered and the condition (if any) evaluates to true. The desired value (0 or 1) should to be set in “Value”.
- **Welcome Back Object (Scene)**: checkbox to enable the sending of a scene run request (through “[General] Scene: send”) when the welcome back function is triggered and the condition (if any) evaluates to true. The desired scene number (1 through 64) should to be entered in “Value”.

The screenshot shows a configuration window with a light blue background. It contains two main sections. The first section is titled 'Welcome Back Object (1 bit)' and has a checked checkbox. Below it is a 'Value' label followed by a dropdown menu showing 'Send 1'. The second section is titled 'Welcome Back Object (Scene)' and also has a checked checkbox. Below it is a 'Scene Number' label followed by a text input field containing the number '1' and two small arrow buttons for incrementing and decrementing the value.

Figure 14 General - Welcome Back Object - Custom Configuration.

2.3 PAGES

The Pages tab contains a single screen, Configuration.

2.3.1 CONFIGURATION

The Configuration screen under Pages allows enabling/disabling the pages that will be available on the screen and configuring some general parameters concerning the buttons and the navigation control.

ETS PARAMETERISATION

Figure 15 Pages – Configuration

- **Page n :** checkbox to enable page n . There is one checkbox for each general purpose page (pages 1 to 4), and two more for the configuration pages: General Configuration and Display. Page 1 is always enabled.
- **Action of the pair buttons:** sets how the two-button controls should behave:
 - Left = 0/Off//Decr./Down; Right = 1/On/Incr./Up,
 - Right = 1/On/Incr./Up; Left = 0/Off//Decr./Down,
 - Every button pair is configured separately, which will incorporate an additional parameter to the configuration window of each pair control to allow defining the action of each of the two buttons (unless the function of the control has been set to “Enumeration” or “[Climate] Mode”).

- **Navigation Control:** sets how the central touch area will perform the navigation between pages. The available options are:
 - Slide: the pages switch by sliding the finger across the central area between the buttons. The parameter **Action** sets the navigation direction:
 - Upward = Next Page; Downward = Previous Page.
 - Upward = Previous Page; Downward = Next Page.
 - Single button: the pages switch by pressing on the central area between the buttons. The parameter **Action** sets the navigation direction:
 - Next Page.
 - Previous Page.
 - Two buttons: the pages change by pressing in the central area between the buttons. In particular, pressing on the top central area will switch the pages in one direction while pressing on the bottom area will do it in the opposite direction. The parameter **Action** sets the navigation direction:
 - Upper = Previous Page; Lower = Next Page
 - Upper = Next Page; Lower = Previous Page..

2.4 PAGE n

When any of the general-purpose pages is enabled from the Configuration screen under the Menu tab, a new tab named **Page n** will appear, where n is the number of the page.

Under this tab, one screen (**Configuration**) will be initially displayed to let the integrator enable or disable any of the boxes in the page. Depending on that, more parameter screens will appear.

Note: figures in this section will show the parameters of a certain page (for instance, page 1). The parameters for other pages are totally analogous.

2.4.1 CONFIGURATION

TMD-Display View features **eight capacitive buttons** at the user's disposal for the execution of actions. Each can be configured to perform a different function per page.

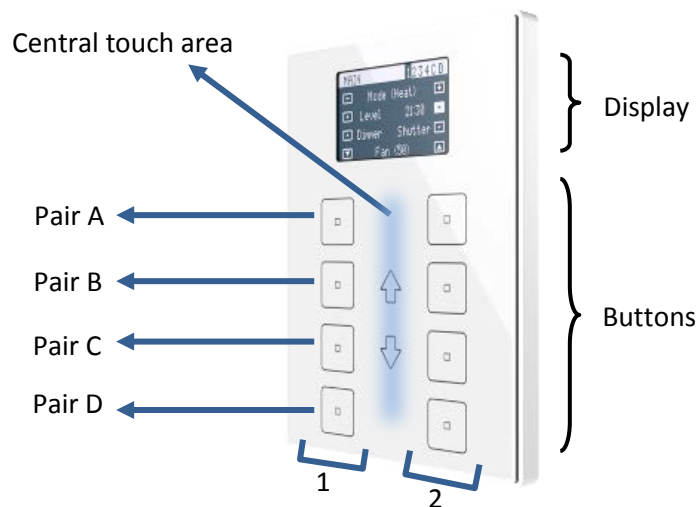


Figure 16 Touch panel

As Figure 15Figure 16 shows, three areas can be distinguished in the front panel:

- **The buttons**, grouped in pairs.
- **The display**, at the top of the front panel.
- **The central touch area**, between the two button columns.

All the buttons are identical; this brings a high level of versatility for a wide variety of applications. The following is a list of the functions that can be assigned to each pair of buttons.

● **Disabled** (none of the two buttons will react to user presses).

● **Pair Buttons** (the two buttons will work jointly):

- Switch.
- Switch + Indicator.
- Two Objects (Short Press / Long Press).
- Counter.
- Scaling.
- Float.
- Enumeration.
- Shutter.
- Light dimmer.
- [Climate] Temperature setpoint.
- [Climate] Mode.
- [Climate] Fan.
- Multimedia
- Indicator

● **Individual Buttons** (each button in the pair will work independently):

- Switch.
- Two Objects (short press / long press).
- Hold & Release.
- Scene.
- Counter constant.
- Scaling constant.
- Float constant.
- Enumeration.
- Dimmer.
- Shutter.
- Timer.
- LED indicator.

Apart from the button function itself, the integrator can select the desired behaviour of the button **LEDs**. The different **illumination modes** have been detailed in [ANNEX I: LED Illumination Modes](#).

The following sections will detail the involved configuration.

ETS PARAMETERISATION



Figure 17 Page *n* – Configuration.

As stated in section 2.1, an independent tab for the parameterisation of the buttons is shown in ETS by default, initially containing only a sub-tab named Configuration.

One drop-down list with the following options is shown per pair of buttons:

- Disabled. See section 0.
- Pair. Selecting this option brings a new tab to the tree on the left (named “**Pair X**”, where X depends on the button pair). See section 2.4.3.
- Individual. Selecting this brings the checkboxes “**Button X1**” and “**Button X2**” (where X depends on the button pair), which allow making each of the two buttons functional or not. Depending on this, up to two new tabs (named “**Button Xn**”) will be included in the tree on the left. See section 2.4.4.

Note: if a button pair is set to “Individual” but the “Button Xn” checkbox is left unmarked, that button will behave as if the pair had been set to “Disabled”.

2.4.2 DISABLED

While a button pair stays disabled, none of the two buttons will be functional: touching them will not cause the execution of actions, nor will make the associated LED light.

ETS PARAMETERISATION

This function has no related parameters.

2.4.3 PAIR

Pair buttons configured to work jointly can be assigned the following control functions:

- **Switch:** pressing one of the two buttons will make TMD-Display View send a binary value to the bus, while pressing on the other will make it send the inverse binary value. It is possible to configure which one does what.

Under a “**state-dependent**” LED illumination (see [ANNEX I: LED Illumination Modes](#)), the LED of the corresponding button will remain on/off according to the current state (on/off) of the switch. On the other hand, under a “**state-dependent (both LEDs)**” LED illumination, both of them will remain on while the switch is in the “on” state, and off while in the “off” state.

- **Switch + Indicator:** the functionality of the buttons is analogous to that of the the switch function although it is also possible to define a numerical indicator (with a configurable type and size), whose value is shown on the display, in parentheses, next to the name configured for the button pair.
- **Two Objects (Short Press / Long Press):** permits sending specific binary values both after a short or a long press on any of the two buttons (i.e., they will work as a joint control; for independent buttons, please configure them as individual). Different objects will be used for the short and long presses.

Moreover, it is possible (in parameters) to make the “**state-dependent**” and “**state-dependent (both LEDs)**” LED illumination modes (see [ANNEX I: LED Illumination Modes](#)) depend on either one object or the other. However, if “**LED Illumination Control (All Buttons)**” (section 2.3.1) has been set to “state-dependent (where available)” only the short press object will be considered

- **Scaling, Counter and Float:** user touches over the buttons will trigger the sending of a certain numerical value to the bus (this value will be progressively increased or decreased every time the user touches one button or the other) through the control object.

The display continuously shows the current numeric value. Moreover, it will reflect the new values as the user interacts with the control, however values are actually sent to the bus 1.5 seconds after the last press.

The “**state-dependent**” and “**state-dependent (both LEDs)**” LED illumination modes are not available for this function (only the “**regular**” and “**dedicated object**” LED illumination are available). See [ANNEX I: LED Illumination Modes](#) for details.

- **Enumeration:** pressing the buttons will make TMD-Display View send a one-byte (unsigned) value to the bus. It is possible to distinguish up to 6 different values. Successively pressing on the buttons will make TMD-Display View walk through the different values, displaying on the screen the text associated to each value.

As the above functions, this type of control provides parameter to **link the control with a box in the display** in order to show the current value on the display. Again, values are sent to the bus 1.5 seconds after the last press.

The “**state-dependent**” and “**state-dependent (both LEDs)**” LED illumination modes are not available for this function (only the “**regular**” and “**dedicated object**” LED illumination are available). See [ANNEX I: LED Illumination Modes](#) for details.

- **Shutter:** this option permits making use of the two buttons to control a shutter actuator connected to the bus. Two alternative control methods are possible:
 - Standard: a long press will make the device send to the KNX bus an order to start moving the shutter (upwards or downwards, depending on the button), while a short press will make it send a stop order (which will be interpreted as an order to step up or to step down –depending on the button– if the shutter was not in motion and such function is available).
 - Hold & Release: as soon as the button is held, the device will send the KNX bus an order to start moving the shutter (upwards or downwards,

depending on the button). Once the button is released, it will send an order to stop the shutter.

The “**state-dependent**” and “**state-dependent (both LEDs)**” LED illumination modes are not available for this function (only the “**regular**” and “**dedicated object**” LED illumination are available). See [ANNEX I: LED Illumination Modes](#) for details.

- **Dimmer:** short-pressing one of the two buttons will make TMD-Display View send a switch-on order to the bus, while doing so on the other button will make it send a switch-off order.

Long presses will make it send a step dimming order (the value of which is configurable) to make a dimmer increase or decrease the light level (and a stop order as soon as the user releases the push button). It is possible to configure which button does what.

Under a “**state-dependent**” LED illumination (see [ANNEX I: LED Illumination Modes](#)), the LED of the corresponding button will remain on/off according to whether the current value of the light level status object (which should be updated by the actual dimmer) is greater than 0% or not. On the other hand, under a “**state-dependent (both LEDs)**” LED illumination, both LEDs together will remain on or off depending on such value.

- **[Climate] Temperature setpoint:** permits controlling the temperature setpoint of an external thermostat by means of a button pair and the display.

This function can also be **linked to a box in the display** in order to show the current setpoint value. The box will reflect the value (in °C) of the status object, which gets automatically updated after sending control orders (that is, after pressing the buttons), being even possible to receive values from the bus, for example, from the corresponding setpoint status object from the external thermostat.

The “**state-dependent**” and “**state-dependent (both LEDs)**” LED illumination modes are not available for this function (only the “**regular**” and “**dedicated object**” LED illumination are available). See [ANNEX I: LED Illumination Modes](#) for details.

- **[Climate] Mode:** pressing one of the two buttons will make TMD-Display View send climate mode values to the bus. Two mode control types are possible: **Heat / Cool** and **Extended** (HVAC).

As in previous functions, the current mode is shown in the display.

The “**state-dependent**” and “**state-dependent (both LEDs)**” LED illumination modes are not available for the Extended mode (only the “**regular**” and “**dedicated object**” LED illumination are available). For the Heat / Cool mode all the illumination options are available. See ANNEX I: LED Illumination Modes for details.

- **[Climate] Fan:** pressing one of the two buttons will make TMD-Display View send a binary value to the bus that allows changing the fan speed (up to **five levels** available). This value can be either binary (increase / decrease orders), scaling (0% - 100%) or enumerated.

Furthermore, it is possible to configure the **speed 0** as an additional fan level, and/or another level to activate the **Auto mode**.

The order to activate the auto mode can be sent over a one-bit object, independent from the fan control object, or alternatively consist in the speed level 0. In the latter case, level 0 will imply triggering the Auto mode.

The three available control types are:

- “1 bit (decrease/increase)”: one-bit values (increase or decrease) will be sent according to the action configured for each button.

In this case, switching from the current fan level to a non-consecutive level involves going through each of the intermediate levels.

- “Scaling”: button presses will make the device send percentage values through the corresponding one-byte object.

Every press will activate the subsequent (or preceding) fan level, depending on the current one. Therefore, the percentage values sent will depend on the number of the available levels and on whether the Auto mode and the speed level 0 are available or not.

- **“Enumeration”**: this control is similar to the one above, although integer values are sent instead of scaling values.

Table 1 shows the relation between the scaling values and the enumeration values, depending on the number of speed levels configured (note that if the Auto mode or the speed level 0 are allowed one more level is added, corresponding to value 0).

% Values based on fan levels						
Speed Levels		1	2	3	4	5
Enumeration value	1	100	50,2	33,3	25,1	20
	2	-	100	66,7	50,2	40
	3	-	-	100	75,3	60
	4	-	-	-	100	80
	5	-	-	-	-	100

Table 1 Fan levels.

For the scaling and enumeration controls, the object is sent to the bus slightly delayed, once the desired value has been set by the user. The current value is continuously shown on the display.

Under a **“state-dependent”** LED illumination, the LED of the corresponding button will remain on/off according to the current state of the fan. On the other hand, under a **“state-dependent (both LEDs)”** LED illumination, both of them will remain on or off while the fan level is greater than 0 or not, respectively. The **“regular”** and **“dedicated object”** LED illumination modes are also available. See ANNEX I: LED Illumination Modes for details.

- **Multimedia**: tapping on one of the two buttons will make TMD-Display View send a binary value to the bus, while pressing on the other will make it send the inverse binary value. It is possible to configure which one does what. Also, this control has a 14-byte object associated, whose value is displayed on the screen between the icons of the buttons.

Note: after a failure of the KNX bus, any value that may have been received through the 14-byte object will be lost.

- **Indicator:** in this case, the two push buttons are functionless, although the display will show the value of an indicator (with a configurable type and size) on the right size.

ETS PARAMETERISATION

When a button pair has been configured to work as a joint control, a specific tab (“**Pair X**”) becomes available under “Buttons” in the tree on the left.

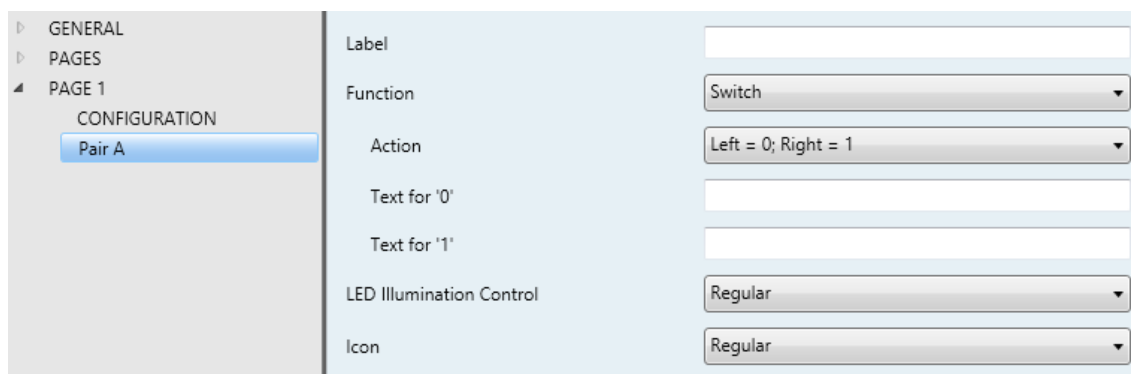


Figure 18 Buttons - Pair A.

