Soft Push button

KNX Soft Push button 55x55 & 70x70 with KNX security

ZS55SOFT

ZS70SOFT

Application Program Version: [1.1]
User Manual Version: [1.1]_b

www.zennio.com

USER MANUAL

CONTENTS

DOCUMENT UPDATES

Version	Changes	Page(s)
[1.1]_a	**Changes in the application program**: - Double press. - Cleaning function. - New independent lock objects for each push button. - New controls and new sending options in existing controls.	-
[1.1]_b	**New range:** - Soft Push button 70x70	

1 INTRODUCTION

1.1 SOFT PUSH BUTTON

The range of **KNX Soft Push button** from Zennio are a fully customisable solution for the control of rooms where user control of air conditioning systems, lighting, blinds, scenes, etc. is required.

One of the key features of this product is its design, which enables the seamless integration of **conventional-looking push buttons** into **KNX** home automation systems.

They are available in two different sizes: **55x55** mm and **70x70** mm, with the possibility of configuring **one**, **two** or **four** activation zones, due to the existence of different versions of single and double keys, with activation at the bottom or at the top and bottom.

Important: *regardless of the size, the application program will be the same and can therefore be used with any size of device.*

The versatility offered by the functionality of buttons is complemented by a configurable analogue/digital input as a **temperature probe,** an **internal temperature sensor** and a **thermostat** function, as well as an elegant design, where the customer can choose the icons, texts and colours and can make use of his own logos. Visit our online Touch-My Design tool to customise your KNX Soft Push button.

The most outstanding features are:

- **1 / 2 / 4 buttons**, which can operate as individual or pair controls.
- Option to set an additional action on individual controls when **double press**.
- Possibility of **locking / unlocking all push buttons** through binary objects or scenes, and of setting a timed locking of the entire device (**cleaning function**).
  - ➢ **Independent** binary locking / unlocking object per control, as well as the possibility to configure an individualised **timed locking** for each control.
- **Welcome Back object** (binary or scene) which is sent to the KNX bus when a pulsation is detected after a certain period (configurable) of inactivity.
- Built-in **temperature sensor.**

- 1 analogue/digital input configurable as **external temperature probe**.
- Independent **Thermostat** function.
- **Heartbeat** or periodical “still-alive” notification.
- **KNX Security**. For detailed information about the functionality and configuration of KNX security, consult the specific user manual “KNX Security”, available in the product section of the Zennio web portal (www.zennio.com).

2 CONFIGURATION

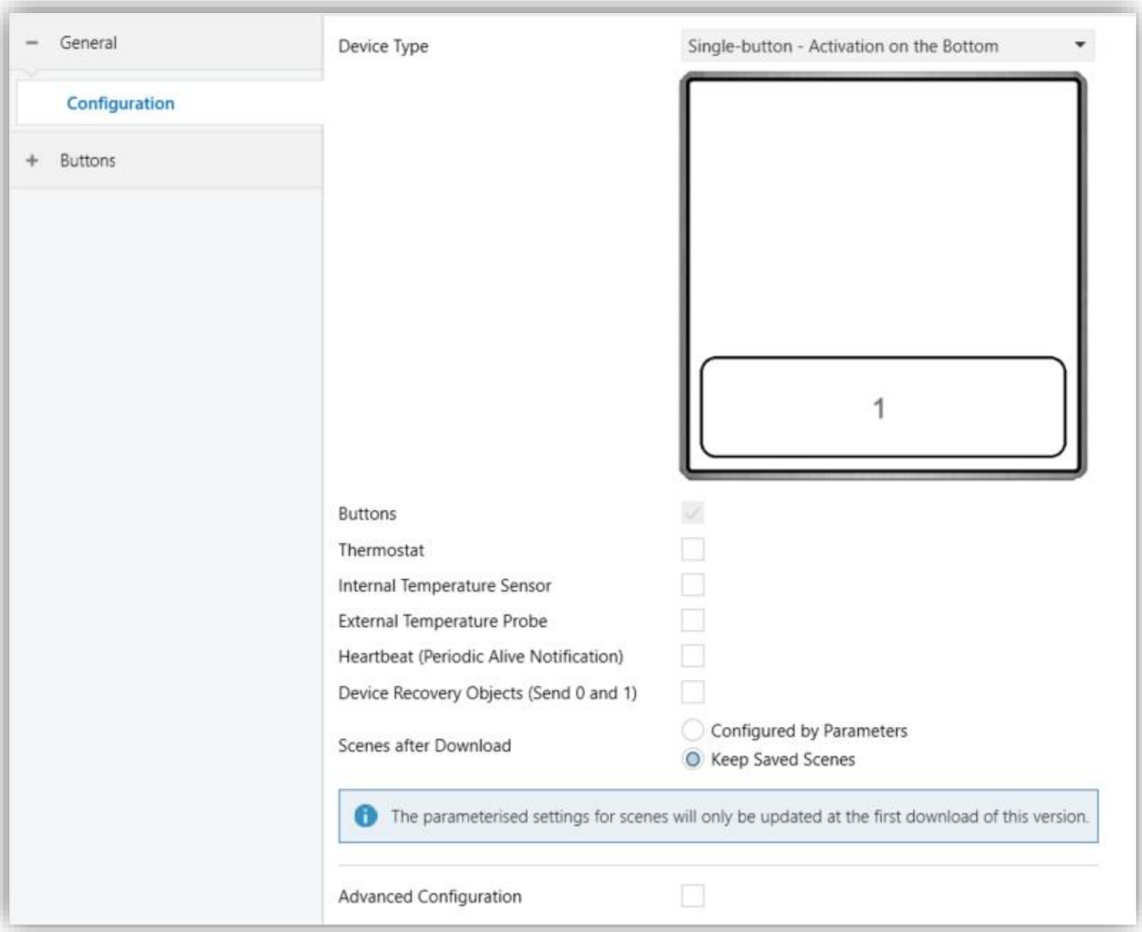
2.1 GENERAL

After importing the corresponding database in ETS and adding the device into the topology of the desired project, the configuration process begins by entering the Parameters tab of the device.

2.1.1 CONFIGURATION

In the "Configuration" tab, the general settings are displayed.

ETS PARAMETERISATION



Device Type: Single-button - Activation on the Bottom

Buttons ☑

Thermostat □

Internal Temperature Sensor □

External Temperature Probe □

Heartbeat (Periodic Alive Notification) □

Device Recovery Objects (Send 0 and 1) □

Scenes after Download: ○ Configured by Parameters ● Keep Saved Scenes

The parameterised settings for scenes will only be updated at the first download of this version.

Advanced Configuration □

Figure 1. Main Configuration.

With regard to the configuration of the push zones, the following parameter is available:

- **Device Type** [*Single-button - Activation on the Bottom / Single-button - Activation on the Top and Bottom / Double-button - Activation on the Bottom / Double-button - Activation on the Top and Bottom*][1]. Determines the distribution of the buttons and push zones.

  Important: *for correct operation of the device, **it is recommended** that the selection of this parameter be in accordance with the arrangement of the keys attached to the real device.*

  According to option selected, ETS will display an image of the button distribution.

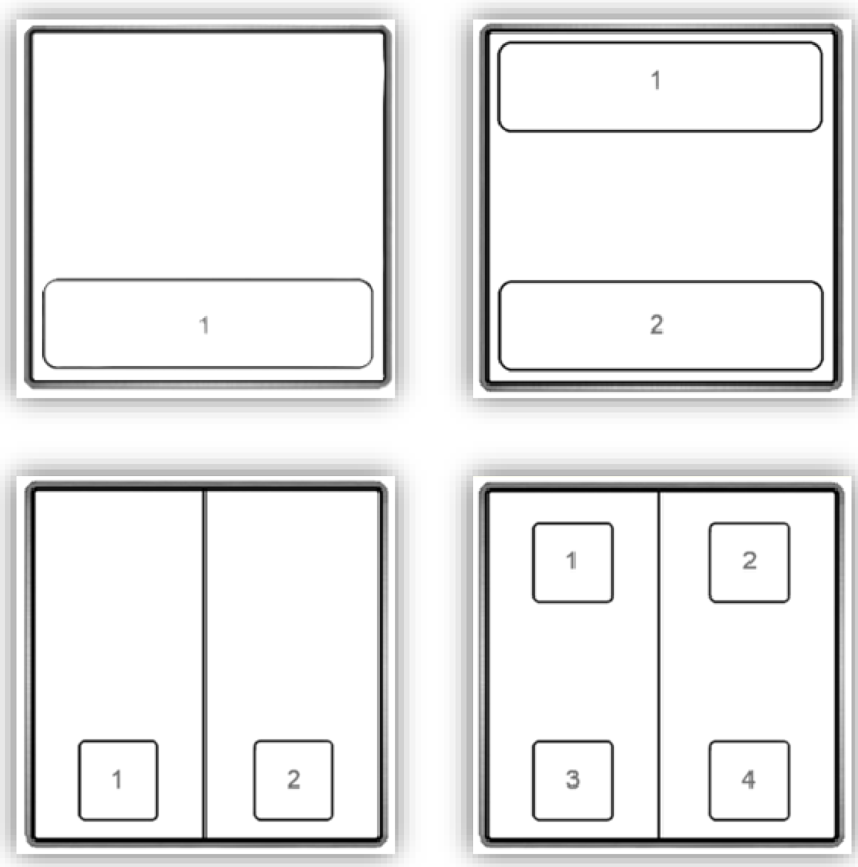


Figure 2. Image displayed in ETS depending on the parameter **Device Type.**

- **Buttons** [*enabled*]: read-only parameter to make it evident that the “Buttons” tab is always enabled in the tab tree on the left. See section 2.2 for details.

[1] The default values of each parameter will be highlighted in blue in this document, as follows: [*default /* rest of options].

- **Thermostat** [*disabled / enabled*]: enables or disables the “Thermostat” tab in the tree on the left. See section **¡Error! No se encuentra el origen de la referencia.** f or details.

- **Internal Temperature Sensor** [*disabled / enabled*]: enables or disables the “Internal Temperature Sensor” tab in the tree on the left. See section 2.1.2 for details.

- **External Temperature Probe** [*disabled / enabled*]: enables or disables the “External Temperature Probe” tab in the tree on the left. See section 2.1.3 for details.

- **Heartbeat (Periodic Alive Notification)** [*disabled / enabled*]: incorporates a one-bit object to the project (“**[Heartbeat] Object to Send ‘1’**”) that will be sent periodically with value “1” to notify that the device is still working (*still alive*).

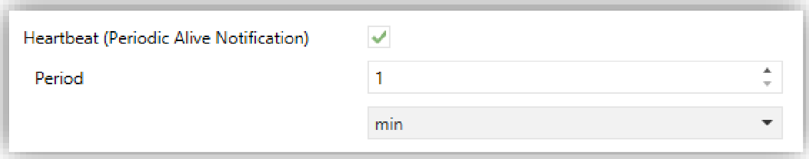

Heartbeat (Periodic Alive Notification) ✓
Period 1
min

Figure 3. Heartbeat.