Parameters common to all types of controls are:

- **Function:** sets the desired function for the button pair: “Switch”, “Switch + Indicator”, “Two objects (short press / long press)”, “Counter”, “Scaling”, “Float”, “Enumeration”, “Shutter”, “Dimmer”, “[Climate] Temperature Setpoint”, “[Climate] Mode” or “[Climate] Fan”, “Multimedia” or “Indicator”.
- **Label:** textbox to enter a string (up to 10 characters) which is displayed on the screen between the icons of the buttons, in order to identify the control.
- **Icon:** allows selecting the icon that is drawn on each button on the display. Please refer to ANNEX II. Button Icons for a list of available icons.

Depending on the function, some more parameters are shown, as described next. Please note that in the next pages the general notation “[Pn] [X]” is used for the name of the communication objects, as “X” depends on the button pair (A, B, etc.). On the other hand, “Pn” will refer to the page number.

Switch

Label	<input type="text"/>
Function	Switch ▼
Action	Left = 0; Right = 1 ▼
Text for '0'	<input type="text"/>
Text for '1'	<input type="text"/>
LED Illumination Control	Dedicated Object ▼
Object Value	0 = Off; 1 = On ▼
Icon	Regular ▼

Figure 19 Pair Buttons – Switch.

- **Action:** assigns each of the two buttons the value to be sent through “[Pn] [X] Switch” (which has the Write flag enabled, so the state of the switch can be updated from external devices). The options are “Left=0; Right=1” and “Left=1; Right=0”.

Note: *this parameter will remain hidden unless having selected “Every button pair is configured separately” in **Action of the pair buttons** (see section 2.3.1).*

- **Text for ‘0’ and Text for ‘1’:** sets the text to be displayed between the icons of the buttons when the values ‘0’ and ‘1’ are sent, respectively.
- **LED illumination control:** sets the behaviour of the LEDs on the buttons. The options are “Regular”, “State-dependent”, “State-dependent (both LEDs)” and “Dedicated Object”.

In case of selecting the latter, the object “[Pn] [X] Led On/Off” will be included in the project topology and a new parameter to select the value to switch off and on the LED shows up:

LED Illumination Control	Dedicated Object ▼
Object Value	0 = Off; 1 = On ▼

Figure 20 LED illumination control - Dedicated object.

Switch + Indicator

Label	<input type="text"/>
Function	Switch + Indicator ▼
Action	Left = 0; Right = 1 ▼
Indicator Type	Counter ▼
Size	1 byte ▼
Signed	<input type="checkbox"/>
Units	<input type="text"/>
LED Illumination Control	State-dependent ▼
Icon	Regular ▼

Figure 21 Pair buttons – Switch + Indicator.

- **Action:** assigns each of the two buttons the value to be sent through “[Pn] [X] Switch” (which has the Write flag enabled, so the state of the switch can be updated from external devices). The options are “Left=0; Right=1” and “Left=1; Right=0”.

Note: *this parameter will remain hidden unless having selected “Every button pair is configured separately” in **Action of the pair buttons** (see section 2.3.1).*

- **Indicator Type:** sets the type and size of the object that receives the value of the indicator to be shown on the display. The options are:
 - “Counter”: the value is received through the communication object “[Pn] [X] 1-byte / 2-byte / 4-byte Signed / Unsigned Int Indicator”. After selecting Counter, the following parameters appear:
 - **Size:** sets the size of the object, “1 byte”, “2 bytes” or “4 bytes Signed Int”

- **Signed:** checkbox to set if the values will be signed or unsigned.
- **Units:** text that will be displayed next to the indicator to identify the units of the value. Up to 4 characters are available.
- **“Scaling”:** the value is received through the communication object “[Pn] [X] 1-byte Indicator (Scaling)”
- **“Temperature”:** the value is received through the communication object “[Pn] [X] Temperature Indicator”.
- **LED illumination control:** sets the behaviour of the LEDs on the buttons. The options are “Regular”, “State-dependent”, “State-dependent (both LEDs)” and “Dedicated Object”.

Two Objects (short Press / long press)

Label	<input type="text"/>
Function	Two Objects (Short Press/Long Press) ▼
Action on Short Press	Left = 0; Right = 1 ▼
Text for '0'	<input type="text"/>
Text for '1'	<input type="text"/>
Action on Long Press	Left = 0; Right = 1 ▼
Long Press Threshold Time	6 <input type="text"/> ds
LED Illumination Control	Dedicated Object ▼
Object Value	0 = Off; 1 = On ▼
Icon	Regular ▼

Figure 22 Pair Buttons - Two Objects (Short Press / Long Press).

- **Action on short press:** sets the value that will be sent through “[Pn] [X] Two switches - Short press” after the user short-presses one of the two buttons.
 - “Left=0; Right=1”.
 - “Left=1; Right=0”.

Note: *this parameter will remain hidden unless having selected “Every button pair is configured separately” in **Action of the pair buttons** (see section 2.3.1).*

- **Text for ‘0’ and Text for ‘1’:** sets the text to be displayed between the icons of the buttons when the values ‘0’ and ‘1’ are sent (respectively) after short-pressing them.
- **Action on long press:** sets the value that will be sent through “[Pn] [X] Two switches: Long press” after the user long-presses one of the two buttons.
 - “Left=0; Right=1”.
 - “Left=1; Right=0”.

Note: *this parameter will remain hidden unless having selected “Every button pair is configured separately” in **Action of the pair buttons** (see section 2.3.1).*

- **Long press threshold time:** sets the minimum time the user should hold the button in order to consider it a long press. The available range is 5 to 50 tenths of a second, being 6 tenths the default value.
- **LED illumination control:** analogous to the homonymous parameter for the Switch function (see above). In this case, however, selecting “State-dependent” or “State-dependent (both LEDs)” brings an additional parameter, “**State Object**”, to make the state of the LEDs correspond either to object “[Pn] [X] Two switches: Short press” (“Short Press Object”) or to object “[Pn] [X] Two switches: Long press” (“Long Press Object”).

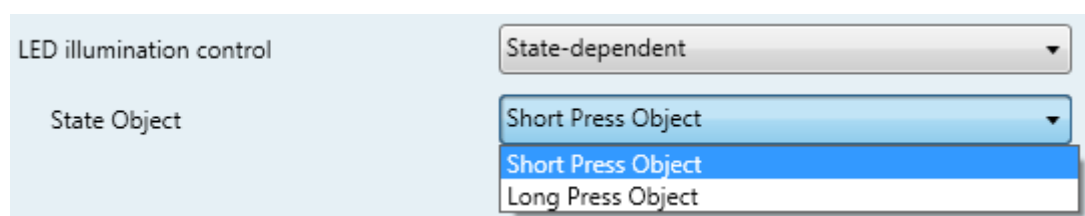


Figure 23 Pair Buttons - Two objects (short press / long press) - LED illumination control.

Counter

Label	<input type="text"/>
Function	Counter
Action	Left = Decrease; Right = Increase
Size	1 byte
Signed	<input type="checkbox"/>
Minimum Value	0
Maximum Value	255
Increment on Short Press	1
Increment on Long Press	10
LED Illumination Control	Regular
Icon	Regular

Figure 24 Pair Buttons – Counter.

● **Action:** assigns each of the two buttons the order to be sent:

- “Left=Decrease; Right=Increase”.
- “Left=Increase; Right=Decrease”.

Note: *this parameter will remain hidden unless having selected “Every button pair is configured separately” in **Action of the pair buttons** (see section 2.3.1).*

● **Size:** sets the desired size of the object, “1 byte” or “2 bytes”.

● **Signed:** checkbox to set whether the object will be signed or unsigned.

The value ranges depend on the above two parameters:

- 1 byte unsigned: 0 – 255.
- 1 byte signed: -128 – 127.
- 2 bytes unsigned: 0 – 65535.
- 2 bytes signed: -32768 – 32767.

- **Minimum Value:** sets which value from the available range will be the minimum value permitted by the control.
- **Maximum Value:** sets which value from the available range will be the maximum value permitted by the control.
- **Increment on Short Press:** sets the increase or decrease to be applied to the current value on every short press over the increase or decrease buttons, respectively (i.e., the smaller the increment, the more presses will be required).
- **Increment on Long Press:** sets the increase or decrease to be applied to the current value on every long press over the increase or decrease buttons, respectively (i.e., the smaller the increment, the more presses will be required).
- **LED illumination control:** analogous to the homonymous parameter for the previous functions (see above). In this case, the only options are “Regular” and “Dedicated Object”.

The counter values will be sent to the bus through the objects “[Pn] [X] [size] **Unsigned Int Value**” or “[Pn] [X] [size] **Signed Int Value**”.

Scaling

Label	<input type="text"/>
Function	Scaling
Action	Left = Decrease; Right = Increase
Minimum Value	0 <input type="button" value="↑"/> <input type="button" value="↓"/> %
Maximum Value	100 <input type="button" value="↑"/> <input type="button" value="↓"/> %
Increment on Short Press	1 <input type="button" value="↑"/> <input type="button" value="↓"/> %
Increment on Long Press	10 <input type="button" value="↑"/> <input type="button" value="↓"/> %
LED Illumination Control	Regular
Icon	Regular

Figure 25 Pair Buttons – Scaling.

- **Action:** assigns each of the two buttons the order to be sent:

- “Left=Decrease; Right=Increase”.
- “Left=Increase; Right=Decrease”.

Note: *this parameter will remain hidden unless having selected “Every button pair is configured separately” in **Action of the pair buttons** (see section 2.3.1).*

- **Minimum Value:** sets which value from the available range will be the minimum value permitted by the control. The available range is 0-100.
- **Maximum Value:** sets which value from the available range will be the maximum value permitted by the control after. The available range is 0-100.
- **Increment on Short Press:** sets the increase or decrease to be applied to the current value on every short press over the increase or decrease buttons, respectively (i.e., the smaller the increment, the more presses will be required). The available range is 0-100.
- **Increment on Long Press:** sets the increase or decrease to be applied to the current value on every long press over the increase or decrease buttons, respectively (i.e., the smaller the increment, the more presses will be required). The available range is 0-100.
- **LED illumination control:** analogous to the homonymous parameter for the previous functions (see above). In this case, the only options are “Regular” and “Dedicated Object”.

The percentage values will be sent to the bus through **“[Pn] [X] 1-byte Scaling Value”**.

Float

- **Action:** assigns each of the two buttons the order to be sent:

- “Left=Decrease; Right=Increase”.
- “Left=Increase; Right=Decrease”.

Note: *this parameter will remain hidden unless having selected “Every button pair is configured separately” in **Action of the pair buttons** (see section 2.3.1).*

Label	<input type="text"/>
Function	Float
Action	Left = Decrease; Right = Increase
Minimum Value	-671088,625
Maximum Value	670760,9375
Increment on Short Press	0,5
Increment on Long Press	1
LED Illumination Control	Regular
Icon	Regular

Figure 26 Pair Buttons – Float.

- **Minimum Value:** sets which value from the available range (-671088.625 to 670760.9375) will be the minimum value permitted by the control.
- **Maximum Value:** sets which value from the available range (-671088.625 to 670760.9375) will be the maximum value permitted by the control.
- **Increment on Short Press:** sets the increase or decrease to be applied to the current value on every short press over the increase or decrease buttons, respectively (i.e., the smaller the increment, the more presses will be required). The available range is 0.1 – 670760.93750 (0.5 by default).
- **Increment on Long Press:** sets the increase or decrease to be applied to the current value on every long press over the increase or decrease buttons, respectively (i.e., the smaller the increment, the more presses will be required). The available range is 0.1 – 670760.93750 (1 by default).
- **LED illumination control:** analogous to the homonymous parameter for the previous functions (see above). In this case, the only options are “Regular” and “Dedicated Object”.

The values will be sent to the bus through object “[Pn] [X] 2-byte Float Value”.

Enumeration

Label	<input type="text"/>
Function	Enumeration ▼
# Enums	1 <input type="button" value="↑"/> <input type="button" value="↓"/>
Value 1	0 <input type="button" value="↑"/> <input type="button" value="↓"/>
Text 1	<input type="text"/>
LED Illumination Control	Regular ▼
Icon	Regular ▼

Figure 27 Pair Buttons – Enumeration.

- **# Enums:** number of states (up to 6) that will be distinguished. For every distinguished state, one parameter named **Value *j*** will become available, in order to set the value sent through the communication object.
- **Value *j*:** numerical value (0 – 255) that will be sent through the control object when the user, after pressing the buttons, sets the control to state *j*. The actual sending will not be performed until 1.5 seconds after the last press.
- **Text *j*:** text to be displayed between the icons of the buttons, to indicate the current enumeration value.
- **LED illumination control:** analogous to the homonymous parameter for the previous functions (see above). In this case, the only options are “Regular” and “Dedicated Object”.

The values will be sent to the bus through object “[Pn] [X] Enumeration”.

Shutter

The move orders will be sent through “[Pn] [X] Move Shutter”, while the stop orders will be sent through “[Pn] [X] Stop Shutter/Step” (for Standard type) or “[Pn] [X] Stop Shutter” (for Hold & Release type).

The parameters for this function are:

Label	<input type="text"/>
Function	Shutter ▼
Action	Left = Downwards; Right = Upwards ▼
Type	Standard ▼
LED Illumination Control	Regular ▼
Icon	Regular ▼

Figure 28 Pair Buttons – Shutter.

- **Action:** assigns each of the two buttons the order to be sent:

- “Left=Down; Right=Up”.
- “Left=Up; Right=Down”.

Note: *this parameter will remain hidden unless having selected “Every button pair is configured separately” in **Action of the pair buttons** (see section 2.3.1).*

- **Type:** sets the desired behaviour of the buttons, “Standard” or “Hold & Release” (the differences between them have been explained in previous pages).
- **LED illumination control:** analogous to the homonymous parameter for the previous functions (see above). In this case, the only options are “Regular” and “Dedicated Object”.

Dimmer

Label	<input type="text"/>
Function	Dimmer ▼
Action	Left = Off/Decrease; Right = On/Increase ▼
Dimming Step	100% ▼
LED Illumination Control	State-dependent ▼
Icon	Regular ▼

Figure 29 Pair Buttons – Dimmer.

The switch orders will be sent through the “[Pn] [X] Light On/Off” one-bit object, while the increase/decrease orders will be through the “[Pn] [PX] Light Dimming” four-bit object.

On the other hand, the “[Pn] [X] Light Indicator” one-byte object may be linked to the light level status object of the dimmer (in fact, this object is only intended to receive values from the bus, not to send them). As explained in 2.4.3, the state-dependent LED lighting will be determined by the value of this object (LED off at 0% and on at any other level).

The parameters for this function are:

- **Action:** assigns each of the two buttons the order to be sent:

- “Left=Off; Right=On”.
- “Left=On; Right=Off”.

Note: *this parameter will remain hidden unless having selected “Every button pair is configured separately” in **Action of the pair buttons** (see section 2.3.1).*

- **Step:** defines the dimming step (“100%”, “50%”, “25%”, “12.5%”, “6.25%”, “3.1%” or “1.5%”) to be sent to the light dimmer with every increase / decrease order.

Note: *since dimmers typically do not apply the new light level immediately (i.e., the step regulation is performed progressively) and since TMD-Display View sends an order to interrupt the step dimming once the user releases the button, it is advisable to configure a step of 100%. This way, the user can perform any dimming step by simply leaving the button pressed and then releasing it without needing to make successive button presses.*

- **LED illumination control:** analogous to the homonymous parameter for the “Switch” and “Two Objects (short press / long press)” function (see above).

[Climate] Temperature Setpoint

Label	<input type="text"/>
Function	[Climate] Temperature Setpoint ▼
Action	Left = Decrease; Right = Increase ▼
Minimum Value	10 °C
Maximum Value	30 °C
LED Illumination Control	Regular ▼
Icon	Regular ▼

Figure 30 Pair Buttons – [Climate] Temperature Setpoint.

- **Action:** assigns each of the two buttons the order to be sent:

- “Left=Decrease; Right=Increase”.
- “Left=Increase; Right=Decrease”.

Note: *this parameter will remain hidden unless having selected “Every button pair is configured separately” in **Action of the pair buttons** (see section 2.3.1).*

- **Minimum Value:** sets which value from the available range (-273 to 670760.9375; 10 by default) will be the minimum value permitted by the control.
- **Maximum Value:** sets which value from the available range (-273 to 670760.9375; 30 by default) will be the maximum value permitted by the control after.
- **LED illumination control:** analogous to the homonymous parameter for the previous functions (see above). In this case, the only options are “Regular” and “Dedicated Object”.

The values are sent to the bus through **“[Pn] [X] (Climate) Temperature setpoint”**.

[Climate] Mode

Label	<input type="text"/>
Function	[Climate] Mode
Mode Type	Heat/Cool
Text for 'Cool'	Cool
Text for 'Heat'	Heat
LED Illumination Control	State-dependent
Icon	Regular

Figure 31 Pair Buttons – [Climate] Mode.

● **Mode Type:** sets mode control type: “Heat/Cool” or “Extended” (HVAC).

- “Heat/Cool”: pressing one of the two buttons will make TMD-Display View send a one-bit value to the bus (Left=Cool, value “0”; Right=Heat, value “1”).
- “Extended”: pressing on the two buttons will make TMD-Display View walk through the different HVAC modes and send a one-byte value to the bus according to the selected mode. Which of the modes should be available can be set through five specific checkboxes.

Mode Type	Extended
Auto	<input checked="" type="checkbox"/>
Text for 'Auto'	Auto
Heat	<input checked="" type="checkbox"/>
Text for 'Heat'	Heat
Cool	<input checked="" type="checkbox"/>
Text for 'Cool'	Cool
Fan	<input checked="" type="checkbox"/>
Text for 'Fan'	Fan
Dry	<input checked="" type="checkbox"/>
Text for 'Dry'	Dry

Figure 32 Pair Buttons – [Climate] Mode – Mode Type: Extended.

- **Text for '[mode]'**: text displayed between the icons of the buttons to indicate the current mode.
- **LED illumination control**: sets the behaviour of the LEDs on the buttons. The options are "Regular", "State-dependent", "State-dependent (both LEDs)" and "Dedicated Object".

The values will be sent to the bus through objects "[Pn] [X] (Climate) Mode" or "[Pn] [X] (Climate) Mode - Extended".

[Climate] Fan

Label	<input type="text"/>
Function	[Climate] Fan
Action	Left = Decrease; Right = Increase
Control Type	1 bit (decrease/increase)
Speed Levels	2
Cyclical	<input type="checkbox"/>
Auto Mode	<input type="checkbox"/>
Allow Speed 0	<input type="checkbox"/>
LED Illumination Control	State-dependent
Icon	Regular

Figure 33 Pair Buttons – [Climate] Fan.

- **Action**: assigns each of the two buttons the order to be sent:

- "Left = Decrease; Right = Increase".
- "Left = Increase; Right = Decrease".

Note: this parameter will remain hidden unless having selected "Every button pair is configured separately" in **Action of the pair buttons** (see section 2.3.1).

- **Control Type**: sets the type of the communication objects that will control the fan level.

- **“1 bit (decrease/increase)”**: orders of speed increase/decrease are sent through the one-bit object “[Pn] [X] Fan control (Type: 1 bit)”.
- **“Scaling”**: scaling values are sent through the one-byte object “[Pn] [X] Fan control (Type: scaling)”.
- **“Enumeration”**: integer values are sent through the one-byte object “[Pn] [X] Fan control (Type: Enumeration)”. When selecting this type of control the following parameter appears:
 - **Text Before Each Level**: text displayed, between the icons of the buttons, preceding the selected level.

For either type of control an indicator object is added to show its value on the display: “[Pn] [X] Fan Indicator”.

- **Speed levels**: sets how many speed levels will be available in the control. 1 to 5 levels.
- **Cyclical**: sets whether scrolling through the speed levels is circular or not. If checked, an increase order in the maximum level switches to the minimum and vice versa (in the minimum level a decrease order switches to the maximum).
- **Auto Mode**: sets whether the fan Auto mode will be available or not. If checked, the following parameters are also shown.
 - **Text for ‘Auto’**: text displayed between the icons of the buttons to indicate that the ‘Auto’ mode has been selected.
 - **Dedicated Object for Auto Mode**: sets how the Auto mode should be activated. In the absence of a dedicated object, the Auto fan mode will be activated by switching to fan level 0.

Auto Mode	<input checked="" type="checkbox"/>
Text for 'Auto'	Auto
Dedicated Object for Auto Mode	<input type="checkbox"/>
Allow Speed 0 (Auto)	<input checked="" type="checkbox"/>

Figure 34 Pair Buttons – [Climate] Fan – Auto Mode.

In this case the fan levels activated by short presses are:

Auto Mode (0)	Minimum	Medium	Maximum
-----------------	---------	--------	---------

On the other hand, marking the checkbox enables the one-bit object “[Pn] [X] Fan control – Auto mode”, which will trigger the Auto mode when it receives the value “1” (or “0”, depending on the subsequent parameter, “Value to Set the Auto Mode”).

Figure 35 Pair Buttons – [Climate] Fan – Dedicated Object for Auto Mode.

However, two alternative (and mutually exclusive) methods are possible to activate this mode:

- By short press: the Auto mode can be reached as a further level above the maximum one. In this case the fan levels activated by short presses are (note that **speed 0 is optional**):

(0)	Minimum	Medium	Maximum	Auto Mode
-------	---------	--------	---------	-----------

- By long press on any of the buttons of the control (requires checking **Long Press to Activate Auto Mode**).

A further long press deactivates back the Auto mode and resumes the previous fan level. Instead, a short press deactivates the Auto mode and switches to the next level (or previous, depending on the button). In this case the fan levels reachable through short presses are (**speed 0 is optional**):

(0)	Minimum	Medium	Maximum
-------	---------	--------	---------

(Here, the Auto mode is only activated by long press).

- **Allow speed 0:** sets whether the speed level 0 will be present or not. When the **Auto Mode without a dedicated object** has been configured, this option will be necessarily activated.
- **LED illumination control:** analogous to the homonymous parameter for the previous functions (see above). The options are “Regular”, “State-dependent”, “State-dependent (both LEDs)” and “Dedicated Object”.

Multimedia

Label	<input type="text"/>
Function	Multimedia
Action	Left = Stop/Backward; Right = Start/Forward
Icon	Regular

Figure 36 Pair Buttons – Multimedia.

- **Action:** assigns each of the two buttons the order to be sent:
 - “Left = Stop/Backward; Right = Start/Forward”.
 - “Left = Start/Forward; Right = Stop/Backward”.

Note: *this parameter will remain hidden unless having selected “Every button pair is configured separately” in **Action of the pair buttons** (see section 2.3.1).*

The orders will be sent to the bus through the object “[Pn] [X] 1-bit Multimedia Control”, while the indicator values should be received through the object “[Pn] [X] Multimedia Text Indicator”.

Indicator

Label	<input type="text"/>
Function	Indicator ▼
Indicator Type	Binary ▼
Text for '0'	<input type="text"/>
Text for '1'	<input type="text"/>

Figure 37 Pair Buttons – Indicator

- **Indicator Type:** sets the type and size of the object that receives the value of the indicator to be shown on the display. The options are:
 - **“Binary”:** the value is received through the object “[Pn] [X] **Binary Indicator**”. When selecting Counter, the following parameter appears:
 - **Text for ‘0’ and Text for ‘1’:** set the text displayed, between the icons of the buttons, after receiving the value ‘0’ or the value ‘1’, respectively.
 - **“Counter”:** the value is received through the communication object “[Pn] [X] **1-byte/2-byte/4-byte Signed/Unsigned Int Indicator**”. When selecting Counter, the following parameters appear:
 - **Size:** sets the desired size of the object, “1 byte”, “2 bytes” or “4 bytes Signed Int”
 - **Signed:** checkbox to set if the allowed values are signed or unsigned.
 - **Units:** text that will be displayed next to the indicator to identify the units of the value. Up to 4 characters are available.
 - **“Scaling”:** the value is received through the “[Pn] [X] **1-byte Indicator (Scaling)**” communication object.
 - **“Enumeration”:** the value is received through the “[Pn] [X] **Enumerated Indicator**” communication object. When selecting the type Enumeration, the following parameters appear:
 - **Value j :** numeric value (0 – 255) that corresponds to element j .

- **Text j :** text to be displayed between the icons of the buttons to indicate that element j is currently selected.
- **“Float”:** the value is received through the “[Pn] [X] 2-byte Float Indicator” object. When selecting the type Float, the following parameter appears:
 - **Units:** text that will be displayed next to the indicator to identify the units of the value. Up to 4 characters are available.

2.4.4 INDIVIDUAL

Buttons configured to work as individual (separate) controls can be assigned any of the following control functions from the corresponding **Button Xn** tab:

- **Switch:** whenever the user touches the button, a binary value will be sent to the KNX bus. This value is configurable and may be 0 or 1, or alternate with every touch according to the sequence $1 \rightarrow 0 \rightarrow 1 \rightarrow \dots$

Under a “**state-dependent**” LED illumination, the LED will remain on/off according to the current state (on/off) of the object.

- **Two Objects (Short Press / Long Press):** specific binary values will be sent both after a short or a long press (a different object will be used in each case). Under a “**state-dependent**” LED illumination, the LED will remain on/off according to the current state (on/off) of either one object or the other, which can be configured in parameters. However, if “**LED Illumination Control (All Buttons)**” (see section 2.3.1) has been set to “state-dependent (where available)”, only the short press object will apply.

- **Hold & Release:** as soon as the user touches the button, a binary value (“0” or “1”, configurable) will be sent to the KNX bus. And as long as the user releases the button, another value (“0” or “1”, also configurable) will be sent through the same object.

The “**state-dependent**” LED illumination mode is not available for this function.

- **Scene:** after the user touches the button, an order to run a specific scene (configurable) will be sent to the bus. If enabled in parameters, orders to save

the scene can also be sent to the bus after a three-second press on the button. The “**state-dependent**” LED mode is not available for this function.

- **Counter Constant:** sends an integer value (configurable) to the bus when the user touches the button. This value can be one-byte or two-byte sized, as well as signed or unsigned. The available ranges are shown next.

	1-byte	2-byte
Unsigned	0 – 255.	0 – 65535
Signed	-128 – 127.	-32768 – 32768.

Table 2 Counter Constant values

The “**state-dependent**” LED illumination mode is analogous as for the Scaling Constant function.

- **Scaling Constant:** sends a percentage value (configurable) to the bus when the user touches the button. Under a “**state-dependent**” LED illumination, the LED will remain on/off depending on whether the current value of the object matches the one parameterised. This object can also be written from the bus, which will update the LED according to the new value.
- **Float Constant:** sends a two-byte floating point value (configurable) to the bus when the user touches the button. The available range is -671088.625 to 670760.9375.

The “**state-dependent**” LED illumination mode is analogous as for the Scaling Constant and Counter Constant functions.

- **Enumeration:** implements a one-button enumeration control that sends one-byte (unsigned) values to the KNX bus. This control is completely analogous to the Enumeration control in pair buttons (see section 2.4.3), however the object is sent to the bus when the button is pressed (not 1.5 seconds later). Moreover, in this case there is no option to link it to the display.

The “**state-dependent**” LED mode is not available for this function.

- **Dimmer:** implements a one-button light control that sends orders to the KNX bus, which can then be executed by light dimmers. These orders consist in:

- Switch-on / Switch-off orders (on short presses).
- Step dimming orders (on long presses) and the subsequent stop order once the button is released.

Being a one-button control, the **switch orders will alternate** (on / off) for every short press, and so will do the step dimming orders (increase / decrease) for every long press. However, there are some exceptions to this alternation:

- On a long press: an increase dimming order will be sent if the light is found to be off (according to the status object). On the other hand, a decrease order will be sent if it is found to be 100%.
- On a short press: a switch-on order will be sent if the light is found to be off (according to the status object). On the other hand, a switch-off order will be sent if it is found to be on (value greater than 0%).

Note that the device considers that the **current light level** is the value of a specific one-byte object provided to be written from the KNX bus (i.e., to receive feedback from the dimmer). This object is internally updated after a short or long press, but linking it to the real dimmer status is highly advisable.

Under a “**state-dependent**” LED illumination, the LED will remain on/off according to the value of the aforementioned status object (i.e., off when the value is 0% and on in any other case).

Note: *after a bus recovery, the light dimmer should send back the status object so the control and the LED update their own state, instead of simply recovering the previous one.*

- **Shutter:** implements a one-button shutter control that sends orders to the KNX bus, which can then be executed by an actuator.

Two control types can be configured:

- Standard: the device will react to both long and short presses, being possible to send the bus the following commands:
 - Move (raise / lower) orders (on **long presses**).

- Stop / Step orders (on **short presses**).

Being a one-button control, the direction of the motion will alternate (upwards / downwards) for both the move and the step orders after every long press. However, there are some exceptions to this alternation:

- On a short press: a step-up order will be sent if the last long press made the shutter move up, or if the current position is found to be 100%. On the other hand, a step-down order will be sent if the last long press made the shutter move down or if the current position is found to be 0%.
- On a long press: a move-up order will be sent if the last short press caused a step-down order or if the current position is found to be 100%. On the other hand, a move-down order will be sent if the last short press caused a step-up order or if the current position is found to be 0%.

As usual in the KNX standard, **stop/step orders** are interpreted by the actuators as a request to move the slats one step up or down (in case the shutter is still) or as a request to interrupt the motion of the shutter (in case it is already moving up or down).

TMD-Display View is aware of the current **position of the shutter** through a specific object which should be linked to the analogous object of the shutter actuator in order to receive feedback.

- Hold & Release: the device will send an order to move the shutter as soon as the button is touched, and the order to stop it as soon as it is released. Hence, short or long touches have the same effect: the shutter will remain in motion as long as the user keeps holding the button.

The direction of this motion (upwards or downwards) will **alternate** with every touch, according to the following sequence: downwards → upwards → downwards → ...

However, there are some exceptions to this alternation:

- If the position of the shutter is found to be 0%, the next order will lower the shutter.

- If the position of the shutter is found to be 100%, the next order will raise the shutter.

TMD-Display View is aware of the current **position of the shutter** through a specific object which should be linked to the analogous object of the shutter actuator, in order to receive feedback. This object is initialised with value “50%” after a download or a bus failure; therefore, the actuator is required to update it with the real value after the bus recovery.

The “**state-dependent**” LED illumination mode is not available for this function.

- **Timer:** this function allows sending a binary value (“1” or “0”) or a scene command to the KNX bus at a certain time, either as a single sending (one-time execution), as a periodic sending or after a countdown time.
- **LED indicator:** user presses will not trigger any function although the LED will turn on or off depending on the values received from the bus.

ETS PARAMETERISATION

When an individual button has been enabled, a specific tab (“**Button Xn**”) becomes available under “Buttons” in the tree on the left.