Note: *the first sending after download or bus failure takes place with a delay of up to 255 seconds, to prevent bus overload. The following sendings follow the parameterised period.*

- **Device Recovery Objects (Send 0 and 1)** [*disabled / enabled*]: this parameter lets the integrator activate two new communication objects (“**[Heartbeat] Device Recovery**”), which will be sent to the KNX bus with values “0” and “1” whenever the device begins operation (for example, after a bus power failure). It is possible to parameterise a certain **delay** [*0…255*] [*s*] to this sending.

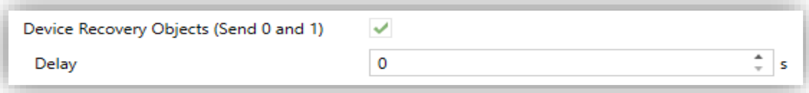

Device Recovery Objects (Send 0 and 1) ✓
Delay 0 s

Figure 4. Device Recovery Objects.

Note: *After download or bus failure, the sending takes place with a delay of up to 6,35 seconds plus the parameterised delay, to prevent bus overload.*

- **Scenes after download** [*Configured by Parameters / Keep Saved Scenes*]: Allows to assign the option to update scenes only on the first download of the device version or to delete the saved scenes after the download of the device version.

  Note: *if "Keep Saved Scenes" option has been configured, but it is the first download of the device or a different version from the current one, the values configured by parameter will be adopted. If new scenes are added in successive downloads, it will be necessary to perform a download by checking the option "Configured by Parameters" to ensure the correct operation of these scenes.*

- **Advanced Configuration** [*disabled / enabled*]: enables or disables the "Advanced" tab in the tree on the left. See section 2.1.3 for details.

2.1.2 INTERNAL TEMPERATURE SENSOR

The **internal temperature probe** 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 Probe**" (available in the product section at the Zennio homepage, www.zennio.com) for detailed information about the functionality and the configuration of the related parameters.

2.1.3 EXTERNAL TEMPERATURE PROBE

Configuration to connect a **Zennio** temperature sensor, 6.8k or 10k Ohms, or a custom NTC probe. Please refer to the specific user manual "**Temperature Probe**", available in the product section, at the Zennio website (www.zennio.com).

2.1.4 THERMOSTAT

Soft Push button implements **one Zennio thermostat** which can be enabled and fully customised.

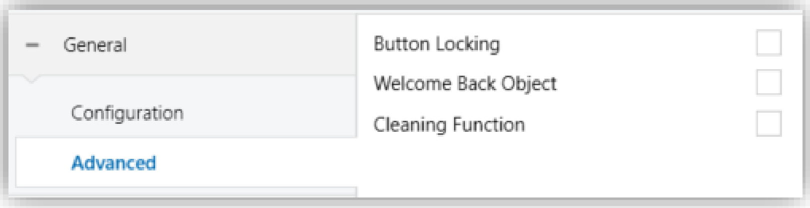
Please refer to the specific manual "**Zennio Thermostat**" (available in the product section at the Zennio website, www.zennio.com) for detailed information about the functionality and the configuration of the related parameters.

2.1.5 ADVANCED CONFIGURATION

Independent tab for the parameterisation of several advanced functions; **Button Locking**, **Welcome Back Object** and **Cleaning Function**. These functions are explained next:

ETS PARAMETERISATION

After enabling the **Advanced configuration** from "Configuration" screen (see section 2.1.1), a new tab will be incorporated into the tree on the left.



Button Locking □
Welcome Back Object □
Cleaning Function □

Figure 5. Advanced.

- **Button Locking** [*disabled / enabled*]: enables or disables the "Button Locking" tab in the tree on the left. See section 2.1.5.1 for details.

- **Welcome Back Object** [*disabled / enabled*]: enables or disables the "Welcome Back Object" tab in the tree on the left. See section 2.1.5.2 for details.

- **Cleaning Function** [*disabled / enabled*]: enables or disables the "Cleaning Function" tab in the tree on the left. See section 2.1.5.3 for details.

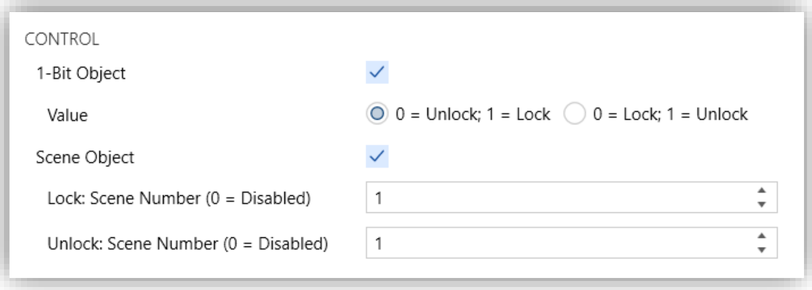
2.1.5.1 BUTTON LOCKING

The push buttons 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, pressing on the buttons will be ignored: no actions will be performed when the user touches on any of the controls.

ETS PARAMETERISATION

After enabling **Button Locking** in "Advanced" tab, a new tab will be incorporated into the tree on the left.



CONTROL

1-Bit Object ☑

Value ● 0 = Unlock; 1 = Lock ○ 0 = Lock; 1 = Unlock

Scene Object ☑

Lock: Scene Number (0 = Disabled) 1

Unlock: Scene Number (0 = Disabled) 1

Figure 6. Button Locking; Control.

- **1 Bit Object** [*disabled* / *enabled*]: when marked, the options will show up to select which value should trigger which action.
  - **Value** [*0 = Unlock; 1 = Lock* / *0 = Lock; 1 = Unlock*]: these values are received through the object "**[General] Button Locking**".
- **Scene Object** [*disabled* / *enabled*]: when marked, two specific textboxes will show up to enter the scene numbers (0 - 64) that should trigger each action. These values are to be received through the general "**[General] Scene: Receive**" object.
  - **Lock: Scene Number (0 = Disabled)** [*0…1...64*].
  - **Unlock: Scene Number (0 = Disabled)** [*0…1...64*].

2.1.5.2 WELCOME BACK OBJECT

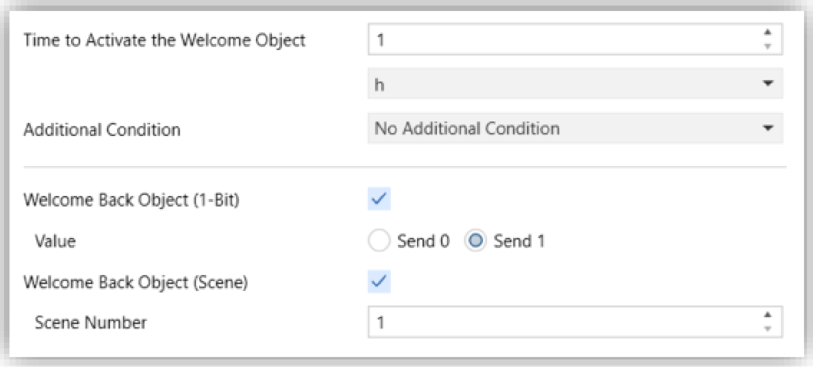
Push buttons can send a specific object (the *welcome back object*) to the KNX bus when the user presses a button after a significant amount of time since the last. Sending it or not can also depend on an **additional configurable condition** consisting in the evaluation of up to five binary objects.

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 that the welcome back object is 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**, a new tab will be incorporated into the tree on the left.



Time to Activate the Welcome Object: 1 h

Additional Condition: No Additional Condition

Welcome Back Object (1-Bit): ☑

Value: ○ Send 0 ● Send 1

Welcome Back Object (Scene): ☑

Scene Number: 1

Figure 7. Welcome Back Object.

This screen contains the following parameters:

- **Timeout to Activate the Welcome Object** [*1…255*] [*s / min / h*]: sets the minimum time 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)** [*disabled / enabled*]: 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 should set in **Value** [*Send 0 / Send 1*].

- **Welcome Back Object (Scene)** [*disabled / enabled*]: 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 value should be set in **Scene Number** [*1…64*].

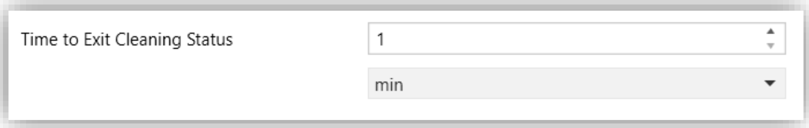
2.1.5.3 CLEANING FUNCTION

This feature is very similar to the blocking of all push buttons, that is, it is used to block and discard any type of pressing on them. The difference is that this function remains active only during a parameterisable time and then stops.

This function is intended to let the user clean the push buttons with the certainty of not triggering unwanted actions.

ETS PARAMETERISATION

After enabling **Cleaning Function** from "Advanced" screen, a new tab will be incorporated into the tree on the left.



Time to Exit Cleaning Status: 1 min

Figure 8. Cleaning Function.

- **Time to Exit Cleaning Status** [*5…15…255*] [*s*] / [*1…255*] [*min / h*]: timeout to deactivate the cleaning function once triggered.

2.2 BUTTONS

The device has **4 real buttons** available for the user to execute actions, and due to the different key versions available, it is possible to configure the device to have **one**, **two** or **four** buttons.

The distribution of the activation zones will depend on the model of the device selected in "Configuration" (see section 2.1.1), being possible to configure them as single button controls or in pairs by **combining any two of them**.

- **Single-button - Activation on the Bottom**: only one individual control is possible.
- **Single-button - Activation on the Top and Bottom**: up to two individual controls or a pair.
- **Double-button - Activation on the Bottom**: up to two individual controls or a pair.
- **Double-button - Activation on the Top and Bottom:** up to four one-button controls can be configured, or up to two two-button controls.

2.2.1 CONFIGURATION

The following is a list of the functions that can be assigned to each button.