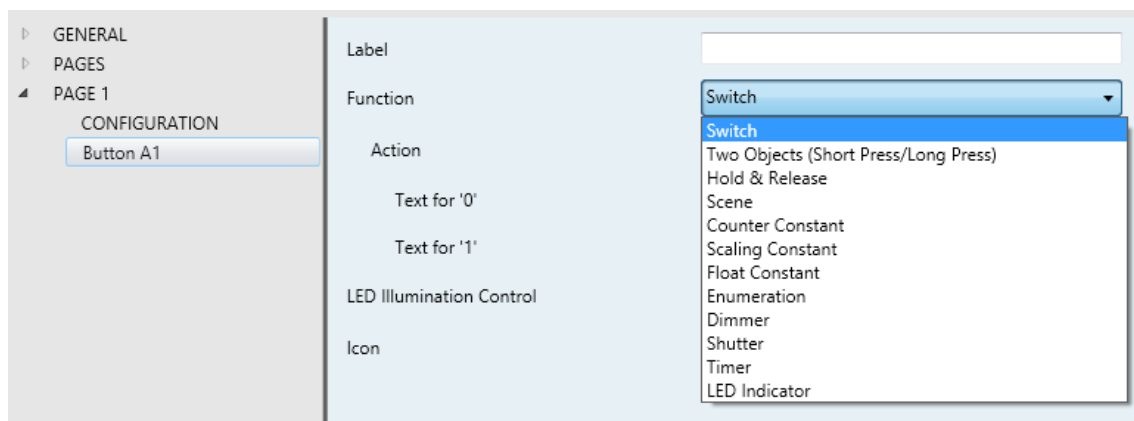


Figure 38 Individual Button.

Parameters common to all types of controls are:

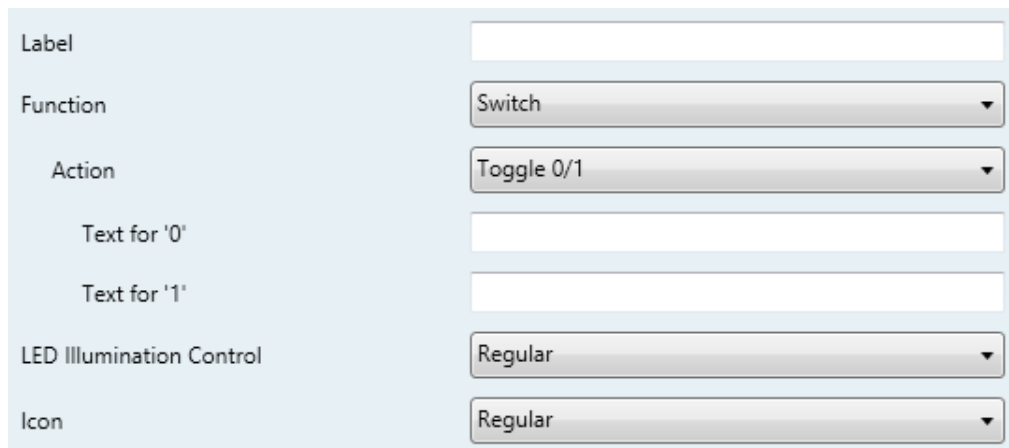
- **Function:** sets the desired function for the button: “Switch”, “Two Objects (short press / long press)”, “Hold & Release”, “Scene”, “Counter constant”,

“Scaling constant”, “Float constant”, “Enumeration”, “Dimmer”, “Shutter”, “Timer” or “LED Indicator”.

- **Label:** text (up to 10 characters) to be displayed to identify the control.
- **Icon:** icon to be displayed over each button in the display. For a list of the available icons, please refer to ANNEX II. Button Icons.

Depending on the function, some more parameters are involved (as described next). Please note that in the next pages “[Pn] [Xj]” is used as a general notation for the communication objects, where “Xj” depends on the button (A1, A2, B1, etc.) and “Pn” depends on the page number.

Switch



Label	<input type="text"/>
Function	Switch
Action	Toggle 0/1
Text for '0'	<input type="text"/>
Text for '1'	<input type="text"/>
LED Illumination Control	Regular
Icon	Regular

Figure 39 Individual Button - Switch.

- **Action:** sets the value to be sent to the bus (through object “[Pn] [Xj] Switch”) when the user touches the button. The options are “Send 0”, “Send 1” and “Toggle 0/1”.
- **Text for ‘0’ and Text for ‘1’:** sets the text to be displayed between the icons of the buttons when the values ‘0’ and ‘1’ are sent, respectively.
- **LED illumination control:** sets the behaviour of the LEDs on the buttons. The options are “Regular”, “State-dependent” and “Dedicated Object”.

In case of selecting the latter, the object “[Pn] [Xj] LED On/Off” will be included in the project topology (the desired values to turn on or turn off the LED are set by the parameter **Object Value**).

Two Objects (Short Press / Long Press)

Label	<input type="text"/>
Function	Two Objects (Short Press/Long Press) ▼
Action on Short Press	Send 0 ▼
Text for '0'	<input type="text"/>
Action on Long Press	Send 0 ▼
Long Press Threshold Time	6 <input type="button" value="↑"/> <input type="button" value="↓"/> ds
LED Illumination Control	State-dependent ▼
State Object	Short Press Object ▼
Icon	Regular ▼

Figure 40 Individual Button – Two Objects (short press / long press).

- **Action on Short press:** sets the value to be sent to the bus (through “[Pn] [Xj] Two switches - Short press”) when the user short-presses the button. The options are “Send 0”, “Send 1”, “Toggle 0/1” and “Send 1-byte Unsigned Int Value”. In case of selecting the latter, an additional parameter (“**Value**”) will be displayed to enter the desired one-byte value (0 - 255).
- **Text for ‘0’ and Text for ‘1’:** sets the text to be displayed between the icons of the buttons when the values ‘0’ and ‘1’ are sent, respectively.
- **Action on Long press:** sets the value to be sent to the bus (through “[Pn] [Xj] Two switches - Long press”) when the user long-presses the button. The options are the same as for the short press.
- **Long Press Threshold Time:** sets the minimum time the user should hold the button in order to consider it a long press. The available range is 5 to 50 tenths of a second, being 5 tenths the default value.
- **LED illumination control:** analogous to the homonymous parameter for the “LED Indicator”, “Switch” and “Hold & Release” functions (see previous pages). The options in this case are “Regular”, “State dependent” and “Dedicated object”.

Selecting “State dependent” brings an additional parameter, “**LED Illumination State Object**”, to make the state of the LED correspond either to object “[Pn] [Xj] **Two objects - Short press**” (“Short Press Object”) or to object “[Pn] [Xj] **Two objects - Long press**” (“Long Press Object”).

Hold & Release

Figure 41 Individual Button – Hold & Release.

- **Action on Hold:** sets the value to be sent to the bus (through “[Pn] [Xj] **Hold & Release**”) when the user touches the button. The options are “Send 0” and “Send 1” (default).
- **Action on Release:** sets the value to be sent to the bus (again, through “[Pn] [Xj] **Hold & Release**”) when the user stops touching the button. The options are “Send 0” (default) and “Send 1”.
- **LED illumination control:** analogous to the homonymous parameter for the “LED Indicator” and “Switch” functions (see above). The options in this case are “Regular” and “Dedicated object”.

Scene

- **Action:** sets whether the value to be sent to the KNX bus (through “[General] **Scene: send**”) when the user touches the button will always be a scene run request (“Run scene”) or –depending on the length of button press– a scene run or save request (“Run (Short Press) + Save (3s Press) Scene”).
- **Scene number:** number of the scene (1 – 64) to be sent to the bus, both in the case of the run requests and the save requests.

- **LED illumination control:** analogous to the homonymous parameter for all the above functions (see previous pages). The options in this case are “Regular”, and “Dedicated object”.

Label	<input type="text"/>
Function	Scene ▼
Action	Run Scene ▼
Scene Number	1 <input type="button" value="▲"/> <input type="button" value="▼"/>
LED Illumination Control	Regular ▼
Icon	Regular ▼

Figure 42 Individual Button – Scene.

Counter Constant / Scaling Constant / Float Constant

Label	<input type="text"/>
Function	Counter Constant ▼
Size	1 byte ▼
Signed	<input type="checkbox"/>
Object Value	0 <input type="button" value="▲"/> <input type="button" value="▼"/>
LED Illumination Control	State-dependent ▼
Icon	Regular ▼

Figure 43 Individual Button - Scaling Constant.

- **Object Value:** sets the value to be sent to the KNX bus when the user touches the button. The available range and the object through which the value is sent depends for each as the table below shows.

In case of selecting Counter Constant, two specific parameters (“**Size**” and “**Signed**”) will be displayed to respectively define the size of the constant (“1 byte” or “2 bytes”) and whether it is a signed value or an unsigned value. Depending on that, the range and the name of the object will vary.

	Available Values	Name of the Object
Scaling Constant	0% – 100%	[Pn] [Xj] 1-byte Scaling Value
Counter Constant	0 – 255	[Pn] [Xj] 1-byte Unsigned Int Value
	-128 – 127	[Pn] [Xj] 1-byte Signed Int Value
	0 – 65535	[Pn] [Xj] 2-byte Unsigned Int Value
	-32768 – 32767	[Pn] [Xj] 2-byte Signed Int Value
Float Constant	-671088.64 – 670760.96	[Pn] [Xj] 2-byte Float Value

Table 3 Constant – Object Value

- **LED illumination control:** analogous to the homonymous parameter for all the above functions (see previous pages). The options in this case are “Regular”, “State dependent” and “Dedicated object”.

Enumeration

Label	<input type="text"/>
Function	Enumeration ▼
# Enums	1 <input type="button" value="↑"/> <input type="button" value="↓"/>
Value 1	0 <input type="button" value="↑"/> <input type="button" value="↓"/>
Text 1	<input type="text"/>
LED Illumination Control	Regular ▼
Icon	Regular ▼

Figure 44 Individual Buttons – Enumeration.

This control is completely analogous to the Enumeration control in pair-button controls (see section 2.4.3), but without the parameter to link the control to the display (in this case, the value is sent to the bus when the button is pressed).

- **# Enums:** number of states (up to 6) that will be distinguished. For every distinguished state, one parameter **Value j** and **Text j** will become available, in order to set the value sent through the communication object.

- **Value j :** numerical value (0 – 255) that will be sent through the object “[Pn] [Xj] Enumeration” when the user, after pressing the buttons, sets the control box to state j .
- **Text j :** text to be displayed, between the icons of the buttons, to indicate the selected enumeration value.
- **LED illumination control:** analogous to the homonymous parameter for all the previous functions (see previous pages). The options in this case are “Regular” and “Dedicated object”.

Dimmer

Label	<input type="text"/>
Function	Dimmer ▼
Dimming Step	100% ▼
LED Illumination Control	State-dependent ▼
Icon	Regular ▼

Figure 45 Individual Button - Dimmer.

The (alternating) switch orders will be sent through the “[Pn] [Xj] Light On/Off” one-bit object, while the (alternating) increase/decrease/stop orders will be through the “[Pn] [Xj] Light Dimming” four-bit object.

On the other hand, the “[Pn] [Xj] Light Indicator” one-byte object may be linked to the light level status object of the dimmer (in fact, this object is only intended to receive values from the bus, not to send them). As explained in 2.4.4, the state-dependent LED lighting will be determined by the value of this object (LED off at 0% and on at any other level).

The parameters for this function are:

- **Step:** defines the dimming step (“100%”, “50%”, “25%”, “12.5%”, “6.25%”, “3.1%” or “1.5%”) to be sent (through “[Pn] [Xj] Light Dimming”) to the light dimmer with every long press.

Note: *since dimmers typically do not apply the new light level immediately (i.e., the step is performed progressively) and since TMD-Display View sends*

an order to interrupt the step dimming once the user releases the button, it is advisable to configure a step of 100%. This way, the user can perform any dimming step by simply leaving the button pressed and then releasing it, without needing to make successive button presses.

- **LED illumination control:** analogous to the homonymous parameter for all the above functions (see previous pages). The options in this case are “Regular”, “State dependent” and “Dedicated object”.

Shutter

Label	<input type="text"/>
Function	Shutter
Type	Standard
LED Illumination Control	Regular
Icon	Regular

Figure 46 Individual Button - Shutter.

The (alternating) move up/down orders will be sent through the “[Pn] [Xj] **Move Shutter (Toggle Direction)**” one-bit object, while the (alternating) step up/down orders will be through the “[Pn] [Xj] **Stop Shutter/Step**” one-bit object.

Additionally, a one-byte object (“[Pn] [Xj] **Shutter Position**”) is provided to link it to the position status object of the shutter actuator (in fact, this object is only intended to receive values from the bus, not to send them). As explained in □ the alternation of the move up/down and the step up/down requests will be conditioned to these statuses, to prevent sending ineffective requests.

The parameters for this function are:

- **Type:** sets the desired control type: “Standard” or “Hold & Release”, which have been described in section 2.4.4.
- **LED illumination control:** analogous to the homonymous parameter for all the previous functions (see previous pages). The options in this case are “Regular” and “Dedicated object”.

Timer

Label	<input type="text"/>
Function	Timer
Action	Send 0
Timer type	Without Repetition
Icon	Regular

Figure 47 Individual Buttons – Timer.

- **Action:** sets the value to send to the bus when the configured time elapses. The options are “Send 0”, “Send 1” and “Send Scene”. The binary values are sent through the 1-bit object “[Pn] [Xj] **Timer Control: “0/1”**”, but the scene values through the generic scene object “[Pn] [Xj] **Scene: Receive**”.
- **Timer type:** sets the timer type.
 - “Without Repetition”: The parameterised value ("0", "1" or a scene) will be sent once through the corresponding object.
 - “Periodic” The parameterised value ("0", "1" or a scene) will be sent daily through the corresponding object.
 - “Countdown” The parameterised value ("0", "1" or a scene) will be sent through the corresponding object once the countdown expires.

The values are sent when the user **enables** the timer and sets when the sending must take place. This can be done by short pressing the corresponding button (each press switches the state: disabled/enabled) or by sending the value "1" through the “[Px][By] **Enable Timer**” object. If the timer has been enabled, the timer label and the trigger time will flash on the screen. Besides, the LED of the button will stay on. If the timer is disabled, just the name of the button will be shown, while the associated LED stays off. The same applies to the countdown timers, although the time is shown preceded by the "-" sign, thus indicating that it is a decreasing value.

Once enabled, it is necessary to set the time (or the countdown interval) for the sending. This is done from the device itself by long-pressing on the button

(in the case of the countdown, the button can also be short-pressed). Then a virtual clock shows up in order to set the desired time by making use of the arrow buttons. To confirm the selection, just short press on the central area.

LED Indicator

Label	<input type="text"/>
Function	LED Indicator ▼
LED Illumination Control Object Value	0 = Off; 1 = On ▼

Figure 48 Individual Buttons – Timer.

- **LED Illumination Control Object Value:** sets the value to switch off and switch on the LED. The options are: “0 = Off; 1 = On” or “0 = On; 1 = Off”.

2.5 CONFIGURATION PAGES

When marking the checkbox corresponding to one of the configuration pages in the Configuration screen under the Pages tab, a new tab is added on the menu on the left.

2.5.1 GENERAL CONFIGURATION

From this ETS window it is possible to enable and to assign a name to certain parameters of TMD-Display View adjustable by the end user at runtime, from the Configuration page.

ETS PARAMETERISATION

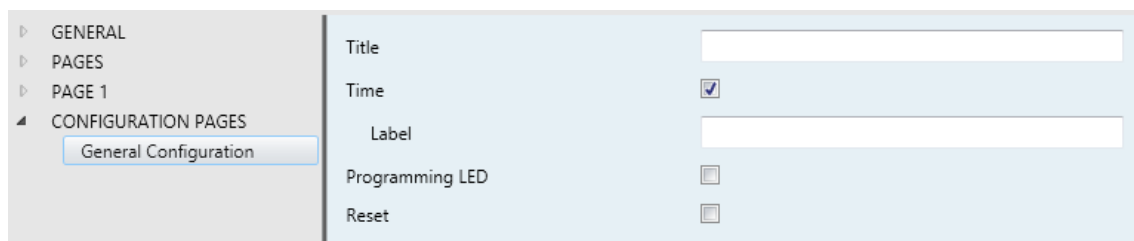


Figure 49 Configuration pages – General Configuration.

- **Title:** a text string of up to 10 characters that identifies the General Configuration page and that will be shown in the upper left side of the display when accessing this page.
- **Label:** textbox to enter the string (up to 10 characters) that is displayed on the screen next to the button, in order to identify the control.
- **Time:** allows setting the time of the clock (with format "hh:mm"). To do so, please press on the corresponding button and make use of the arrow buttons (long presses cause ten-minute steps, while short presses imply one-minute steps). To confirm, just press on the central area between the buttons.

Note: bus power failures may introduce delays, as the internal clock needs power to work. In such case, the clock should be set to the correct time once the bus power returns.

- **Programming LED:** when the physical address of the device needs to be downloaded, it is possible to enter the programming mode directly by short-pressing this button. This feature overcomes the need for accessing the rear

side of the panel to physically press the programming button. While the programming mode is active, the LED associated to this button in the Configuration page will stay on. To leave the programming mode, just short-press this button again (the programming LED and the button LED will be switched off).

- **Reset:** a long press of at least 3 seconds reset all the stored values and set TMD-Display View to the initial state it had after the last download performed.

2.5.2 DISPLAY

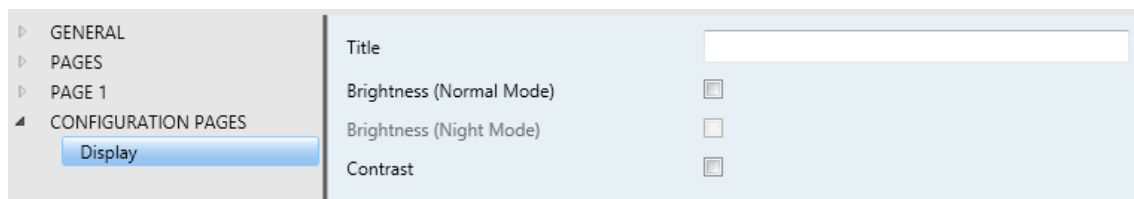


Figure 50 Configuration pages – Display.

- **Title:** a text string of up to 10 characters that identifies the Display page and that will be shown in the left upper side of the display when accessing this page.
- **Label:** textbox to enter the string (up to 10 characters) that is displayed on the screen next to the button, in order to identify the control.
- **Brightness (Normal Mode):** allows setting the display brightness level for the normal mode (values from 1 to 10). Press the button corresponding to the brightness and simply make use of the arrow buttons to modify this level.
- **Brightness (Night Mode):** analogous to the above one, but referred to the night mode. This parameter is only available after enabling the night mode in the LEDs and Display tab (see section 2.2.4).
- **Contrast:** allows setting the display contrast level (values from 1 to 10). Press the button corresponding to the Contrast and simply make use of the arrow buttons to modify this level.

2.6 INPUTS

TMD-Display View incorporates **two analogue/digital inputs**, each configurable as a:

- **Binary Input**, for the connection of a pushbutton or a switch/sensor.
- **Temperature Probe**, for the connection of a temperature sensor (such as models ZN1AC-NTC68 S/E/F and SQ-AmbienT from Zennio).
- **Motion Detector**, for the connection of a motion detector (models ZN1IO-DETEC-P and ZN1IO-DETEC-X from Zennio).

Important: older models of the Zennio motion detector (e.g., ZN1IO-DETEC and ZN1IO-DETEC-N) will not work properly with TMD-Display View.

ETS PARAMETERISATION

When **Inputs** has been activated in the General parameters screen, the following drop-lists will be available for the selection of the specific functions required.

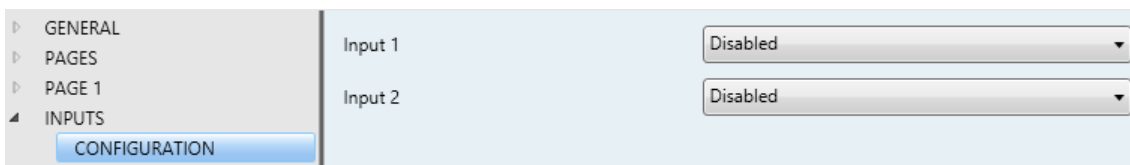


Figure 51. Inputs - Configuration

All inputs are disabled by default. Depending on the function selected for each input, additional tabs will be included in the menu on the left.

2.6.1 BINARY INPUT

Please refer to the specific user manual “**Binary Inputs**”, available in the TMD-Display View section, at the Zennio website (www.zennio.com).

2.6.2 TEMPERATURE PROBE

Please refer to the specific user manual “**Temperature Probe**”, available in the TMD-Display View section, at the Zennio website (www.zennio.com).

2.6.3 MOTION DETECTOR

It is possible to connect motion detectors (models **ZN1IO-DETEC-P** and **ZN1IO-DETEC-X** from Zennio) to the input ports of TMD-Display View.

Please refer to the specific user manual “**Motion Detector**”, available in the TMD-Display View section, at the Zennio website (www.zennio.com).

Notes:

- *This device **does not include luminosity measurement capabilities** (only motion detection). Therefore, the two models mentioned offer equivalent functionality when connected to this device*
- Motion detectors with references ZN1IO-DETEC and ZN1IO-DETEC-N are **not compatible** with TMD-Display View (may report inaccurate measurements if connected to this device).
- When connected to TMD-Display View, the rear micro-switch of model ZN1IO-DETEC-P should be set to position “**Type B**”.

2.7 THERMOSTAT

TMD-Display View implements **one Zennio thermostat** which can be enabled and fully customised.

For details please refer to the specific manual “**Zennio Thermostat**” available in the TMD-Display View section at the Zennio homepage (www.zennio.com).

ANNEX I: LED ILLUMINATION MODES

The central LED indicator of every button, by default (in most functions), will turn on for a brief instant whenever the button is touched. This behaviour is referred to as the “**Regular Illumination**”.

However, in most cases it is possible to assign different behaviours to the LEDs. Which options are available will depend on the function parameterised for the button, but will always include some of the following:

- **Regular Illumination:** the LED will light for an instant once the button is touched.
- **State-Dependent Illumination:** the LED will or will not light, depending on the value of the communication object that corresponds to the function implemented by the button. The exact correspondence between the different values of the object and the different states of the LED may be slightly different from one type of control to another, and is detailed for each function.
- **State-Dependent Illumination (both LEDs):** only applies to buttons configured as pair controls. The two LEDs of the control will light or not, depending on the value of the related object and on the particular control type parameterised for that pair of buttons. The only difference compared to the previous case is that, under “both LEDs”, the two LEDs will always turn off or on simultaneously, as if it were a unique indicator consisting of two LEDs.
- **Dedicated Object:** the LED will light or not depending on the value (“0” or “1”, configurable) of a binary, independent object. In the case of the pair controls, the value “0” will make one of the LEDs light (leaving the other one off), while the value “1” will make them switch their states.

Note:

Regarding the LEDs, it is interesting to distinguish the following cases:

- Disabled button: the LED will remain off, and the button will have no function.
- Button configured as “Individual” with “LED Indicator” function: the button will still have no function. The LED may be turned on/off through a binary object.

- Button configured as any other control type: the behaviour of the LED will be configurable according to the following table (being also possible to leave it turned off).

		Regular	State-dep.	State-dep. (both LEDs)	Dedicated object
PAIR	Switch	✓	✓	✓	✓
	Switch + Indicator	✓	✓	✓	✓
	Two Objects	✓	✓	✓	✓
	Counter	✓	-	-	✓
	Scaling	✓	-	-	✓
	Float	✓	-	-	✓
	Enumeration	✓	-	-	✓
	Shutter	✓	-	-	✓
	Dimmer	✓	✓	✓	✓
	[Climate] Temperature	✓			✓
	[Climate] Mode	✓	✓	✓	✓
	[Climate] Fan	✓	✓	✓	✓
	Multimedia	-	-	-	-
	Indicator	-	-	-	-
INDIVIDUAL	Binary	✓	✓	-	✓
	Two Objects	✓	✓	-	✓
	Hold & Rel.	✓	-	-	✓
	Scene	✓	-	-	✓
	Constants	✓	✓	-	✓
	Enumeration	✓	-	-	✓
	Dimmer	✓	✓	-	✓
	Shutter	✓	-	-	✓
	Timer	-	-	-	-
	LED Indicator	-	-	-	✓

Table 4 Functions vs. LED Illumination Options.

Although the behaviour of the LEDs can be configured independently for each control (see section 2.3.1), it is also possible to define a **general behaviour for all of them** thus not being then necessary to configure the same option multiple times. In case of opting for a general configuration, the options are:














- **Regular Illumination.**
- **State-Dependent (when available).** Functions where “state-dependent” is not available will use the regular illumination.
- **State-Dependent (when available) (both LEDs).** Functions where “state-dependent” is not available will use the regular illumination.
- **Dedicated Object.** One binary communication object per control will be included in the project topology so that the LED of every control turns on/off depending on its own object.

ETS PARAMETERISATION








For details on the parameterisation of the LED illumination modes please refer to the pages that cover the specific function being assigned to the button (section 2.4).

ANNEX II. BUTTON ICONS

Icons for individual buttons:

Name	Icon
Default	
Off	
On	
0	
1	
Minus	
Plus	
Down	
Up	
Left	
Right	
Stop	
Play	

Icons for pair buttons: they are similar to those for the individual buttons; however they are grouped into pairs. The one on the left will be shown on the left button of the pair, while the one on the right will be drawn on the right button.

Name	Icon
Default	
Off/On	
0/1	
Minus/Plus	
Down/Up	
Left/Right	
Stop/Play	

ANNEX II: COMMUNICATION OBJECTS

- “**Functional range**” shows the values that, with independence of any other values permitted by the bus according to the object size, may be of any use or have a particular meaning because of the specifications or restrictions from both the KNX standard or the application programme itself.

Number	Size	I/O	Flags	Data type (DPT)	Functional Range	Name	Function
1	3 Bytes	I	C - - W -	DPT_TimeOfDay	00:00:00 - 23:59:59	[General] Time of Day	Time of Day External Reference
2	1 Byte	I	C - - W -	DPT_SceneNumber	0-63	[General] Scene: Receive	0-63 (Run Scene 1-64)
3	1 Byte		C T - - -	DPT_SceneControl	0-63; 128-191	[General] Scene: Send	0-63 / 128-191 (Run/Save Scene 1-64)
4	2 Bytes	I	C - - W -	DPT_Value_Temp	-273,00 - 670760,00	[General] Screensaver - External Temperature	Temperature to be shown on the screensaver
5	1 Bit	I	C - - W -	DPT_Switch	0/1	[General] LEDs & Disp - Brightness Mode	0 = Normal; 1 = Night
	1 Bit	I	C - - W -	DPT_Switch	0/1	[General] LEDs & Disp - Brightness Mode	0 = Night; 1 = Normal
6	1 Byte	I	C - - W -	DPT_SceneNumber	0-10	[General] Display - Contrast	0 = Level 1; ... ; 10 (Scene Format)
7	1 Bit	I	C - - W -	DPT_Enable	0/1	[General] Sounds - Disabling Button Sound	0 = Disable Sound; 1 = Enable Sound
	1 Bit	I	C - - W -	DPT_Enable	0/1	[General] Sounds - Disabling Button Sound	0 = Enable Sound; 1 = Disable Sound
8	1 Bit	I	C - - W -	DPT_Trigger	0/1	[General] Sounds - Doorbell	0 = No Action; 1 = Doorbell
	1 Bit	I	C - - W -	DPT_Trigger	0/1	[General] Sounds - Doorbell	0 = Doorbell; 1 = No Action
9	1 Bit	I	C - - W -	DPT_Alarm	0/1	[General] Sounds - Alarm	0 = Stop Alarm; 1 = Start Alarm
	1 Bit	I	C - - W -	DPT_Alarm	0/1	[General] Sounds - Alarm	0 = Start Alarm; 1 = Stop Alarm
10	1 Bit	I	C - - W -	DPT_Enable	0/1	[General] Touch Locking	0 = Unlock; 1 = Lock
	1 Bit	I	C - - W -	DPT_Enable	0/1	[General] Touch Locking	0 = Lock; 1 = Unlock
11	1 Bit	O	C T R - -	DPT_Switch	0/1	[General] Welcome Back Object: "1"	1-bit Generic Control
	1 Bit	O	C T R - -	DPT_Switch	0/1	[General] Welcome Back Object: "0"	1-bit Generic Control
12	1 Bit	I	C - - W -	DPT_Bool	0/1	[General] Welcome back - Additional Condition	Additional Condition Object 1
13	1 Bit	I	C - - W -	DPT_Bool	0/1	[General] Welcome back - Additional Condition	Additional Condition Object 2
14	1 Bit	I	C - - W -	DPT_Bool	0/1	[General] Welcome back - Additional Condition	Additional Condition Object 3
15	1 Bit	I	C - - W -	DPT_Bool	0/1	[General] Welcome back - Additional Condition	Additional Condition Object 4
16	1 Bit	I	C - - W -	DPT_Bool	0/1	[General] Welcome back - Additional Condition	Additional Condition Object 5
17, 30, 43, 56, 69, 82, 95, 108, 121, 134, 147, 160, 173, 186, 199, 212	1 Bit	I/O	C T R W -	DPT_Switch	0/1	[Pn] [X1] Switch: "0"	1-bit Generic Control
	1 Byte	I/O	C T R W -	DPT_Value_1_Ucount	0 - 255	[Pn] [X1] Two Objects - Short Press: (1-byte)	Send selected 1-byte value on short press
	1 Bit	I/O	C T R W -	DPT_Switch	0/1	[Pn] [X1] Switch: "1"	1-bit Generic Control
	1 Bit	I/O	C T R W -	DPT_Switch	0/1	[Pn] [X1] Switch: "0/1"	1-bit Generic Control
	1 Bit	O	C T R - -	DPT_Switch	0/1	[Pn] [X1] Hold & Release	1-bit Generic Control
	1 Bit	O	C T R - -	DPT_Switch	0/1	[Pn] [X] Light On/Off	0 = Off; 1 = On