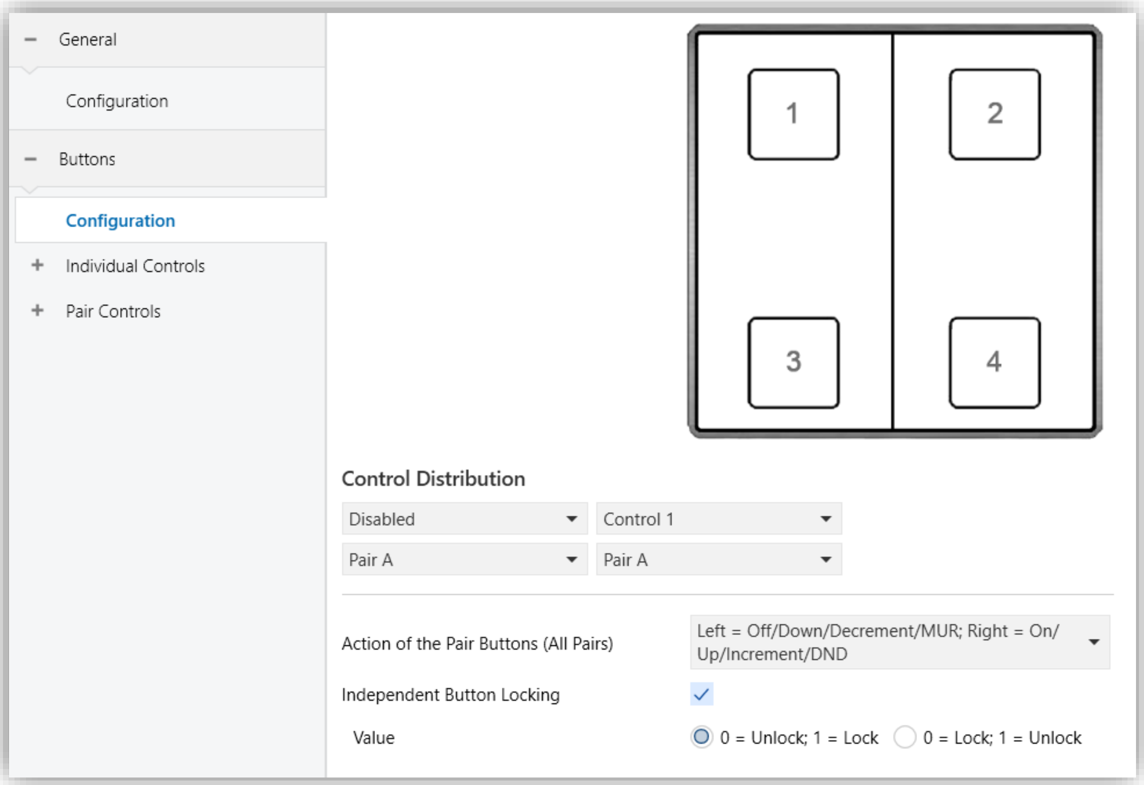
- **Disabled.** The push button will not react to keystrokes.
- **Pair A** or **B.** The number of available pairs being the function of such pair one of the following:
  - Switch.
  - Two objects (Short Press / Long Press).
  - Dimmer.
  - Shutter.
  - Enumeration.
  - Counter.
  - Romo State (Indoor).

- **Individual** (one-button control):
  - Switch.
  - Hold & Release.
  - Two objects (Short Press / Long Press).
  - Scene.
  - Scaling Constant.
  - Counter Constant.
  - Float Constant.
  - Dimmer.
  - Shutter.
  - Room State (Indoor).
  - Enumeration.

The next sections explain the configuration involved for each of the above functions.

ETS PARAMETERISATION

An independent tab for the parameterisation of the buttons is shown in ETS by default, initially containing only a sub-tab named “Configuration”.



Control Distribution

Disabled	Control 1
Pair A	Pair A

Action of the Pair Buttons (All Pairs): Left = Off/Down/Decrement/MUR; Right = On/Up/Increment/DND

Independent Button Locking: ☑

Value: ● 0 = Unlock; 1 = Lock ○ 0 = Lock; 1 = Unlock

Figure 9. Buttons - Configuration.

- **Control Distribution** [*Disabled / Control 1 / … / Control 4 / Pair A / Pair B*]: selects which individual control, or member of a pair, will be placed to each push button.

> **Note**: *in case of associating a disabled individual control, o pair, to a push button, an error message will appear in ETS. An error message will also appear if the same pair is configured to more than 2 different push buttons.*

If at least one two-button control is being configured (either Pair X), an additional parameter (**Action of the Pair Buttons (All Pairs)**) will be available to determine an operation criterion. The options are:

- [*Left = Off/Down/Decrement/MUR; Right = On/Up/Increment/DND*].
- [*Right = Off/Down/Decrement/DND; Left = On/Up Increment/MUR*].
- [*Every button pair is configured separately*].

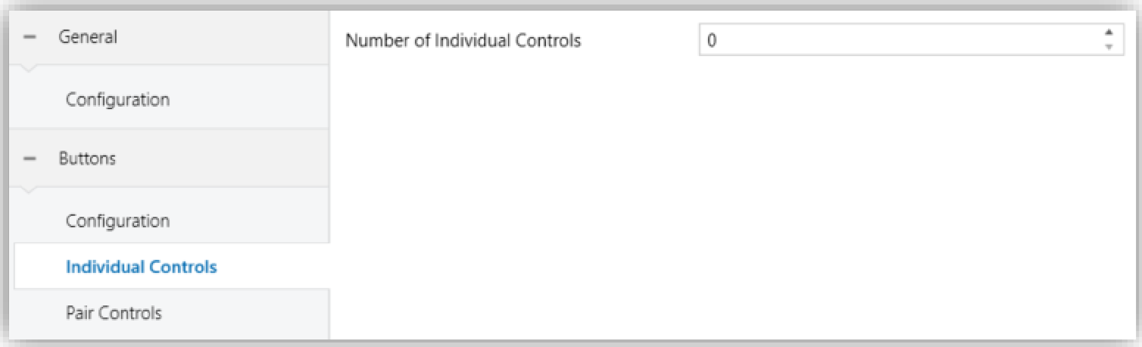
> **Note**: *the parameterisable options will vary according to the relative position of the buttons of the pairs, so the nomenclature of "Left" may be changed to "Lower" and "Right" to "Upper".*

- **Independent Lock** [*disabled / enabled*]: if this parameter is enabled, a new homonym parameter will be displayed on the tab of each control (either individual or pair).
  - **Value** [*0 = Unlock; 1 = Lock / 0 = Lock; 1 = Unlock*]: polarity taken by all independent locking communication objects if enabled in their respective tabs.

2.2.2 INDIVIDUAL CONTROLS

The application program provides up to 4 different individual controls. Each of these controls will be configured independently and placed on the desired push button (see section 2.2.1) allowing the possibility to configure the same control for several push buttons.

ETS PARAMETERISATION



Number of Individual Controls: 0

Figure 10. Buttons - Individual Controls.

- **Number of Individual Controls** [*0…4*]: selects the number of available controls. For each one, a new tab will be displayed in the left tree ("x Control", see section 2.2.2.1).

2.2.2.1 X CONTROL

Buttons configured to work as individual (separate) controls can be assigned any of the following control functions:

- **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", "1" or alternate with every touch according to the sequence 1 → 0 → 1 → …

- **Hold & Release**: when the user touches the button, a configurable value (binary, 1 byte with and without sign, or 2 bytes with or without sign) will be sent to the KNX bus. When the user releases the button, another configurable value (binary, 1 byte with and without sign, or 2 bytes with or without sign) will be sent through the same object if the same type of data point (DPT) has been selected for both actions.

  In the case of configuring different data point types for each action, the parameterised value shall be sent by its corresponding object.

- **Two Objects (Short Press/Long Press)**: specific configurable values (binary, 1-byte signed or unsigned, or 2-byte signed or unsigned) will be sent both after a short or a long press (a different object will be used in each case).

- **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.

- **Scaling Constant**: sends a percentage value (configurable) to the bus when the user touches the button.

- **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 1. Value range - Counter type constant.

- **Float Constant**: sends a two-byte floating point value (configurable) to the bus when the user touches the button. The available range is -671088.64 to 670433.28.

- **Dimmer**: implements a one-button light control that sends orders to the KNX bus, which can then be executed by light dimmers. These orders can be configured for short press or long press:

  - **Action on Short Press**:

    - [*Send On*]*:* a value of On is sent.
    - [*Send Off*]*:* a value of Off is sent.
    - [*Switch Between On and Off*]: toggles between sending On and Off. Commutation is initiated according to the lighting percentage status. If a value of 0% is present, an On will be sent. Otherwise, the value sent will be an Off.
    - [*Scene*]: the configured scene value is sent.
    - [*Absolute Dimming*]: a percentage set in the range [*0-100*] [*%*] is sent.

  - **Action on Long Press**:

    - [*Dimming Down*]: the parameterised control step is adjusted downwards.

- [*Dimming Up*]*:* the parameterised control pitch is adjusted upwards.
- [*Switch Between Dimming Up and Down*]*:* toggles between dimming up and down based on the last percentage status value. In case of 0%, it will send the object to increment, and vice versa for the value of 100%. If this value is between 1-99%, it will send the inverse regulation step to the last one sent.

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 recommended.

Note: *after a bus recovery, the light dimmer should send back the status object so the control 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).

Push buttons switches are 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.

- [*Hold & Release*]: the device will send an order to move the shutter when 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.

- **Room State (Indoor):** allows controlling the room states ("*Normal*", "*Make Up Request" and "Do Not Disturb*"). Pressing the button will activate the *Do Not Disturb* or *Make Up Request* status (as configured) or deactivate it to return to *Normal* status.

  Depending on the parameterisation and the current value of object, after a short press the following values will be transmitted.

Parameterisation	Current Object Value	Transmitted Value
Make Up Request	Do Not Disturb / Normal	Make Up Request
	Make Up Request	Normal
Do Not Disturb	Normal / Make Up Request	Do Not Disturb
	Do Not Disturb	Normal

Table 2. Room States.

- **Enumeration:** when the user touches the button, a 1-byte numeric value (unsigned) will be sent alternating between a list of up to 6 different values.

ETS PARAMETERISATION

When an individual button has been enabled, a specific tab ("x Control") becomes available under "Buttons" in the tree on the left.

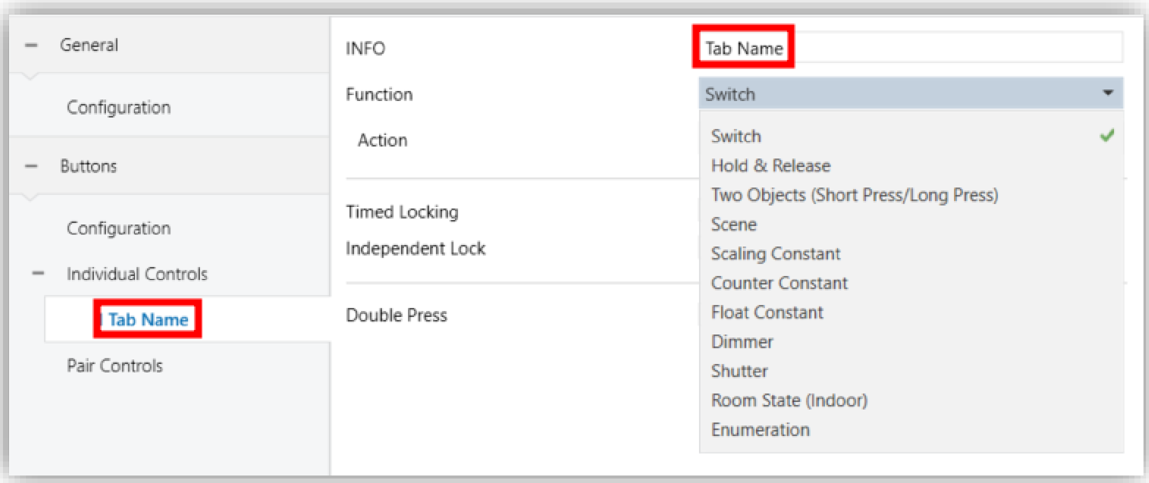


Figure 11. Individual Button.

Textbox **INFO** allows changing the default name of the tab in the left menu, as shows the following figure, identifying each of the enabled controls and their communication objects with the prefix **"[Ix][*INFO*]".**

The main parameter that needs to be configured is:

- **Function** [*Switch / Hold & Release / Two Objects (Short Press/Long Press) / Scene / Scaling Constant / Counter Constant / Float Constant / Dimmer / Shutter / Room State (Indoor) / Enumeration*]: sets the desired function for the button.