1 Bit	O	CTR--	DPT_Switch	0/1	[Pn] [X1] Light On/Off (Toggle)	0 = Off; 1 = On
1 Bit	O	CTR--	DPT_Step	0/1	[Pn] [X] Stop Shutter/Step	0 = Stop Shutter/Step Up; 1 = Stop Shutter/Step Down
1 Bit	O	CTR--	DPT_Step	0/1	[Pn] [X1] Stop Shutter/Step	0 = Stop Shutter/Step Up; 1 = Stop Shutter/Step Down
1 Bit	I/O	CTRW-	DPT_Switch	0/1	[Pn] [X] Two Objects - Short Press	1-bit Generic Control
1 Bit	I/O	CTRW-	DPT_Switch	0/1	[Pn] [X1] Two Objects - Short Press: "0"	1-bit Generic Control
1 Bit	I/O	CTRW-	DPT_Switch	0/1	[Pn] [X1] Two Objects - Short Press: "1"	1-bit Generic Control
1 Bit	I/O	CTRW-	DPT_Switch	0/1	[Pn] [X1] Two Objects - Short Press: "0/1"	1-bit Generic Control
1 Bit	I/O	CTRW-	DPT_Switch	0/1	[Pn] [X] Switch	1-bit Generic Control
1 Bit	O	CTR--	DPT_Trigger	0/1	[Pn] [X] Stop Shutter	0, 1 = Stop Shutter
1 Bit	O	CTR--	DPT_Trigger	0/1	[Pn] [X1] Stop Shutter	0, 1 = Stop Shutter
1 Bit	I/O	CTRW-	DPT_Heat_Cool	0/1	[Pn] [X] (Climate) Mode	Left = Cool; Right = Heat
1 Byte	I/O	CTRW-	DPT_HVACContrMode	0 = Auto 1 = Calor 3 = Frio 9 = Viento 14 = Seco	[Pn] [X] (Climate) Mode - Extended	Rotating change of selected modes on short press
1 Bit	O	CTR--	DPT_Start	0/1	[Pn] [X] 1-bit Multimedia Control	1 = Start/Forward; 0 = Stop/Backward
1 Bit	I/O	CTRW-	DPT_Enable	0/1	[Pn] [X1] Enable Timer	0 = Disabled; 1 = Enabled
1 Bit	O	CTR--	DPT_Step	0/1	[Pn] [X] Fan Control (Type: 1-bit)	0 = Decrease; 1 = Increase
1 Byte	O	CTR--	DPT_Scaling	0% - 100%	[Pn] [X] Fan Control (Type: Scaling)	0%, 33%, 67%, 100%
1 Byte	O	CTR--	DPT_Scaling	0% - 100%	[Pn] [X] Fan Control (Type: Scaling)	0%, 50%, 100%
1 Byte	O	CTR--	DPT_Scaling	0% - 100%	[Pn] [X] Fan Control (Type: Scaling)	Auto, 50%, 100%
1 Byte	O	CTR--	DPT_Scaling	0% - 100%	[Pn] [X] Fan Control (Type: Scaling)	50%, 100%
1 Byte	O	CTR--	DPT_Scaling	0% - 100%	[Pn] [X] Fan Control (Type: Scaling)	33%, 67%, 100%
1 Byte	O	CTR--	DPT_Scaling	0% - 100%	[Pn] [X] Fan Control (Type: Scaling)	Auto, 33%, 67%, 100%
1 Byte	O	CTR--	DPT_Scaling	0% - 100%	[Pn] [X] Fan Control (Type: Scaling)	25%, 50%, 75%, 100%
1 Byte	O	CTR--	DPT_Scaling	0% - 100%	[Pn] [X] Fan Control (Type: Scaling)	0%, 25%, 50%, 75%, 100%
1 Byte	O	CTR--	DPT_Scaling	0% - 100%	[Pn] [X] Fan Control (Type: Scaling)	Auto, 25%, 50%, 75%, 100%
1 Byte	O	CTR--	DPT_Scaling	0% - 100%	[Pn] [X] Fan Control (Type: Scaling)	100%
1 Byte	O	CTR--	DPT_Scaling	0% - 100%	[Pn] [X] Fan Control (Type: Scaling)	Auto, 20%, 40%, 60%, 80%, 100%
1 Byte	O	CTR--	DPT_Scaling	0% - 100%	[Pn] [X] Fan Control (Type: Scaling)	0%, 20%, 40%, 60%, 80%, 100%
1 Byte	O	CTR--	DPT_Scaling	0% - 100%	[Pn] [X] Fan Control (Type: Scaling)	20%, 40%, 60%, 80%, 100%
1 Byte	O	CTR--	DPT_Value_1_Ucount	0 - 255	[Pn] [X] Fan Control (Type: Enumeration)	Auto, 1
1 Byte	O	CTR--	DPT_Value_1_Ucount	0 - 255	[Pn] [X] Fan Control (Type: Enumeration)	1, 2
1 Byte	O	CTR--	DPT_Value_1_Ucount	0 - 255	[Pn] [X] Fan Control (Type: Enumeration)	1
1 Byte	O	CTR--	DPT_Value_1_Ucount	0 - 255	[Pn] [X] Fan Control (Type: Enumeration)	0, 1
1 Byte	O	CTR--	DPT_Scaling	0% - 100%	[Pn] [X] Fan Control (Type: Scaling)	Auto, 100%
1 Byte	O	CTR--	DPT_Value_1_Ucount	0 - 255	[Pn] [X] Fan Control (Type: Enumeration)	0, 1, 2
1 Byte	O	CTR--	DPT_Value_1_Ucount	0 - 255	[Pn] [X] Fan Control (Type: Enumeration)	Auto, 1, 2

	1 Byte	O	CTR--	DPT_Value_1_Ucount	0 - 255	[Pn] [X] Fan Control (Type: Enumeration)	Auto, 1, 2, 3
	1 Byte	O	CTR--	DPT_Value_1_Ucount	0 - 255	[Pn] [X] Fan Control (Type: Enumeration)	0, 1, 2, 3, 4, 5
	1 Byte	O	CTR--	DPT_Value_1_Ucount	0 - 255	[Pn] [X] Fan Control (Type: Enumeration)	1, 2, 3, 4, 5
	1 Byte	O	CTR--	DPT_Value_1_Ucount	0 - 255	[Pn] [X] Fan Control (Type: Enumeration)	Auto, 1, 2, 3, 4, 5
	1 Byte	O	CTR--	DPT_Value_1_Ucount	0 - 255	[Pn] [X] Fan Control (Type: Enumeration)	Auto, 1, 2, 3, 4
	1 Byte	O	CTR--	DPT_Value_1_Ucount	0 - 255	[Pn] [X] Fan Control (Type: Enumeration)	0, 1, 2, 3, 4
	1 Byte	O	CTR--	DPT_Value_1_Ucount	0 - 255	[Pn] [X] Fan Control (Type: Enumeration)	1, 2, 3, 4
	1 Byte	O	CTR--	DPT_Value_1_Ucount	0 - 255	[Pn] [X] Fan Control (Type: Enumeration)	1, 2, 3
	1 Byte	O	CTR--	DPT_Value_1_Ucount	0 - 255	[Pn] [X] Fan Control (Type: Enumeration)	0, 1, 2, 3
	1 Byte	O	CTR--	DPT_Scaling	0% - 100%	[Pn] [X] Fan Control (Type: Scaling)	0%, 100%
18, 31, 44, 57, 70, 83, 96, 109, 122, 135, 148, 161, 174, 187, 200, 213	1 Byte	I	C--W-	DPT_Scaling	0% - 100%	[Pn] [X] Fan Indicator	0 - 100%
	1 Byte	I	C--W-	DPT_Value_1_Ucount	0 - 255	[Pn] [X] Fan Indicator	Enumerated Value
	1 Bit	I	C--W-	DPT_Switch	0/1	[Pn] [X] [Status] Switch	Value of 1-bit Generic Control
	1 Bit	I	C--W-	DPT_Switch	0/1	[Pn] [X] [Status] Two Objects - Short Press	Value of 1-bit Generic Control
	1 Bit	I	C--W-	DPT_Heat_Cool	0/1	[Pn] [X] [Status] (Climate) Mode	Left = Cool; Right = Heat
	1 Byte	I	C--W-	DPT_HVACContrMode	0 = Auto 1 = Calor 3 = Frio 9 = Viento 14 = Seco	[Pn] [X] [Status] (Climate) Mode - Extended	Rotating change of selected modes on short press
	1 Bit	I	C--W-	DPT_Switch	0/1	[Pn] [X1] [Status] Switch	Value of 1-bit Generic Control
	1 Bit	I	C--W-	DPT_Switch	0/1	[Pn] [X1] [Status] Hold & Release	Value of 1-bit Generic Control
	1 Bit	I	C--W-	DPT_BinaryValue	0/1	[Pn] [X1] [Status] Two Objects - Short Press: (1-bit)	Value of short press object: (1-bit)
	1 Byte	I	C--W-	DPT_Value_1_Ucount	0 - 255	[Pn] [X1] [Status] Two Objects - Short Press: (1-byte)	Value of short press object: (1-byte)
	1 Bit	I	C--W-	DPT_Step	0/1	[Pn] [X1] [Status] Stop Shutter/Step	0 = Stop Shutter/Step Up; 1 = Stop Shutter/Step Down
	1 Bit	I	C--W-	DPT_Trigger	0/1	[Pn] [X1] [Status] Stop Shutter	0, 1 = Stop Shutter
	1 Bit	I	C--W-	DPT_Switch	0/1	[Pn] [X] [Status] Switch	1-bit Generic Control
	1 Bit	I	C--W-	DPT_Start	0/1	[Pn] [X] [Status] 1-bit Multimedia Control	1 = Start/Forward; 0 = Stop/Backward
19, 32, 45, 58, 71, 84, 97, 110, 123, 136, 149, 162, 175, 188, 201, 214	4 Bit	O	CTR--	DPT_Control_Dimming	0x0 (Detener) 0x1 (Reducir 100%) ... 0x7 (Reducir 1%) 0x8 (Detener) 0x9 (Subir 100%) ... 0xF (Subir 1%)	[Pn] [X] Light Dimming	4-bit Dimming Control
	4 Bit	O	CTR--	DPT_Control_Dimming	0x0 (Detener) 0x1 (Reducir 100%) ... 0x7 (Reducir 1%) 0x8 (Detener) 0x9 (Subir 100%)	[Pn] [X1] Light Dimming (Toggle)	4-bit Dimming Control

					... 0xF (Subir 1%)		
20, 33, 46, 59, 72, 85, 98, 111, 124, 137, 150, 163, 176, 189, 202, 215	1 Byte	I	C - - W -	DPT_Scaling	0% - 100%	[Pn] [X] Light Indicator	0 - 100%
	1 Byte	I	C - - W -	DPT_Scaling	0% - 100%	[Pn] [X1] Light Indicator	0 - 100%
	1 Byte	I	C - - W -	DPT_Scaling	0% - 100%	[Pn] [X1] Shutter Position	0 - 100%
	2 Bytes	I	C - - W -	DPT_Value_2_Ucount	0 - 65535	[Pn] [X] 2-byte Unsigned Int Indicator	0 - 65535
	2 Bytes	I	C - - W -	DPT_Value_2_Count	-32768 - 32767	[Pn] [X] 2-byte Signed Int Indicator	-32768 - 32767
	1 Byte	I	C - - W -	DPT_Value_1_Ucount	0 - 255	[Pn] [X] 1-byte Unsigned Int Indicator	0 - 255
	1 Byte	I	C - - W -	DPT_Value_1_Count	-128 - 127	[Pn] [X] 1-byte Signed Int Indicator	-128 - 127
	1 Byte	I	C - - W -	DPT_Scaling	0% - 100%	[Pn] [X] 1-byte Indicator (Scaling)	0 - 100%
	2 Bytes	I	C - - W -	DPT_Value_Temp	-273,00 - 670760,00	[Pn] [X] Temperature Indicator	-273.00°C - 670760.96°C
	4 Bytes	I	C - - W -	13.xxx		[Pn] [X] 4-byte Signed Int Indicator	-2147483648 - 2147483647
	2 Bytes	I	C - - W -	9.xxx	-671088.64 - 670760.96	[Pn] [X] 2-byte Float Indicator	-671088.64 - 670760.96
	1 Bit	I	C - - W -	DPT_Switch	0/1	[Pn] [X] Binary Indicator	1-bit Generic Indicator
	1 Byte	I	C - - W -	DPT_Value_1_Ucount	0 - 255	[Pn] [X] Enumerated Indicator	1-byte Enumeration
	1 Byte	I	C - - W -	DPT_Scaling	0% - 100%	[Pn] [X] Shutter Position	0 - 100%
	1 Bit	I	C - - W -	DPT_Switch	0/1	[Pn] [X] [Status] Two Objects - Long Press	Value of 1-bit Generic Control
	1 Byte	I	C - - W -	DPT_Scaling	0% - 100%	[Pn] [X] [Status] 1-byte Scaling Value	0 - 100%
	1 Byte	I	C - - W -	DPT_Value_1_Ucount	0 - 255	[Pn] [X] [Status] 1-byte Unsigned Int Value	0 - 255
	1 Byte	I	C - - W -	DPT_Value_1_Count	-128 - 127	[Pn] [X] [Status] 1-byte Signed Int Value	-128 - 127
	2 Bytes	I	C - - W -	DPT_Value_2_Count	-32768 - 32767	[Pn] [X] [Status] 2-byte Signed Int Value	-32768 - 32767
	2 Bytes	I	C - - W -	DPT_Value_2_Ucount	0 - 65535	[Pn] [X] [Status] 2-byte Unsigned Int Value	0 - 65535
	1 Byte	I	C - - W -	DPT_Value_1_Ucount	0 - 255	[Pn] [X] [Status] Enumeration	Rotating value change
	2 Bytes	I	C - - W -	9.xxx	-671088.64 - 670760.96	[Pn] [X] [Status] 2-byte Float Value	-671088.64 - 670760.96
	2 Bytes	I	C - - W -	DPT_Value_Temp	-273,00 - 670760,00	[Pn] [X] [Status] (Climate) Temperature Setpoint	-273.00°C - 670760.96°C
	1 Bit	I	C - - W -	DPT_Switch	0/1	[Pn] [X] [Status] Fan Control - Auto Mode	Auto Mode Status
	1 Bit	I	C - - W -	DPT_BinaryValue	0/1	[Pn] [X1] [Status] Two Objects - Long Press: (1-bit)	Value of long press object: (1-bit)
	1 Byte	I	C - - W -	DPT_Value_1_Ucount	0 - 255	[Pn] [X1] [Status] Two Objects - Long Press: (1-byte)	Value of long press object: (1-byte)
	1 Byte	I	C - - W -	DPT_Scaling	0% - 100%	[Pn] [X1] [Status] 1-byte Scaling Value	0 - 100%
	1 Byte	I	C - - W -	DPT_Value_1_Ucount	0 - 255	[Pn] [X1] [Status] 1-byte Unsigned Int Value	0 - 255
	1 Byte	I	C - - W -	DPT_Value_1_Count	-128 - 127	[Pn] [X1] [Status] 1-byte Signed Int Value	-128 - 127
	2 Bytes	I	C - - W -	DPT_Value_2_Ucount	0 - 65535	[Pn] [X1] [Status] 2-byte Unsigned Int Value	0 - 65535
	2 Bytes	I	C - - W -	DPT_Value_2_Count	-32768 - 32767	[Pn] [X1] [Status] 2-byte Signed Int Value	-32768 - 32767
	2 Bytes	I	C - - W -	9.xxx	-671088.64 - 670760.96	[Pn] [X1] [Status] 2-byte Float Value	-671088.64 - 670760.96
	1 Byte	I	C - - W -	DPT_Value_1_Ucount	0 - 255	[Pn] [X1] [Status] Enumeration	Rotating value change
	1 Bit	I	C - - W -	DPT_Switch	0/1	[Pn] [X1] [Status] Timer Control	1-bit Generic Control
21, 34, 47, 60, 73, 86, 99, 112,	1 Bit	I/O	C T R W -	DPT_UpDown	0/1	[Pn] [X1] Move Shutter (Toggle Direction)	0 = Up; 1 = Down
	1 Bit	I/O	C T R W -	DPT_UpDown	0/1	[Pn] [X] Move Shutter	0 = Up; 1 = Down