Depending on the function, some more parameters are involved (as described next). Please note that in the next pages "**[Ix][]**" is used as a general notation for the communication objects, where "x" depends on the particular individual button.

However, there is another parameter that appears for all previously selected functions except *Shutter*, as its use in this case is not useful.

- **Timed Locking** [*disabled / enabled*]: allows an individual button to be locked for a configured time after being pressed. When marked, a specific parameter appears:

  - **Lock Time** [*1…5…60*] [*s*]. It starts counting as soon as the button is released.

  After bus failure, the button starts without timed locking. It will start again when it is pressed again. This locking is independent of the general button locking, it has no influence on the locking object.

In case of enabling the **Independent Lock** parameter in the "Configuration" tab (see section 2.2.1), the homonymous parameter shall be displayed in the tab of each individual control.

- **Independent Lock** [*disabled / enabled*]: if selected, the new communication object (**"[Ix] Independent Locking"**) shall be appeared.

- **Double Press** [*disabled / enabled*]: offers the possibility to configure an additional action on the control for two consecutive short presses. If enabled, the following parameters will be displayed:

  - **Time Between Presses** [*1…5…50*] [*ds*]: allows you to configure the maximum waiting time between releasing the first press and performing the second.

  - **Sending Type** [*1-Bit Value / 1-Byte Value / Scene*]: configures the action to be performed in the event of a double press on the same push button.

    If the option "*1-Bit Value*" is set, the following parameter shall be displayed:

    - **Action** [*Send 0 / Send 1 / Toggle 0/1*]: value to be sent by the corresponding object (**"[Ix][] Double Press - Send: "0""**, **"[Ix][] Double Press - Send: "1""** or **"[Ix][] Double Press - Toggle: "0/1""**).

In case of selecting "*1-Byte Value*", the parameter shall be displayed:

- **Value** [*0…255*]: value to be sent through the object **"[Ix][] Double Press – Send: "1-Byte Value""**.

Finally, is "*Scene*" is parameterised:

- [*Run Scene / Save Scene*]: determines whether the double press triggers the scene or simply saves it.
- **Scene Number** [*1…64*]: allows you to select the scene number sent by the general scene object (**"[General] Scenes: Send"**).

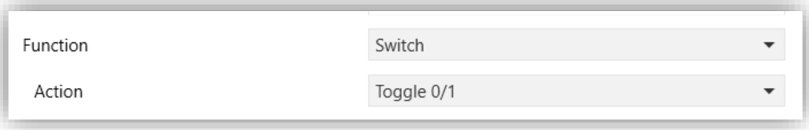
Notes:

➢ *If double press is enabled, it will not be possible to select the timed locking for that control, as both functionalities cannot coexist simultaneously.*

➢ *When this functionality is enabled, after a single short press, the object associated with the main action will be sent after the time between presses has elapsed.*

➢ *The double press parameter shall not be available in the individual controls "Hold & Release", "Shutter" and "Enumeration".*

2.2.2.1.1 Switch

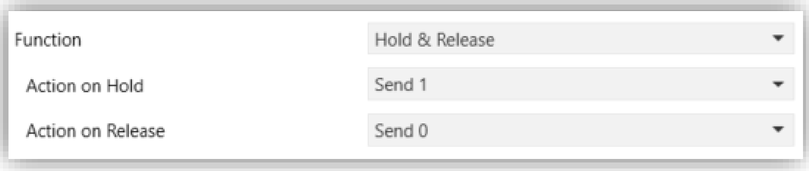


Function	Switch
Action	Toggle 0/1

Figure 12. Individual Button - Switch.

- **Action** [*Toggle 0/1 / Send 0 / Send 1*]: sets the value to be sent to the bus (through object **"[Ix][] Switch - Control: "X""**, where X is the parameterised action) when the user touches the button.

2.2.2.1.2 Hold & Release



Function	Hold & Release
Action on Hold	Send 1
Action on Release	Send 0

Figure 13. Individual Button - Hold & Release.

- **Action on Hold** [*Send 0 / Send 1 / Send 1-Byte Unsigned Value / Send 1-Byte Signed Value / Send 2-Byte Unsigned Value / Send 2-Byte Signed Value*]: sets the value to be sent to the bus (through "**[Ix][] Hold & Release - Switch Control**", "**[Ix][] Hold & Release - Control: "1-Byte Unsigned"**", "**[Ix][] Hold & Release - Control: "1-Byte Signed"**", "**[Ix][] Hold & Release - Control: "2-Byte Unsigned"**" or "**[Ix][] Hold & Release - Control: "2-Byte Signed"**" respectively) when the user touches the button.

  If an option different than 1 bit is configured, the **Value** parameter will be appeared, with its available range depending on Table 3.

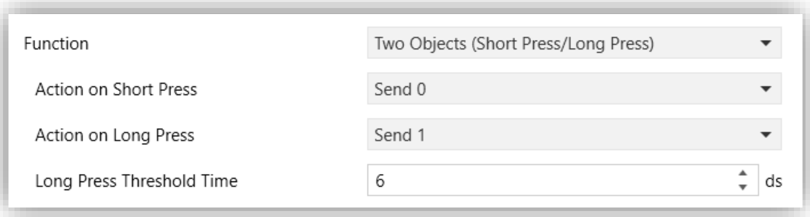
	1 byte	2 bytes
Unsigned	[*0…255*]	[*0…65535*]
Signed	[*-128…0…127*]	[*-32768…0…32767*]

Table 3. 1-Byte and 2-Byte value range.

- **Action on Release** [*Send 0 / Send 1 / Send 1-Byte Unsigned Value / Send 1-Byte Signed Value / Send 2-Byte Unsigned Value / Send 2-Byte Signed Value*]: sets the value to be sent to the bus (through the objects already mentioned in the action on hold) when the user stops touching the button.

  Again, if an option different than 1 bit is configured, the **Value** parameter will be appeared, with its available range depending on Table 3.

2.2.2.1.3 Two Objects (Short Press / Long Press)



Function: Two Objects (Short Press/Long Press)
Action on Short Press: Send 0
Action on Long Press: Send 1
Long Press Threshold Time: 6 ds

Figure 14. Individual Button - Two Objects (Short Press/Long Press).

- **Action on Short Press** [*Send 0 / Send 1 / Toggle 0/1 / Send 1-Byte Unsigned Value / Send 1-Byte Signed Value / Send 2-Byte Unsigned Value / Send 2-Byte Signed Value*]: sets the value to be sent to the bus (through "**[Ix][] Two Objects - Short Press Control: "X"**, where X is the parametrised action) when the user short-presses the button.

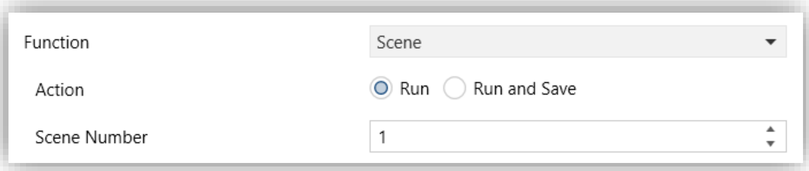
In case of selecting a non-binary option, an additional parameter **Value** will be displayed to enter the desired value whose available range will vary according to Table 3.

- **Action on Long Press** [*Send 0 / Send 1 / Toggle 0/1 / Send 1-Byte Unsigned Value / Send 1-Byte Signed Value / Send 2-Byte Unsigned Value / Send 2-Byte Signed Value*]: sets the value to be sent to the bus (through "**[In][] Two Objects - Long Press Control: "X"**, where X is the parametrised action) when the user long-presses the button.

  As with short press, in case of selecting a non-binary option, an additional parameter **Value** will be appeared, whose available range will vary according to Table 3.

- **Long Press Threshold Time** [*0…6…50*] [*ds*]: sets the minimum time the user should hold the button in order to consider it a long press.

2.2.2.1.4 Scene



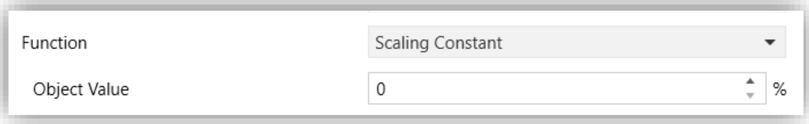
Function: Scene
Action: ● Run ○ Run and Save
Scene Number: 1

Figure 15. Individual Button - Scene.

- **Action** [*Run / Run and Save*]: 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 or –depending on the length of button press– a scene run or save request.

- **Scene Number** [*0…1…64*]: number of the scene to be sent to the bus, both in the case of the run requests and the save requests.

2.2.2.1.5 Scaling Constant / Counter Constant / Float Constant



Function: Scaling Constant
Object Value: 0 %

Figure 16. Individual Button - Scaling Constant.

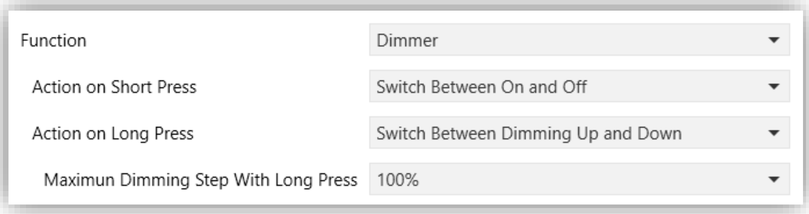
- **Object Value**: sets the value to be sent to the KNX bus when the user touches the button. The available range and the object through which the value is sent depend for each case, as the table below shows.

  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 or 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*] [*%*]	**[Ix][] Percentage - Control**
Counter Constant	[*0…255*] [*-128…0…127*] [*0…65535*] [*-32768…0…32767*]	**[Ix][] Integer - 1-Byte Unsigned Control** **[Ix][] Integer - 1-Byte Signed Control** **[Ix][] Integer - 2-Byte Unsigned Control** **[Ix][] Integer - 1-Byte Signed Control**
Float Constant	[*-671088,64…0…670433,28*]	**[Ix][] Float - 2-Byte Float**

Table 4. Constant type numerical control.

2.2.2.1.6 Dimmer



Function: Dimmer

Action on Short Press: Switch Between On and Off

Action on Long Press: Switch Between Dimming Up and Down

Maximun Dimming Step With Long Press: 100%

Figure 17. Individual Button - Dimmer.

Depending on which action is configured for short press, a different object is used. In the case of switching On/Off commands, they are sent via the one-bit object **"[Ix][] Light - Switch Control"**. If the sending of an absolute dimming is chosen, it is used the object **"[Ix][] Light - Absolute Dimming"**. However, to launch a scene, the object involved will be **"[General] Scene: Send"**.

As for the increment / decrement / stop on long press commands, they will be sent via the four-bit object "**[Ix][] Light - Dimming Control**".

On the other hand, the "**[Ix][] Light - Percentage Status**" 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).

The parameters for this function are:

- **Action on Short Press** [*Send Off / Send On / Switch Between On and Off / Absolute Dimming / Scene*]: sets the value to be sent to the bus (via the corresponding object) when the user makes a short press on the button.

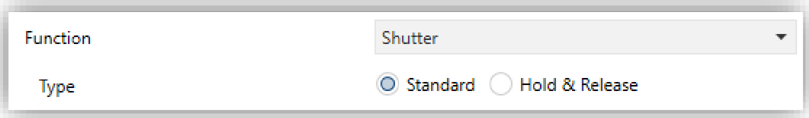
  If the fourth option is selected, an additional parameter appears (**Value** [*0...50...100*] [*%*]) to enter the desired percentage value. With the last option, another parameter called **Scene Number** appears with possible values between [*1...64*].

- **Action on Long Press** [*Dimming Down / Dimming Up / Switch Between Dimming Up and Down*]: sets the action to send to the bus when the user performs a long press on the button.

  - **Maximum Dimming Step With Long Press** [*100% / 50% / 25% / 12,5% / 6,25% / 3,1% / 1,5%*]: defines the dimming step to be sent (through "**[Ix][] Light - Dimming Control**") 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 push buttons send 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.*

2.2.2.1.7 Shutter



Function: Shutter

Type: ● Standard ○ Hold & Release

Figure 18. Individual Button - Shutter.

The (alternating) move up/down orders for Standard type will be sent through the "**[Ix][] Shutter - Move Control**" one-bit object, while the (alternating) step up/down orders will be through the "**[Btn] [Ix] Shutter - Stop/Step Control**" one-bit object. ". In case of

Hold/Release, the movement commands will be sent by the same object as standard but as there are no steps, the sending to stop the shutter will be done by the object "**[Ix][] Shutter - Stop Control**".

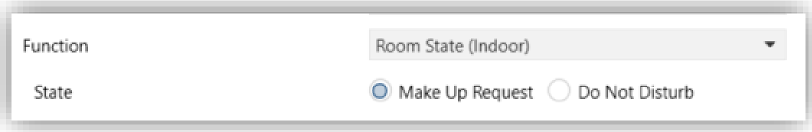
Additionally, a one-byte object ("**[Ix][] Shutter - Percentage Status**") is provided to link it to the position status object of the shutter actuator (in fact, this object is only intended to receive values from the bus, not to send them).

The parameters for this function are:

- **Type** [*Standard / Hold & Release*]: sets the desired control type.

2.2.2.1.8 Room State (Indoor)

When this function is assigned to the button, the object for the control "**[Ix][] Room State - Control**" is enabled. This object will also be a status indicator, as well as two other one-bit objects to indicate the status of the room individually ("**[Ix][] Room State - Make Up Room (Status)**" and "**[Ix][] Room Status - Do Not Disturb (Status)**").



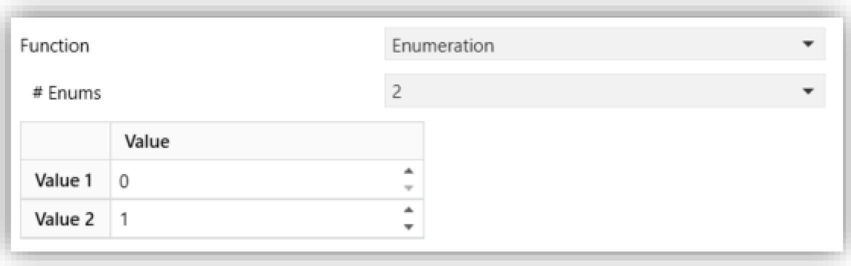
Function: Room State (Indoor)

State: ● Make Up Request ○ Do Not Disturb

Figure 19. Individual Button - Room state.

- **State** [*Make Up Request / Do Not Disturb*]: sets the state that is activated with this button. Commutes between "*Normal*" ("0") and the selected state: "*Make Up Room*" ("1") and "*Do Not Disturb*" ("2").

2.2.2.1.9 Enumeration



Function: Enumeration

Enums: 2

	Value
Value 1	0
Value 2	1

Figure 20. Enumeration.

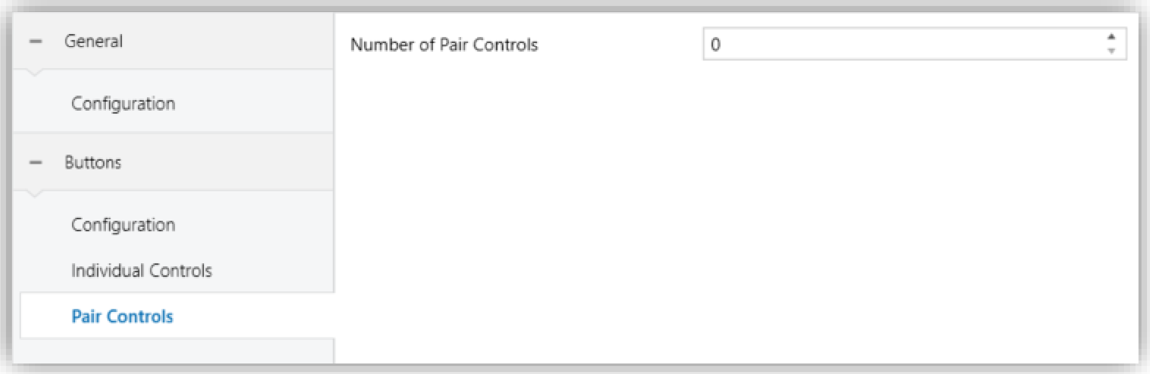
- **# Enums** [*1…2…6*]: determines the number of elements (N) in the enumeration.

- **Value N** [0…255]: value assigned to each enumerated element sent by the KNX bus when pressed (through the object **"[Ix][] Enumeration - Control"**). By default, each element will take a different value within the range from "0" to "N – 1".

2.2.3 PAIR CONTROLS

Up to two configurable pairs are available in the application program. Each pair can be configured independently and assigned to two different push buttons (see section 2.2.1).

ETS PARAMETERISATION



Number of Pair Controls 0

Figure 21. Buttons - Pair Controls.

- **Number of Pair Controls** [0…2]: selects the number of available pairs. For each pair, a new tab will be displayed in the left tree ("X Pair", see section 2.2.3.1).

2.2.3.1 X PAIR

Buttons configured to work as a joint control can be assigned the following functions:

- **Switch**: pressing one of the two buttons will make push buttons 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.

- **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.

- **Dimmer**: short-pressing one of the two buttons will make push buttons 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.

- **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.
- **Enumeration:** when either of the two buttons is pressed, an unsigned 1-byte value from a list of up to 6 elements will be sent. If the press is made on the left (or lower) button, the selection will decrease, whereas pressing the right (or upper) button will increase it.
- **Counter:** this function allows sending values of 1 byte or 2 bytes, with or without sign, and can sweep through the whole range of available values with short and long presses. Both the minimum and maximum value, as well as the increments for short and long presses, must be configured.
- **Room State (Indoor):** this function is very similar to its individual counterpart control, except that it allows the "*Make Up Room*" and "*Do Not Disturb*" states to be assigned to each button of the pair, respectively. The operation of the control is shown in Table 5.

Push button assigned to	Current Object Value	Transmitted Value
Make Up Room	Do Not Disturb / Normal	Make Up Room
	Make Up Room	Normal
Do Not Disturb	Normal / Make Up Room	Do Not Disturb
	Do Not Disturb	Normal

Table 5. Room States (Pair).

ETS PARAMETERISATION

Once two buttons have been assigned to a particular pair, a specific tab ("X Pair") becomes available under "Buttons" in the tab tree.

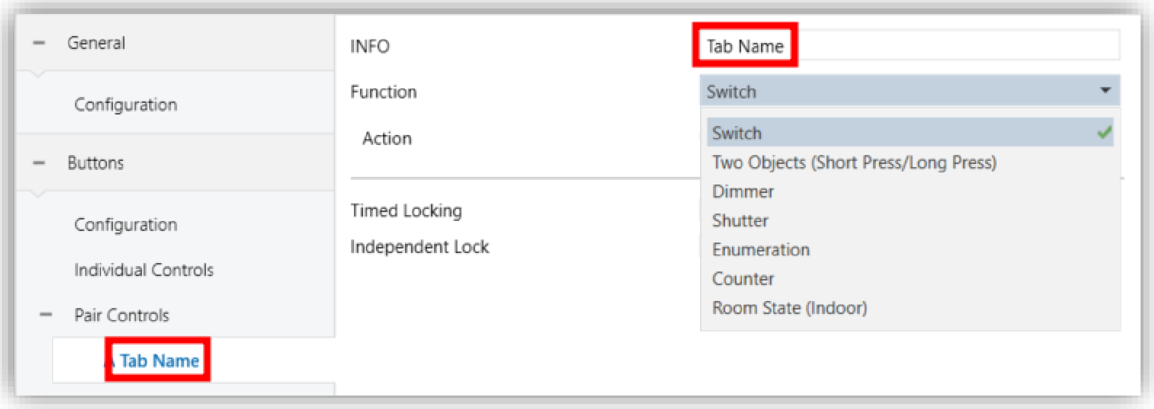


Figure 22. Button Pair.

Textbox **INFO** allows changing the default name of the tab in the left menu, as shows the following figure, identifying each of the enabled pairs and their communication objects with the prefix **"[PX][*INFO*]".**

The main parameter that needs to be configured is:

- **Function** [*Switch / Two Objects (Short Press/Long Press) / Dimmer / Shutter / Enumeration / Counter / Room State (Indoor)*]: sets the desired function for the button pair.

Depending on the function, some more parameters are shown, as described next. Please note that in the next pages the general notation "**[PX][]**" is used for the name of the communication objects, as "X" depends on the button pair (A or B).

However, there is another parameter that appears for all previously selected functions except "*Shutter*", as its use in this case is not useful.

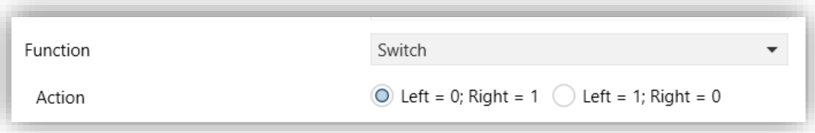
- **Timed Locking** [*disabled / enabled*]: allows a pair of buttons to be locked for a configured time after being pressed. When marked, a specific parameter appears **Lock Time** [*0…5…60*] [*s*]. It starts counting as soon as the button is released.

  After bus failure, the pair starts without timed locking. It will start again when it is pressed again. This locking is independent of the general button locking, it has no influence on the locking object.

In case of enabling the **Independent Lock** parameter in the "Configuration" tab (see section 2.2.1), the homonymous parameter shall be displayed in the tab of each pair control.

- **Independent Lock** [*disabled / enabled*]: if selected, the new communication object (**"[PX] Independent Locking"**) shall be appeared.