125, 138, 151, 164, 177, 190, 203, 216	2 Bytes	I/O	CTRW -	DPT_Value_2_Ucount	0 - 65535	[Pn] [X1] 2-byte Unsigned Int Value	0 – 65535
	1 Byte	I/O	CTRW -	DPT_Value_1_Ucount	0 - 255	[Pn] [X1] 1-byte Unsigned Int Value	0 – 255
	1 Byte	I/O	CTRW -	DPT_Scaling	0% - 100%	[Pn] [X1] 1-byte Scaling Value	0 - 100%
	2 Bytes	I/O	CTRW -	9.xxx	-671088.64 - 670760.96	[Pn] [X1] 2-byte Float Value	-671088.64 – 670760.96
	1 Byte	I/O	CTRW -	DPT_Value_1_Count	-128 - 127	[Pn] [X1] 1-byte Signed Int Value	-128 - 127
	2 Bytes	I/O	CTRW -	DPT_Value_2_Count	-32768 - 32767	[Pn] [X1] 2-byte Signed Int Value	-32768 - 32767
	1 Bit	I/O	CTRW -	DPT_Switch	0/1	[Pn] [X] Two Objects - Long Press	1-bit Generic Control
	1 Bit	I/O	CTRW -	DPT_Switch	0/1	[Pn] [X1] Two Objects - Long Press: "0"	1-bit Generic Control
	1 Bit	I/O	CTRW -	DPT_Switch	0/1	[Pn] [X1] Two Objects - Long Press: "1"	1-bit Generic Control
	1 Bit	I/O	CTRW -	DPT_Switch	0/1	[Pn] [X1] Two Objects - Long Press: "0/1"	1-bit Generic Control
	2 Bytes	I/O	CTRW -	DPT_Value_2_Count	-32768 - 32767	[Pn] [X] 2-byte Signed Int Value	-32768 - 32767
	1 Byte	I/O	CTRW -	DPT_Value_1_Count	-128 - 127	[Pn] [X] 1-byte Signed Int Value	-128 - 127
	2 Bytes	I/O	CTRW -	9.xxx	-671088.64 - 670760.96	[Pn] [X] 2-byte Float Value	-671088.64 – 670760.96
	1 Byte	I/O	CTRW -	DPT_Scaling	0% - 100%	[Pn] [X] 1-byte Scaling Value	0 - 100%
	1 Byte	I/O	CTRW -	DPT_Value_1_Ucount	0 - 255	[Pn] [X] 1-byte Unsigned Int Value	0 – 255
	2 Bytes	I/O	CTRW -	DPT_Value_2_Ucount	0 - 65535	[Pn] [X] 2-byte Unsigned Int Value	0 – 65535
	1 Byte	I/O	CTRW -	DPT_Value_1_Ucount	0 - 255	[Pn] [X1] Enumeration	Rotating value change
	1 Byte	I/O	CTRW -	DPT_Value_1_Ucount	0 - 255	[Pn] [X] Enumeration	Rotating value change
	2 Bytes	I/O	CTRW -	DPT_Value_Temp	-273,00 - 670760,00	[Pn] [X] (Climate) Temperature Setpoint	-273.00°C – 670760.96°C
	1 Bit	I/O	CTRW -	DPT_Switch	0/1	[Pn] [X1] Timer Control: "0"	1-bit Generic Control
	1 Bit	I/O	CTRW -	DPT_Switch	0/1	[Pn] [X1] Timer Control: "1"	1-bit Generic Control
	1 Bit	I/O	CTRW -	DPT_Switch	0/1	[Pn] [X] Fan Control - Auto Mode	Switch Auto Mode on Long Press
	1 Bit	I/O	CTRW -	DPT_Switch	0/1	[Pn] [X] Fan Control - Auto Mode	Switch Auto Mode on Short Press
	1 Byte	I/O	CTRW -	DPT_Value_1_Ucount	0 - 255	[Pn] [X1] Two Objects - Long Press: (1-byte)	Send selected 1-byte value on long press
22, 35, 48, 61, 74, 87, 100, 113, 126, 139, 152, 165, 178, 191, 204, 217	1 Bit	I	C - - W -	DPT_Switch	0/1	[Pn] [X] LED On/Off	0 = Off; 1 = On
	1 Bit	I	C - - W -	DPT_Switch	0/1	[Pn] [X1] LED On/Off	0 = On; 1 = Off
	1 Bit	I	C - - W -	DPT_Switch	0/1	[Pn] [X1] LED On/Off	0 = Off; 1 = On
	1 Bit	I	C - - W -	DPT_Switch	0/1	[Pn] [X] LED On/Off	0 = On; 1 = Off
23, 36, 49, 62, 75, 88, 101, 114, 127, 140, 153, 166, 179, 192, 205, 218	14 Bytes	I	C - - W -	DPT_String_ASCII		[Pn] [X] Multimedia Text Indicator	14-byte String
24, 37, 50, 63, 76, 89, 102, 115, 128, 141, 154, 167, 180, 193, 206, 219	1 Bit	I/O	CTRW -	DPT_Switch	0/1	[Pn] [X2] Switch: "0"	1-bit Generic Control
	1 Byte	I/O	CTRW -	DPT_Value_1_Ucount	0 - 255	[Pn] [X2] Two Objects - Short Press: (1-byte)	Send selected 1-byte value on short press
	1 Bit	I/O	CTRW -	DPT_Switch	0/1	[Pn] [X2] Switch: "1"	1-bit Generic Control
	1 Bit	I/O	CTRW -	DPT_Switch	0/1	[Pn] [X2] Switch: "0/1"	1-bit Generic Control
	1 Bit	O	CTR - -	DPT_Switch	0/1	[Pn] [X2] Hold & Release	1-bit Generic Control
	1 Bit	O	CTR - -	DPT_Switch	0/1	[Pn] [X2] Light On/Off (Toggle)	0 = Off; 1 = On

	1 Bit	O	C T R - -	DPT_Step	0/1	[Pn] [X2] Stop Shutter/Step	0 = Stop Shutter/Step Up; 1 = Stop Shutter/Step Down
	1 Bit	I/O	C T R W -	DPT_Switch	0/1	[Pn] [X2] Two Objects - Short Press: "0"	1-bit Generic Control
	1 Bit	I/O	C T R W -	DPT_Switch	0/1	[Pn] [X2] Two Objects - Short Press: "1"	1-bit Generic Control
	1 Bit	I/O	C T R W -	DPT_Switch	0/1	[Pn] [X2] Two Objects - Short Press: "0/1"	1-bit Generic Control
	1 Bit	O	C T R - -	DPT_Trigger	0/1	[Pn] [X2] Stop Shutter	0, 1 = Stop Shutter
	1 Bit	I/O	C T R W -	DPT_Enable	0/1	[Pn] [X2] Enable Timer	0 = Disabled; 1 = Enabled
25, 38, 51, 64, 77, 90, 103, 116, 129, 142, 155, 168, 181, 194, 207, 220	1 Bit	I	C - - W -	DPT_Switch	0/1	[Pn] [X2] [Status] Switch	Value of 1-bit Generic Control
	1 Bit	I	C - - W -	DPT_Switch	0/1	[Pn] [X2] [Status] Hold & Release	Value of 1-bit Generic Control
	1 Bit	I	C - - W -	DPT_BinaryValue	0/1	[Pn] [X2] [Status] Two Objects - Short Press: (1-bit)	Value of short press object: (1-bit)
	1 Byte	I	C - - W -	DPT_Value_1_Ucount	0 - 255	[Pn] [X2] [Status] Two Objects - Short Press: (1-byte)	Value of short press object: (1-byte)
	1 Bit	I	C - - W -	DPT_Step	0/1	[Pn] [X2] [Status] Stop Shutter/Step	0 = Stop Shutter/Step Up; 1 = Stop Shutter/Step Down
	1 Bit	I	C - - W -	DPT_Trigger	0/1	[Pn] [X2] [Status] Stop Shutter	0, 1 = Stop Shutter
26, 39, 52, 65, 78, 91, 104, 117, 130, 143, 156, 169, 182, 195, 208, 221	4 Bit	O	C T R - -	DPT_Control_Dimming	0x0 (Detener) 0x1 (Reducir 100%) ... 0x7 (Reducir 1%) 0x8 (Detener) 0x9 (Subir 100%) ... 0xF (Subir 1%)	[Pn] [X2] Light Dimming (Toggle)	4-bit Dimming Control
27, 40, 53, 66, 79, 92, 105, 118, 131, 144, 157, 170, 183, 196, 209, 222	1 Bit	I	C - - W -	DPT_Switch	0/1	[Pn] [X2] [Status] Timer Control	1-bit Generic Control
	1 Byte	I	C - - W -	DPT_Scaling	0% - 100%	[Pn] [X2] Light Indicator	0 - 100%
	1 Byte	I	C - - W -	DPT_Scaling	0% - 100%	[Pn] [X2] Shutter Position	0 - 100%
	1 Bit	I	C - - W -	DPT_BinaryValue	0/1	[Pn] [X2] [Status] Two Objects - Long Press: (1-bit)	Value of long press object: (1-bit)
	1 Byte	I	C - - W -	DPT_Value_1_Ucount	0 - 255	[Pn] [X2] [Status] Two Objects - Long Press: (1-byte)	Value of long press object: (1-byte)
	1 Byte	I	C - - W -	DPT_Scaling	0% - 100%	[Pn] [X2] [Status] 1-byte Scaling Value	0 - 100%
	1 Byte	I	C - - W -	DPT_Value_1_Ucount	0 - 255	[Pn] [X2] [Status] 1-byte Unsigned Int Value	0 - 255
	1 Byte	I	C - - W -	DPT_Value_1_Count	-128 - 127	[Pn] [X2] [Status] 1-byte Signed Int Value	-128 - 127
	2 Bytes	I	C - - W -	DPT_Value_2_Ucount	0 - 65535	[Pn] [X2] [Status] 2-byte Unsigned Int Value	0 - 65535
	2 Bytes	I	C - - W -	DPT_Value_2_Count	-32768 - 32767	[Pn] [X2] [Status] 2-byte Signed Int Value	-32768 - 32767
	2 Bytes	I	C - - W -	9.xxx	-671088.64 - 670760.96	[Pn] [X2] [Status] 2-byte Float Value	-671088.64 - 670760.96
	1 Byte	I	C - - W -	DPT_Value_1_Ucount	0 - 255	[Pn] [X2] [Status] Enumeration	Rotating value change
28, 41, 54, 67, 80, 93, 106, 119, 132, 145, 158, 171, 184, 197, 210, 223	1 Bit	I/O	C T R W -	DPT_Switch	0/1	[Pn] [X2] Timer Control: "0"	1-bit Generic Control
	1 Bit	I/O	C T R W -	DPT_Switch	0/1	[Pn] [X2] Timer Control: "1"	1-bit Generic Control
	1 Bit	I/O	C T R W -	DPT_UpDown	0/1	[Pn] [X2] Move Shutter (Toggle Direction)	0 = Up; 1 = Down
	2 Bytes	I/O	C T R W -	DPT_Value_2_Ucount	0 - 65535	[Pn] [X2] 2-byte Unsigned Int Value	0 - 65535
	1 Byte	I/O	C T R W -	DPT_Value_1_Ucount	0 - 255	[Pn] [X2] 1-byte Unsigned Int Value	0 - 255
	1 Byte	I/O	C T R W -	DPT_Scaling	0% - 100%	[Pn] [X2] 1-byte Scaling Value	0 - 100%

	2 Bytes	I/O	C T R W -	9.xxx	-671088.64 - 670760.96	[Pn] [X2] 2-byte Float Value	-671088.64 – 670760.96
	1 Byte	I/O	C T R W -	DPT_Value_1_Count	-128 - 127	[Pn] [X2] 1-byte Signed Int Value	-128 - 127
	2 Bytes	I/O	C T R W -	DPT_Value_2_Count	-32768 - 32767	[Pn] [X2] 2-byte Signed Int Value	-32768 - 32767
	1 Bit	I/O	C T R W -	DPT_Switch	0/1	[Pn] [X2] Two Objects - Long Press: "0"	1-bit Generic Control
	1 Bit	I/O	C T R W -	DPT_Switch	0/1	[Pn] [X2] Two Objects - Long Press: "1"	1-bit Generic Control
	1 Bit	I/O	C T R W -	DPT_Switch	0/1	[Pn] [X2] Two Objects - Long Press: "0/1"	1-bit Generic Control
	1 Byte	I/O	C T R W -	DPT_Value_1_Ucount	0 - 255	[Pn] [X2] Enumeration	Rotating value change
29, 42, 55, 68, 81, 94, 107, 120, 133, 146, 159, 172, 185, 198, 211, 224	1 Byte	I/O	C T R W -	DPT_Value_1_Ucount	0 - 255	[Pn] [X2] Two Objects - Long Press: (1-byte)	Send selected 1-byte value on long press
	1 Bit	I	C - - W -	DPT_Switch	0/1	[Pn] [X2] LED On/Off	0 = On; 1 = Off
	1 Bit	I	C - - W -	DPT_Switch	0/1	[Pn] [X2] LED On/Off	0 = Off; 1 = On
225	1 Byte	I	C - - W -	DPT_SceneControl	0-63; 128-191	[Thermostat] Scene Input	Scene Value
226	2 Bytes	I	C - - W -	DPT_Value_Temp	-273,00 - 670760,00	[T1] Temperature Source 1	External Sensor Temperature
227	2 Bytes	I	C - - W -	DPT_Value_Temp	-273,00 - 670760,00	[T1] Temperature Source 2	External Sensor Temperature
228	2 Bytes	O	C T R - -	DPT_Value_Temp	-273,00 - 670760,00	[T1] Effective Temperature	Effective Control Temperature
229	1 Byte	I	C - - W -	DPT_HVACMode	1=Confort 2=Standby 3=Económico 4=Protección	[T1] Special Mode	1-byte HVAC Mode
230	1 Bit	I	C - - W -	DPT_Trigger	0/1	[T1] Special Mode: comfort	0 = Nothing; 1 = Trigger
	1 Bit	I	C - - W -	DPT_Switch	0/1	[T1] Special Mode: comfort	0 = Off; 1 = On
231	1 Bit	I	C - - W -	DPT_Trigger	0/1	[T1] Special Mode: standby	0 = Nothing; 1 = Trigger
	1 Bit	I	C - - W -	DPT_Switch	0/1	[T1] Special Mode: standby	0 = Off; 1 = On
232	1 Bit	I	C - - W -	DPT_Trigger	0/1	[T1] Special Mode: economy	0 = Nothing; 1 = Trigger
	1 Bit	I	C - - W -	DPT_Switch	0/1	[T1] Special Mode: economy	0 = Off; 1 = On
233	1 Bit	I	C - - W -	DPT_Trigger	0/1	[T1] Special Mode: protection	0 = Nothing; 1 = Trigger
	1 Bit	I	C - - W -	DPT_Switch	0/1	[T1] Special Mode: protection	0 = Off; 1 = On
234	1 Bit	I	C - - W -	DPT_Window_Door	0/1	[T1] Window Status (input)	0 = Closed; 1 = Open
235	1 Bit	I	C - - W -	DPT_Trigger	0/1	[T1] Comfort Prolongation	0 = Nothing; 1 = Timed Comfort
236	1 Byte	O	C T R - -	DPT_HVACMode	1=Confort 2=Standby 3=Económico 4=Protección	[T1] Special Mode Status	1-byte HVAC Mode
237	2 Bytes	I	C - - W -	DPT_Value_Temp	-273,00 - 670760,00	[T1] Setpoint	Thermostat Setpoint Input
	2 Bytes	I	C - - W -	DPT_Value_Temp	-273,00 - 670760,00	[T1] Basic Setpoint	Reference Setpoint
238	1 Bit	I	C - - W -	DPT_Step	0/1	[T1] Setpoint Step	0 = -0.5°C; 1 = +0.5°C
239	2 Bytes	I	C - - W -	DPT_Value_Tempd	-670760,00 - 670760,00	[T1] Setpoint Offset	Float Offset Value
240	2 Bytes	O	C T R - -	DPT_Value_Temp	-273,00 - 670760,00	[T1] Setpoint Status	Current Setpoint