2.2.3.1.1 Switch



Function: Switch

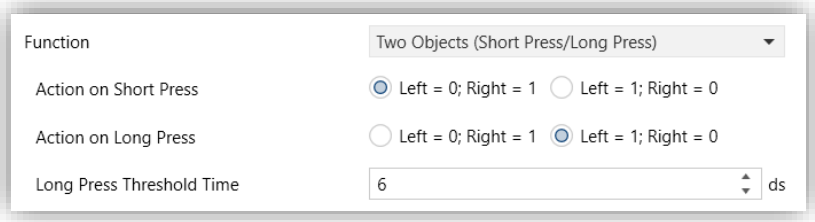
Action: ● Left = 0; Right = 1 ○ Left = 1; Right = 0

Figure 23. Pair Buttons - Switch.

- **Action** [*Left = 0; Right = 1 / Left = 1; Right = 0*]: assigns each of the two buttons the value to be sent through "**[PX][] Switch - Control**" (which has the Write flag enabled, so the state of the switch can be updated from external devices).

  Note: *this parameter will remain hidden unless having selected "Every button pair is configured separately" in **Action of the Pair Buttons (All Pairs)**.*

2.2.3.1.2 Two Objects (Short Press/Long press)



Function: Two Objects (Short Press/Long Press)

Action on Short Press: ● Left = 0; Right = 1 ○ Left = 1; Right = 0

Action on Long Press: ○ Left = 0; Right = 1 ● Left = 1; Right = 0

Long Press Threshold Time: 6 ds

Figure 24. Pair Buttons - Two Objects (Short Press/Long Press).

- **Action on Short Press** [*Left = 0; Right = 1 / Left = 1; Right = 0*]: sets the value that will be sent through "**[PX][] Two objects - Short Press Control**" after the user short-presses one of the two buttons.

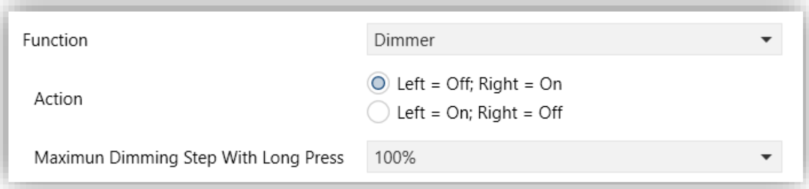
  Note: *this parameter will remain hidden unless having selected "Every button pair is configured separately" in **Action of the Pair Buttons (All Pairs)**.*

- **Action on Long Press** [*Left = 0; Right = 1 / Left = 1; Right = 0*]: sets the value that will be sent through "**[PX][] Two objects - Long Press Control**" after the user long-presses one of the two buttons.

Note: t*his parameter will remain hidden unless having selected "Every button pair is configured separately" in **Action of the Pair Buttons (All Pairs)**.*

- **Long Press Threshold Time** [*5…6…50*] [*ds*]: sets the minimum time the user should hold the button in order to consider it a long press.

2.2.3.1.3 Dimmer



Function: Dimmer

Action: ● Left = Off; Right = On ○ Left = On; Right = Off

Maximun Dimming Step With Long Press: 100%

Figure 25. Pair Buttons - Dimmer.

The switch orders will be sent through the "**[PX][] Light - Switch Control**" one-bit object, while the increase/decrease orders will be through the "**[PX][] Light - 4-bits Dimming Control**" four-bit object.

On the other hand, the "**[PX][] Light - Percentage Status**" 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).

The parameters for this function are:

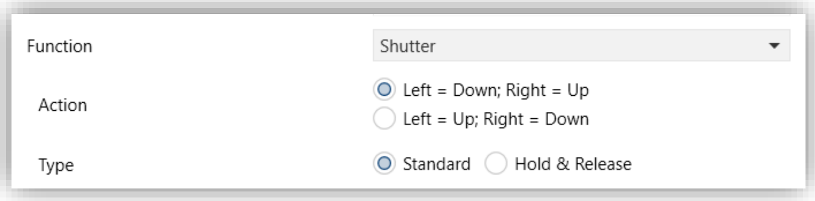
- **Action** [*Left = Off; Right = On / Left = On; Right = Off*]: assigns each of the two buttons the order to be sent.

  Note: *this parameter will remain hidden unless having selected "Every button pair is configured separately" in **Action of the Pair Buttons (All Pairs)**.*

- **Maximum Dimming Step With Long Press** [*100% / 50% / 25% / 12,5% / 6,25% / 3,1% / 1,5%*]: defines the dimming step 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 push buttons send 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.*

2.2.3.1.4 Shutter



Function: Shutter

Action: ● Left = Down; Right = Up ○ Left = Up; Right = Down

Type: ● Standard ○ Hold & Release

Figure 26. Pair Buttons - Shutter.

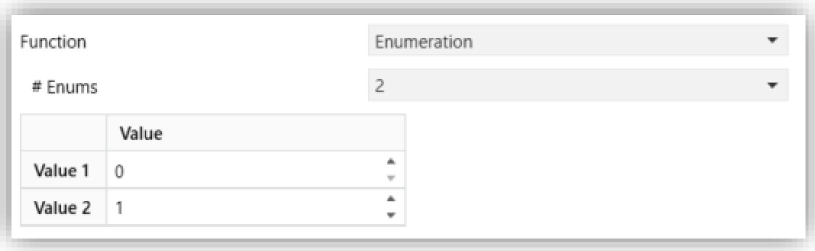
The move orders will be sent through “**[PX][] Shutter - Move Control**”, while the stop orders will be sent through “**[PX][] Shutter - Stop/Step Control**” (for Standard type) or “**[PX][] Shutter - Stop Control**” (for Hold & Release type). The parameters for this function are:

- **Action** [*Left = Down; Right = Up / Left = Up; Right = Down*]: assigns each of the two buttons the order to be sent.

  Note: *this parameter will remain hidden unless having selected “Every button pair is configured separately” in **Action of the Pair Buttons (All Pairs)**.*

- **Type** [*Standard / Hold & Release*]: sets the desired behaviour of the buttons.

2.2.3.1.5 Enumeration



Function: Enumeration

Enums: 2

	Value
Value 1	0
Value 2	1

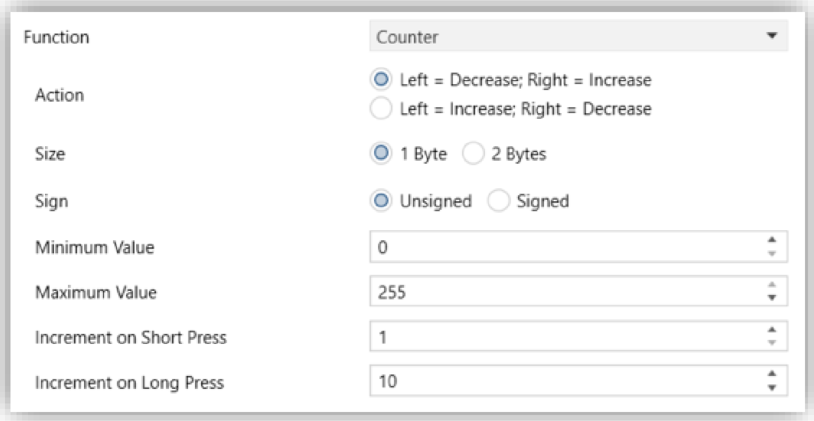
Figure 27. Pair Buttons - Enumeration.

- **# Enums** [*1…2…6*]: determines the number of elements (N) in the enumeration.

- **Value N** [*0…255*]: value assigned to each enumerated element sent by the KNX bus when pressed (through the object **“[Ix][] Enumeration - Control”**). By default, each element will take a different value within the range from “0” to “N – 1”.

Note: *this type of control does not allow the configuration of the action of each member of the pair.*

2.2.3.1.6 Counter



Function: Counter

Action: ● Left = Decrease; Right = Increase ○ Left = Increase; Right = Decrease

Size: ● 1 Byte ○ 2 Bytes

Sign: ● Unsigned ○ Signed

Minimum Value: 0

Maximum Value: 255

Increment on Short Press: 1

Increment on Long Press: 10

Figure 28. Pair Buttons - Counter.

If the "*Counter*" control is assigned to any pair of push buttons, button presses will cause a numerical value to be sent to the bus via the corresponding object (for more information, see Table 6). This value will increase or decrease as the buttons of the pair are pressed.

The available parameters for the configuration of this control are:

- **Action** [*Left = Decrease; Right = Increase / Left = Increase; Right = Decrease*]: sets which of the two buttons will be used to increment the current numeric value and which will be used to decrement it.

  Note: *this parameter will remain hidden unless having selected "Every button pair is configured separately" in **Action of the Pair Buttons (All Pairs)**.*

- **Size** [*1 Byte / 2 Bytes*]: size of the communication object.

- **Sign** [*Unsigned / Signed*]: sets whether the range allows negative or positive values only.

- **Minimum Value:** minimum value of any possible range that can be reached by successive presses of the decrease button.

- **Maximum Value:** maximum value of any possible range that can be reached by successive presses of the increase button.

- **Increment on Short Press:** increase or decrease that will take place with each short press on the respective buttons.

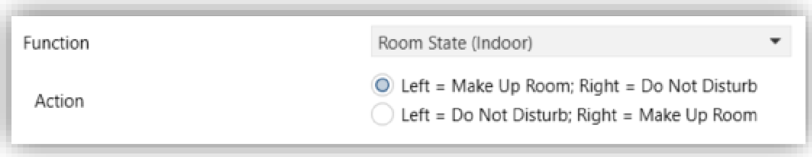
- **Increment on Long Press:** increase or decrease that will take place with each long press on the respective buttons.

Table 6 lists the functional ranges of the different data types, signed and unsigned, as well as the name of the associated control objects respectively.

	Available Values	Increment on Short Press	Increment on Long Press	Object Name
Counter	[*0…255*] [*-128…0…127*] [*0…65535*] [*-32768…0…32767*]	[*1…255*] [*1…127*] [*1…65535*] [*1…32767*]	[*10…255*] [*10…127*] [*10…65535*] [*10…32767*]	**"[PX][] Integer - 1-Byte Unsigned Control"** **"[PX][] Integer - 1-Byte Signed Control"** **"[PX][] Integer - 2-Byte Unsigned Control"** **"[PX][] Integer - 2-Byte Signed Control"**

Table 6. Ranges and names of "*Counter*" control objects (Pair).

2.2.3.1.7 Room State (Indoor)



Function: Room State (Indoor)

Action:
○ Left = Make Up Room; Right = Do Not Disturb (selected)
○ Left = Do Not Disturb; Right = Make Up Room

Figure 29. Pair Buttons - Room State (Indoor).

By assigning the function to a pair control, a control object **"[PX][] Room State - Control"** is enabled, which in turn will act as a status object, plus two other 1-bit objects to indicate the state of the room (**"[PX][] Room State - Make Up Room (Status)"** and **"[PX][] Room State - Do Not Distub (Status)"**).

- **Action** [*Left = Make Up Room; Right = Do Not Disturb / Left = Do Not Disturb; Right = Make Up Room*]: sets the status that is assigned to each push button in the pair. It will toggle between "*Normal*" ("0") and the chosen state: "*Make Up Room*" ("1") or "*Do Not Disturb*" ("2").

  Note: *this parameter will remain hidden unless having selected "Every button pair is configured separately" in **Action of the Pair Buttons (All Pairs)**.*

ANNEX I. 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 program itself.

Number	Size	I/O	Flags	Data type (DPT)	Functional Range	Name	Function
1	1 Bit	O	C R - T -	DPT_Trigger	0/1	[Heartbeat] Object to Send '1'	Sending of '1' Periodically
2	1 Bit	O	C R - T -	DPT_Trigger	0/1	[Heartbeat] Device Recovery	Send 0
3	1 Bit	O	C R - T -	DPT_Trigger	0/1	[Heartbeat] Device Recovery	Send 1
4	1 Byte	I	C - W - -	DPT_SceneNumber	0 - 63	[General] Scene: Receive	0 - 63 (Run Scene 1-64)
5	1 Byte	O	C - - T -	DPT_SceneControl	0-63; 128-191	[General] Scene: Send	0 - 63/128 - 191 (Run/Save Scene 1-64)
6	1 Bit	I	C - W - -	DPT_Enable	0/1	[General] Button Locking	0 = Unlock; 1 = Lock
	1 Bit	I	C - W - -	DPT_Enable	0/1	[General] Button Locking	0 = Lock; 1 = Unlock
7	1 Bit	O	C - - T -	DPT_Switch	0/1	[General] Welcome Back Object	Switch Object Sent on Wake Up
8, 9, 10, 11, 12	1 Bit	I	C - W - -	DPT_Switch	0/1	[General] Welcome Back Object - Additional Condition	Additional Condition Object x
13	1 Bit	I	C - W T -	DPT_Ack	0/1	[General] Cleaning Function	0 = Nothing; 1 = Clean Now
14, 24, 34, 44	1 Bit	I	C - W T -	DPT_Switch	0/1	[Ix][x] Switch - Control: “0”	1-Bit Generic Control
	1 Bit	I	C - W T -	DPT_Switch	0/1	[Ix][x] Hold & Release - Switch Control	1-Bit Generic Control
	1 Bit	I	C - W T -	DPT_Switch	0/1	[Ix][x] Two Objects - Short Press Control: "0"	1-Bit Generic Control
	1 Bit	I	C - W T -	DPT_Switch	0/1	[Ix][x] Light - Switch Control	(Short Press) Switch Between On and Off
	1 Bit	O	C - - T -	DPT_Trigger	0/1	[Ix][x] Shutter - Stop Control	(End Pressing) Stop Shutter
	1 Bit	O	C - - T -	DPT_Step	0/1	[Ix][x] Shutter - Stop/Step Control	(Short Press) 0 = Stop Shutter/Step Up; 1 = Stop Shutter/Step Down
	1 Bit	O	C R - T -	DPT_Switch	0/1	[Ix][x] Make Up In Progress	0 = Off; 1 = On
	1 Bit	I	C - W T U	DPT_Occupancy	0/1	[Ix][x] Presence - Occupancy State	0 = Not Occupied; 1 = Occupied
	1 Bit	I	C - W T -	DPT_Switch	0/1	[Ix][x] Switch - Control: “1”	1-Bit Generic Control
	1 Bit	I	C - W T -	DPT_Switch	0/1	[Ix][x] Switch - Control: “0/1”	1-Bit Generic Control
	1 Bit	I	C - W T -	DPT_Switch	0/1	[Ix][x] Two Objects - Short Press Control: "1"	1-Bit Generic Control
	1 Bit	I	C - W T -	DPT_Switch	0/1	[Ix][x] Two Objects - Short Press Control: "0/1"	1-Bit Generic Control
	1 Bit	I	C - W T -	DPT_Switch	0/1	[Ix][x] Light - On Control	(Short Press) Send On
	1 Bit	I	C - W T -	DPT_Switch	0/1	[Ix][x] Light - Off Control	(Short Press) Send Off

15, 25, 35, 45	4 Bit	I	C - W T -	DPT_Control_Dimming	0x0/0x8 (Stop) 0x1…0x7 (Dec.) 0x9…0xF (Inc.)	[Ix][x] Light - Dimming Control	(Long Press) Switch Between Dimming Up and Down
	4 Bit	I	C - W T -	DPT_Control_Dimming	0x0/0x8 (Stop) 0x1…0x7 (Dec.) 0x9…0xF (Inc.)	[Ix][x] Light - Dimming Control	(Long Press) Dimming Up
	4 Bit	I	C - W T -	DPT_Control_Dimming	0x0/0x8 (Stop) 0x1…0x7 (Dec.) 0x9…0xF (Inc.)	[Ix][x] Light - Dimming Control	(Long Press) Dimming Down
16, 26, 36, 46	1 Bit	O	C - - T -	DPT_UpDown	0/1	[Ix][x] Shutter - Move Control	(Start Pressing) Switch Between Up and Down
	1 Bit	I	C - W T -	DPT_Switch	0/1	[Ix][x] Two Objects - Long Press Control: "0"	1-Bit Generic Control
	1 Bit	O	C - - T -	DPT_UpDown	0/1	[Ix][x] Shutter - Move Control	(Long Press) 0 = Up; 1 = Down
	1 Bit	O	C R - T -	DPT_Bool	0/1	[Ix][x] Room State - Do Not Disturb (Status)	0 = No Active; 1 = Active
	1 Bit	I	C - W T -	DPT_Switch	0/1	[Ix][x] Two Objects - Long Press Control: "1"	1-Bit Generic Control
	1 Bit	I	C - W T -	DPT_Switch	0/1	[Ix][x] Two Objects - Long Press Control: "0/1"	1-Bit Generic Control
18, 28, 38, 48	1 Bit	O	C R - T -	DPT_Bool	0/1	[Ix][x] Room State - Make Up Room (Status)	0 = No Active; 1 = Active
19, 29, 39, 49	1 Byte	I	C - W T -	DPT_Scaling	0% - 100%	[Ix][x] Percentage - Control	0% … 100%
	1 Byte	I	C - W T -	DPT_Value_1_Ucount	0 - 255	[Ix][x] Integer - 1-Byte Unsigned Control	0 … 255
	1 Byte	I	C - W T -	DPT_Value_1_Count	-128 - 127	[Ix][x] Integer - 1-Byte Signed Control	-128 … 127
	2 Bytes	I	C - W T -	DPT_Value_2_Ucount	0 - 65535	[Ix][x] Integer - 2-Byte Unsigned Control	0 … 65535
	2 Bytes	I	C - W T -	DPT_Value_2_Count	-32768 - 32767	[Ix][x] Integer - 2-Byte Signed Control	-32768 … 32767
	2 Bytes	I	C - W T -	9.xxx	-671088.64 - 670433.28	[Ix][x] Float - 2-Byte Float	-671088.64 … 670433.28
	1 Byte	I	C - W T -	DPT_Value_1_Ucount	0 - 255	[Ix][x] Two Objects - Short Press Control: 1-Byte Unsigned	Send Selected 1-Byte Unsigned Value on Short Press
	1 Byte	I	C - W T -	DPT_Scaling	0% - 100%	[Ix][x] Shutter - Percentage Status	0% = Top; 100% = Bottom
	1 Byte	I	C - W T -	DPT_Scaling	0% - 100%	[Ix][x] Light - Percentage Status	0% - 100%
	1 Byte	I	C - W T -	DPT_Room_State	0 - 2	[Ix][x] Room State - Control	0 = Normal; 1 = Make Up Room; 2 = Do Not Disturb
	1 Byte	I/O	C R W T U	DPT_Room_State	0 - 2	[Ix][x] Room State	0 = Normal; 1 = Make Up Room; 2 = Do Not Disturb; 3 = Make Up in Progress
	1 Byte	I	C - W T -	DPT_Value_1_Ucount	0 - 255	[Ix][x] Enumeration - Control	Rotating value change
19, 21, 29, 31, 39, 41, 49, 51	1 Byte	I	C - W T -	DPT_Value_1_Count	-128 - 127	[Ix][x] Hold & Release - Control: "1-Byte Signed"	Send Selected 1-Byte Signed Value

	1 Byte	I	C - W T -	DPT_Value_1_Ucount	0 - 255	[Ix][x] Hold & Release - Control: "1-Byte Unsigned"	Send Selected 1-Byte Unsigned Value
	2 Bytes	I	C - W T -	DPT_Value_2_Count	-32768 - 32767	[Ix][x] Hold & Release - Control: "2-Byte Signed"	Send Selected 2-Byte Signed Value
	2 Bytes	I	C - W T -	DPT_Value_2_Ucount	0 - 65535	[Ix][x] Hold & Release - Control: "2-Byte Unsigned"	Send Selected 2-Byte Unsigned Value
19, 29, 39, 49	1 Byte	I	C - W T -	DPT_Value_1_Count	-128 - 127	[Ix][x] Two Objects - Short Press Control: 1-Byte Signed	Send Selected 1-Byte Signed Value on Short Press
	2 Bytes	I	C - W T -	DPT_Value_2_Ucount	0 - 65535	[Ix][x] Two Objects - Short Press Control: 2-Byte Unsigned	Send Selected 2-Byte Unsigned Value on Short Press
	2 Bytes	I	C - W T -	DPT_Value_2_Count	-32768 - 32767	[Ix][x] Two Objects - Short Press Control: 2-Byte Signed	Send Selected 2-Byte Signed Value on Short Press
20, 30, 40, 50	1 Byte	I	C - W T -	DPT_Scaling	0% - 100%	[Ix][x] Light - Absolute Dimming	(Short Press) Send Selected 1-Byte Value
21, 31, 41, 51	1 Byte	I	C - W T -	DPT_Value_1_Ucount	0 - 255	[Ix][x] Two Objects - Long Press Control: 1-Byte Unsigned	Send Selected 1-Byte Unsigned Value on Long Press
	1 Byte	I	C - W T -	DPT_Value_1_Count	-128 - 127	[Ix][x] Two Objects - Long Press Control: 1-Byte Signed	Send Selected 1-Byte Signed Value on Long Press
	2 Bytes	I	C - W T -	DPT_Value_2_Ucount	0 - 65535	[Ix][x] Two Objects - Long Press Control: 2-Byte Unsigned	Send Selected 2-Byte Unsigned Value on Long Press
	2 Bytes	I	C - W T -	DPT_Value_2_Count	-32768 - 32767	[Ix][x] Two Objects - Long Press Control: 2-Byte Signed	Send Selected 2-Byte Signed Value on Long Press
22, 32, 42, 52	1 Bit	I	C - W T -	DPT_Enable	0/1	[Ix] Independent Locking	0 = Unlock; 1 = Lock
	1 Bit	I	C - W T -	DPT_Enable	0/1	[Ix] Independent Locking	0 = Lock: 1 = Unlock
23, 33, 43, 53	1 Byte	I	C - W T -	DPT_Value_1_Ucount	0 - 255	[Ix][x] Double Press - Send: "1-Byte Value"	Send Selected 1-Byte Value
	1 Bit	I	C - W T -	DPT_Switch	0/1	[Ix][x] Double Press - Send: "1"	1-Bit Generic Control
	1 Bit	I	C - W T -	DPT_Switch	0/1	[Ix][x] Double Press - Toggle: "0/1"	1-Bit Generic Control
	1 Bit	I	C - W T -	DPT_Switch	0/1	[Ix][x] Double Press - Send: "0"	1-Bit Generic Control
114, 124	1 Bit	I	C - W T -	DPT_Switch	0/1	[Px][x] Switch - Control	1-Bit Generic Control
	1 Bit	I	C - W T -	DPT_Switch	0/1	[Px][x] Two Objects - Short Press Control	1-Bit Generic Control
	1 Bit	I	C - W T -	DPT_Switch	0/1	[Px][x] Light - Switch Control	0 = Off; 1 = On
	1 Bit	O	C - - T -	DPT_Step	0/1	[Px][x] Shutter - Stop/Step Control	0 = Stop/Step Up; 1 = Stop/Step Down
	1 Bit	O	C - - T -	DPT_Trigger	0/1	[Px][x] Shutter - Stop Control	0/1 = Stop
115, 125	4 Bit	I	C - W T -	DPT_Control_Dimming	0x0/0x8 (Stop) 0x1…0x7 (Dec.) 0x9…0xF (Inc.)	[Px][x] Light - 4-bits Dimming Control	4-Bit Dimming Control
116, 126	1 Bit	I	C - W T -	DPT_Switch	0/1	[Px][x] Two Objects - Long Press Control	1-Bit Generic Control
	1 Bit	O	C - - T -	DPT_UpDown	0/1	[Px][x] Shutter - Move Control	0 = Up; 1 = Down
	1 Bit	O	C R - T -	DPT_Bool	0/1	[Px][x] Room State - Do Not Disturb (Status)	0 = No Active; 1 = Active