241	2 Bytes	O	CTR--	DPT_Value_Temp	-273,00 - 670760,00	[T1] Basic Setpoint Status	Current Basic Setpoint
242	2 Bytes	O	CTR--	DPT_Value_Tempd	-670760,00 - 670760,00	[T1] Setpoint Offset Status	Current Setpoint Offset
243	1 Bit	I	C--W-	DPT_Reset	0/1	[T1] Setpoint Reset	Reset Setpoint to Default
	1 Bit	I	C--W-	DPT_Reset	0/1	[T1] Offset Reset	Reset offset
244	1 Bit	I	C--W-	DPT_Heat_Cool	0/1	[T1] Mode	0 = Cool; 1 = Heat
245	1 Bit	O	CTR--	DPT_Heat_Cool	0/1	[T1] Mode Status	0 = Cool; 1 = Heat
246	1 Bit	I	C--W-	DPT_Switch	0/1	[T1] On/Off	0 = Off; 1 = On
247	1 Bit	O	CTR--	DPT_Switch	0/1	[T1] On/Off Status	0 = Off; 1 = On
248	1 Byte	O	CTR--	DPT_Scaling	0% - 100%	[T1] Control Variable (Cool)	PI Control (Continuous)
249	1 Byte	O	CTR--	DPT_Scaling	0% - 100%	[T1] Control Variable (Heat)	PI Control (Continuous)
250	1 Bit	O	CTR--	DPT_Switch	0/1	[T1] Control Variable (Cool)	2-Point Control
	1 Bit	O	CTR--	DPT_Switch	0/1	[T1] Control Variable (Cool)	PI Control (PWM)
251	1 Bit	O	CTR--	DPT_Switch	0/1	[T1] Control Variable (Heat)	2-Point Control
	1 Bit	O	CTR--	DPT_Switch	0/1	[T1] Control Variable (Heat)	PI Control (PWM)
252	1 Bit	O	CTR--	DPT_Switch	0/1	[T1] Additional Cool	Temp >= (Setpoint+Band) => "1"
253	1 Bit	O	CTR--	DPT_Switch	0/1	[T1] Additional Heat	Temp <= (Setpoint-Band) => "1"
254	1 Bit	O	CTR--	DPT_Switch	0/1	[T1] PI State (Cool)	0 = PI signal 0%; 1 = PI signal greater than 0%
255	1 Bit	O	CTR--	DPT_Switch	0/1	[T1] PI State (Heat)	0 = PI signal 0%; 1 = PI signal greater than 0%
256, 260	2 Bytes	O	CTR--	DPT_Value_Temp	-273,00 - 670760,00	[I1] Current Temperature	Temperature sensor value
257, 261	1 Bit	O	CTR--	DPT_Alarm	0/1	[I1] Overcooling	0 = No Alarm; 1 = Alarm
258, 262	1 Bit	O	CTR--	DPT_Alarm	0/1	[I1] Overheating	0 = No Alarm; 1 = Alarm
259, 263	1 Bit	O	CTR--	DPT_Alarm	0/1	[I1] Probe Error	0 = No Alarm; 1 = Alarm
264	2 Bytes	O	CTR--	DPT_Value_Temp	-273,00 - 670760,00	[Internal Probe] Current Temperature	Temperature sensor value
265	1 Bit	O	CTR--	DPT_Alarm	0/1	[Internal Probe] Overcooling	0 = No Alarm; 1 = Alarm
266	1 Bit	O	CTR--	DPT_Alarm	0/1	[Internal Probe] Overheating	0 = No Alarm; 1 = Alarm
267	1 Byte	I	C--W-	DPT_SceneControl	0-63; 128-191	[Motion Detector] Scene Input	Scene Value
268	1 Byte		CT---	DPT_SceneControl	0-63; 128-191	[Motion Detector] Scene Output	Scene Value
269, 298	1 Byte	O	CTR--	DPT_Scaling	0% - 100%	[In] Luminosity	0-100%
270, 299	1 Bit	O	CTR--	DPT_Alarm	0/1	[In] Open Circuit Error	0 = No Error; 1 = Open Circuit Error
271, 300	1 Bit	O	CTR--	DPT_Alarm	0/1	[In] Short Circuit Error	0 = No Error; 1 = Short Circuit Error
272, 301	1 Byte	O	CTR--	DPT_Scaling	0% - 100%	[In] Presence State (Scaling)	0-100%
273, 302	1 Byte	O	CTR--	DPT_HVACMode	1=Confort 2=Standby 3=Económico 4=Protección	[In] Presence State (HVAC)	Auto, Comfort, Standby, Economy, Building Protection
274, 303	1 Bit	O	CTR--	DPT_Occupancy	0/1	[In] Presence State (Binary)	Binary Value
	1 Bit	O	CTR--	DPT_Trigger	0/1	[In] Presence: Slave Output	1 = Motion Detected
275, 304	1 Bit	I	C--W-	DPT_Window_Door	0/1	[In] Presence Trigger	Binary Value to Trigger the Presence Detection

276, 305	1 Bit	I	C - - W -	DPT_Trigger	0/1	[In] Presence: Slave Input	0 = Nothing; 1 = Detection from slave device
277, 306	2 Bytes	I	C - - W -	DPT_TimePeriodSec		[In] Presence: Waiting Time	0-65535 s.
278, 307	2 Bytes	I	C - - W -	DPT_TimePeriodSec		[In] Presence: Listening Time	1-65535 s.
279, 308	1 Bit	I	C - - W -	DPT_Switch	0/1	[In] Presence: Enable	According to parameters
280, 309	1 Bit	I	C - - W -	DPT_Switch	0/1	[In] Presence: Day/Night	According to parameters
281, 310	1 Bit	O	C T R - -	DPT_Occupancy	0/1	[In] Presence: Occupancy State	0 = Not Occupied; 1 = Occupied
282, 311	1 Bit	I	C - - W -	DPT_Trigger	0/1	[In] External Motion Detection	0 = Nothing; 1 = Motion detected by an external sensor
283, 288, 293, 312, 317, 322	1 Byte	O	C T R - -	DPT_Scaling	0% - 100%	[In] [Cj] Detection State (Scaling)	0-100%
284, 289, 294, 313, 318, 323	1 Byte	O	C T R - -	DPT_HVACMode	1=Confort 2=Standby 3=Económico 4=Protección	[In] [Cj] Detection State (HVAC)	Auto, Comfort, Standby, Economy, Building Protection
285, 290, 295, 314, 319, 324	1 Bit	O	C T R - -	DPT_Switch	0/1	[In] [Cj] Detection State (Binary)	Binary Value
286, 291, 296, 315, 320, 325	1 Bit	I	C - - W -	DPT_Switch	0/1	[In] [Cj] Enable Channel	According to parameters
287, 292, 297, 316, 321, 326	1 Bit	I	C - - W -	DPT_Switch	0/1	[In] [Cj] Force State	0 = No Detection; 1 = Detection
327, 333	1 Bit	I	C - - W -	DPT_Switch	0/1	[In] Input Lock	1 = Locked; 0 = Unlocked
328, 334	1 Bit		C T - - -	DPT_Switch	0/1	[In] [Short Press] 0	Sending of 0
	1 Bit		C T - - -	DPT_Switch	0/1	[In] [Short Press] 1	Sending of 1
	1 Bit	I	C T - W -	DPT_Switch	0/1	[In] [Short Press] 0/1 Switching	Switching 0/1
	1 Bit		C T - - -	DPT_UpDown	0/1	[In] [Short Press] Move Up Shutter	Sending of 0 (Up)
	1 Bit		C T - - -	DPT_UpDown	0/1	[In] [Short Press] Move Down Shutter	Sending of 1 (Down)
	1 Bit		C T - - -	DPT_UpDown	0/1	[In] [Short Press] Move Up/Down Shutter	Switching 0/1 (Up/Down)
	1 Bit		C T - - -	DPT_Step	0/1	[In] [Short Press] Stop/Step Up Shutter	Sending of 0 (Stop/Step Up)
	1 Bit		C T - - -	DPT_Step	0/1	[In] [Short Press] Stop/Step Down Shutter	Sending of 1 (Stop/Step Down)
	1 Bit		C T - - -	DPT_Step	0/1	[In] [Short Press] Stop/Step Shutter (Switched)	Switching of 0/1 (Stop/Step Up/Down)
	4 Bit		C T - - -	DPT_Control_Dimming	0x0 (Detener) 0x1 (Reducir 100%) ... 0x7 (Reducir 1%) 0x8 (Detener) 0x9 (Subir 100%) ... 0xF (Subir 1%)	[In] [Short Press] Brighter	Increase Brightness
	4 Bit		C T - - -	DPT_Control_Dimming	0x0 (Detener) 0x1 (Reducir 100%) ... 0x7 (Reducir 1%) 0x8 (Detener) 0x9 (Subir 100%)	[In] [Short Press] Darker	Decrease Brightness

					... 0xF (Subir 1%)		
	4 Bit		CT---	DPT_Control_Dimming	0x0 (Detener) 0x1 (Reducir 100%) ... 0x7 (Reducir 1%) 0x8 (Detener) 0x9 (Subir 100%) ... 0xF (Subir 1%)	[In] [Short Press] Brighter/Darker	Switch Bright/Dark
	1 Bit		CT---	DPT_Switch	0/1	[In] [Short Press] Dimmer ON	Sending of 1 (ON)
	1 Bit		CT---	DPT_Switch	0/1	[In] [Short Press] Dimmer OFF	Sending of 0 (OFF)
	1 Bit	I	CT-W-	DPT_Switch	0/1	[In] [Short Press] Dimmer ON/OFF	Switching 0/1
	1 Byte		CT---	DPT_SceneControl	0-63; 128-191	[In] [Short Press] Run Scene	Sending of 0 - 63
	1 Byte		CT---	DPT_SceneControl	0-63; 128-191	[In] [Short Press] Save Scene	Sending of 128 - 191
	1 Bit	I/O	CTRW-	DPT_Switch	0/1	[In] [Switch/Sensor] Edge	Sending of 0 or 1
	1 Byte		CT---	DPT_Value_1_Ucount	0 - 255	[In] [Short Press] Constant Integer Value	0 - 255
	1 Byte		CT---	DPT_Scaling	0% - 100%	[In] [Short Press] Constant Percentage Value	0% - 100%
	2 Bytes		CT---	DPT_Value_2_Ucount	0 - 65535	[In] [Short Press] Constant Integer Value	0 - 65535
	2 Bytes		CT---	9.xxx	-671088.64 - 670760.96	[In] [Short Press] Constant Float Value	Float value
329, 335	1 Byte	I	C--W-	DPT_Scaling	0% - 100%	[In] [Short Press] Shutter Status (Input)	0% = Top; 100% = Bottom
	1 Byte	I	C--W-	DPT_Scaling	0% - 100%	[In] [Short Press] Dimming Status (Input)	0% - 100%
330, 336	1 Bit		CT---	DPT_Switch	0/1	[In] [Long Press] 0	Sending of 0
	1 Bit		CT---	DPT_Switch	0/1	[In] [Long Press] 1	Sending of 1
	1 Bit	I	CT-W-	DPT_Switch	0/1	[In] [Long Press] 0/1 Switching	Switching 0/1
	1 Bit		CT---	DPT_UpDown	0/1	[In] [Long Press] Move Up Shutter	Sending of 0 (Up)
	1 Bit		CT---	DPT_UpDown	0/1	[In] [Long Press] Move Down Shutter	Sending of 1 (Down)
	1 Bit		CT---	DPT_UpDown	0/1	[In] [Long Press] Move Up/Down Shutter	Switching 0/1 (Up/Down)
	1 Bit		CT---	DPT_Step	0/1	[In] [Long Press] Stop/Step Up Shutter	Sending of 0 (Stop/Step Up)
	1 Bit		CT---	DPT_Step	0/1	[In] [Long Press] Stop/Step Down Shutter	Sending of 1 (Stop/Step Down)
	1 Bit		CT---	DPT_Step	0/1	[In] [Long Press] Stop/Step Shutter (Switched)	Switching of 0/1 (Stop/Step Up/Down)
	4 Bit		CT---	DPT_Control_Dimming	0x0 (Detener) 0x1 (Reducir 100%) ... 0x7 (Reducir 1%) 0x8 (Detener) 0x9 (Subir 100%) ... 0xF (Subir 1%)	[In] [Long Press] Brighter	Long Pr. -> Brighter; Release -> Stop
	4 Bit		CT---	DPT_Control_Dimming	0x0 (Detener) 0x1 (Reducir 100%) ... 0x7 (Reducir 1%)	[In] [Long Press] Darker	Long Pr. -> Darker; Release -> Stop

					0x8 (Detener) 0x9 (Subir 100%) ... 0xF (Subir 1%)		
	4 Bit		C T - - -	DPT_Control_Dimming	0x0 (Detener) 0x1 (Reducir 100%) ... 0x7 (Reducir 1%) 0x8 (Detener) 0x9 (Subir 100%) ... 0xF (Subir 1%)	[In] [Long Press] Brighter/Darker	Long Pr. -> Brighter/Darker; Release -> Stop
	1 Bit		C T - - -	DPT_Switch	0/1	[In] [Long Press] Dimmer ON	Sending of 1 (ON)
	1 Bit		C T - - -	DPT_Switch	0/1	[In] [Long Press] Dimmer OFF	Sending of 0 (OFF)
	1 Bit	I	C T - W -	DPT_Switch	0/1	[In] [Long Press] Dimmer ON/OFF	Switching 0/1
	1 Byte		C T - - -	DPT_SceneControl	0-63; 128-191	[In] [Long Press] Run Scene	Sending of 0 - 63
	1 Byte		C T - - -	DPT_SceneControl	0-63; 128-191	[In] [Long Press] Save Scene	Sending of 128 - 191
	1 Bit	O	C T R - -	DPT_Alarm	0/1	[In] [Switch/Sensor] Alarm: Breakdown or sabotage	1 = Alarm; 0 = No Alarm
	2 Bytes		C T - - -	9.xxx	-671088.64 - 670760.96	[In] [Long Press] Constant Float Value	Float value
	2 Bytes		C T - - -	DPT_Value_2_Ucount	0 - 65535	[In] [Long Press] Constant Integer Value	0 - 65535
	1 Byte		C T - - -	DPT_Scaling	0% - 100%	[In] [Long Press] Constant Percentage Value	0% - 100%
331, 337	1 Bit		C T - - -	DPT_Trigger	0/1	[In] [Long Press/Release] Stop Shutter	Release -> Stop Shutter
332, 338	1 Byte	I	C - - W -	DPT_Scaling	0% - 100%	[In] [Long Press] Dimming Status (Input)	0% - 100%
	1 Byte	I	C - - W -	DPT_Scaling	0% - 100%	[In] [Long Press] Shutter Status (Input)	0% = Top; 100% = Bottom

Join and send us your inquiries
about Zennio devices:

<http://support.zennio.com>

Zennio Avance y Tecnología S.L.
C/ Río Jarama, 132. Nave P-8.11
45007 Toledo (Spain).

Tel. +34 925 232 002

www.zennio.com
info@zennio.com



RoHS