118, 128	1 Bit	O	C R - T -	DPT_Bool	0/1	[Px][x] Room State - Make Up Room (Status)	0 = No Active; 1 = Active
119, 129	1 Byte	I/O	C R W T U	DPT_Room_State	0 - 2	[Px][x] Room State	0 = Normal; 1 = Make Up Room; 2 = Do Not Disturb
	1 Byte	I	C - W T -	DPT_Value_1_Ucount	0 - 255	[Px][x] Enumeration - Control	0 … 255
	1 Byte	I	C - W T -	DPT_Room_State	0 - 2	[Px][x] Room State - Control	0 = Normal; 1 = Make Up Room; 2 = Do Not Disturb
	2 Bytes	I	C - W T -	DPT_Value_2_Count	-32768 - 32767	[Px][x] Integer - 2-Byte Signed Control	-32768 … 32767
	1 Byte	I	C - W T -	DPT_Value_1_Ucount	0 - 255	[Px][x] Integer - 1-Byte Unsigned Control	0 … 255
	1 Byte	I	C - W T -	DPT_Value_1_Count	-128 - 127	[Px][x] Integer - 1-Byte Signed Control	-128 … 127
	2 Bytes	I	C - W T -	DPT_Value_2_Ucount	0 - 65535	[Px][x] Integer - 2-Byte Unsigned Control	0 … 65535
122, 132	1 Bit	I	C - W T -	DPT_Enable	0/1	[Px] Independent Locking	0 = Unlock; 1 = Lock
	1 Bit	I	C - W T -	DPT_Enable	0/1	[Px] Independent Locking	0 = Lock: 1 = Unlock
164	1 Byte	I	C - W - -	DPT_SceneControl	0-63; 128-191	[Thermostat] Scenes	0 – 63 (Execute 1 – 64); 128 – 191 (Save 1 – 64)
165	2 Bytes	I	C - W T U	DPT_Value_Temp	-273.00º - 670433.28º	[Tx] Temperature Source 1	External Sensor Temperature
166	2 Bytes	I	C - W T U	DPT_Value_Temp	-273.00º - 670433.28º	[Tx] Temperature Source 2	External Sensor Temperature
167	2 Bytes	O	C R - T -	DPT_Value_Temp	-273.00º - 670433.28º	[Tx] Effective Temperature	Effective Control Temperature
168	1 Byte	I	C - W - -	DPT_HVACMode	1=Comfort 2=Standby 3=Economy 4=Building Protection	[Tx] Special Mode	1-Byte HVAC Mode
169	1 Bit	I	C - W - -	DPT_Ack	0/1	[Tx] Special Mode: Comfort	0 = Nothing; 1 = Trigger
	1 Bit	I	C - W - -	DPT_Switch	0/1	[Tx] Special Mode: Comfort	0 = Off; 1 = On
170	1 Bit	I	C - W - -	DPT_Ack	0/1	[Tx] Special Mode: Standby	0 = Nothing; 1 = Trigger
	1 Bit	I	C - W - -	DPT_Switch	0/1	[Tx] Special Mode: Standby	0 = Off; 1 = On
171	1 Bit	I	C - W - -	DPT_Ack	0/1	[Tx] Special Mode: Economy	0 = Nothing; 1 = Trigger
	1 Bit	I	C - W - -	DPT_Switch	0/1	[Tx] Special Mode: Economy	0 = Off; 1 = On
172	1 Bit	I	C - W - -	DPT_Ack	0/1	[Tx] Special Mode: Protection	0 = Nothing; 1 = Trigger
	1 Bit	I	C - W - -	DPT_Switch	0/1	[Tx] Special Mode: Protection	0 = Off; 1 = On
173	1 Bit	I	C - W - -	DPT_Window_Door	0/1	[Tx] Window Status (Input)	0 = Closed; 1 = Open
174	1 Bit	I	C - W - -	DPT_Trigger	0/1	[Tx] Comfort Prolongation	0 = Nothing; 1 = Timed Comfort
175	1 Byte	O	C R - T -	DPT_HVACMode	1=Comfort 2=Standby 3=Economy 4=Building Protection	[Tx] Special Mode Status	1-Byte HVAC Mode
176	2 Bytes	I	C - W - -	DPT_Value_Temp	-273.00º - 670433.28º	[Tx] Setpoint	Thermostat Setpoint Input
	2 Bytes	I	C - W - -	DPT_Value_Temp	-273.00º - 670433.28º	[Tx] Basic Setpoint	Reference Setpoint

177	1 Bit	I	C - W - -	DPT_Step	0/1	[Tx] Setpoint Step	0 = Decrease Setpoint; 1 = Increase Setpoint
178	2 Bytes	I	C - W - -	DPT_Value_Tempd	-671088.64º - 670433.28º	[Tx] Setpoint Offset	Float Offset Value
179	2 Bytes	O	C R - T -	DPT_Value_Temp	-273.00º - 670433.28º	[Tx] Setpoint Status	Current Setpoint
180	2 Bytes	O	C R - T -	DPT_Value_Temp	-273.00º - 670433.28º	[Tx] Basic Setpoint Status	Current Basic Setpoint
181	2 Bytes	O	C R - T -	DPT_Value_Tempd	-671088.64º - 670433.28º	[Tx] Setpoint Offset Status	Current Setpoint Offset
182	1 Bit	I	C - W - -	DPT_Reset	0/1	[Tx] Setpoint Reset	Reset Setpoint to Default
	1 Bit	I	C - W - -	DPT_Reset	0/1	[Tx] Offset Reset	Reset Offset
183	1 Bit	I	C - W - -	DPT_Heat_Cool	0/1	[Tx] Mode	0 = Cool; 1 = Heat
184	1 Bit	O	C R - T -	DPT_Heat_Cool	0/1	[Tx] Mode Status	0 = Cool; 1 = Heat
185	1 Bit	I	C - W - -	DPT_Switch	0/1	[Tx] On/Off	0 = Off; 1 = On
186	1 Bit	O	C R - T -	DPT_Switch	0/1	[Tx] On/Off Status	0 = Off; 1 = On
187	1 Bit	I/O	C R W - -	DPT_Switch	0/1	[Tx] Main System (Cool)	0 = System 1; 1 = System 2
188	1 Bit	I/O	C R W - -	DPT_Switch	0/1	[Tx] Main System (Heat)	0 = System 1; 1 = System 2
189	1 Bit	I	C - W - -	DPT_Enable	0/1	[Tx] Enable/Disable Secondary System (Cool)	0 = Disable; 1 = Enable
190	1 Bit	I	C - W - -	DPT_Enable	0/1	[Tx] Enable/Disable Secondary System (Heat)	0 = Disable; 1 = Enable
191, 197	1 Byte	O	C R - T -	DPT_Scaling	0% - 100%	[Tx] [Sx] Control Variable (Cool)	PI Control (Continuous)
192, 198	1 Byte	O	C R - T -	DPT_Scaling	0% - 100%	[Tx] [Sx] Control Variable (Heat)	PI Control (Continuous)
	1 Byte	O	C R - T -	DPT_Scaling	0% - 100%	[Tx] [Sx] Control Variable	PI Control (Continuous)
193, 199	1 Bit	O	C R - T -	DPT_Switch	0/1	[Tx] [Sx] Control Variable (Cool)	2-Point Control
	1 Bit	O	C R - T -	DPT_Switch	0/1	[Tx] [Sx] Control Variable (Cool)	PI Control (PWM)
194, 200	1 Bit	O	C R - T -	DPT_Switch	0/1	[Tx] [Sx] Control Variable (Heat)	2-Point Control
	1 Bit	O	C R - T -	DPT_Switch	0/1	[Tx] [Sx] Control Variable (Heat)	PI Control (PWM)
	1 Bit	O	C R - T -	DPT_Switch	0/1	[Tx] [Sx] Control Variable	2-Point Control
	1 Bit	O	C R - T -	DPT_Switch	0/1	[Tx] [Sx] Control Variable	PI Control (PWM)
195, 201	1 Bit	O	C R - T -	DPT_Switch	0/1	[Tx] [Sx] PI State (Cool)	0 = PI Signal 0%; 1 = PI Signal Greater than 0%
196, 202	1 Bit	O	C R - T -	DPT_Switch	0/1	[Tx] [Sx] PI State (Heat)	0 = PI Signal 0%; 1 = PI Signal Greater than 0%
	1 Bit	O	C R - T -	DPT_Switch	0/1	[Tx] [Sx] PI State	0 = PI Signal 0%; 1 = PI Signal Greater than 0%
203	2 Bytes	O	C R - T -	DPT_Value_Temp	-273.00º - 670433.28º	[External Probe] Current Temperature	Temperature Sensor Value
204	1 Bit	O	C R - T -	DPT_Alarm	0/1	[External Probe] Overcooling	0 = No Alarm; 1 = Alarm
205	1 Bit	O	C R - T -	DPT_Alarm	0/1	[External Probe] Overheating	0 = No Alarm; 1 = Alarm
206	1 Bit	O	C R - T -	DPT_Alarm	0/1	[External Probe] Probe Error	0 = No Alarm; 1 = Alarm
207	2 Bytes	O	C R - T -	DPT_Value_Temp	-273.00º - 670433.28º	[Internal Probe] Current Temperature	Temperature Sensor Value
208	1 Bit	O	C R - T -	DPT_Alarm	0/1	[Internal Probe] Overcooling	0 = No Alarm; 1 = Alarm
209	1 Bit	O	C R - T -	DPT_Alarm	0/1	[Internal Probe] Overheating	0 = No Alarm; 1 = Alarm

Join and send us your inquiries
about Zennio devices:

https://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