# MAXinBOX SHUTTER 4CH / 8CH

## 4-Channel / 8-Channel Shutter Actuator

**ZIO-MBSHU4**
**ZIO-MBSHU8**

Application program version: [1.2]
User manual edition: [1.2]_a

www.zennio.com

# CONTENTS

# DOCUMENT UPDATES

| Version | Changes | Page(s) |
|---|---|---|
| [1.2]_a | **Changes in the application program**:<br>• Optimisation of the shutters and logic functions modules. | - |
| [1.1]_a | **Changes in the application program**:<br>• Heartbeat functionality added.<br>• Shutters module updated to version 0.3. | - |

# 1  INTRODUCTION

## 1.1  MAXINBOX SHUTTER 4CH / 8CH

**MAXinBOX SHUTTER 4CH** and **MAXinBOX SHUTTER 8CH** from Zennio are KNX specific actuators (of 4 or 8 channels, respectively) for controlling motorised shutter / blind systems.

The most outstanding features are:

- **8 / 16 relay outputs**, configurable as up to 4 / 8 independent shutter channels (with or without slats).

- **20** customisable, multi-operation **logic functions**.

- **Scene-triggered action control**, with an optional delay in the execution.

- **Manual operation / supervision** of the shutter channels through the on-board pushbuttons and LEDs.

- **Heartbeat** or periodic "still-alive" notification.

## 1.2  INSTALLATION

MAXinBOX SHUTTER 4CH / 8CH connects to the KNX bus through the on-board KNX connector.

Once the device is provided with power from the KNX bus, both the individual address and the associated application program may be downloaded.

This device does not need any additional external power since it is entirely powered through the KNX bus.
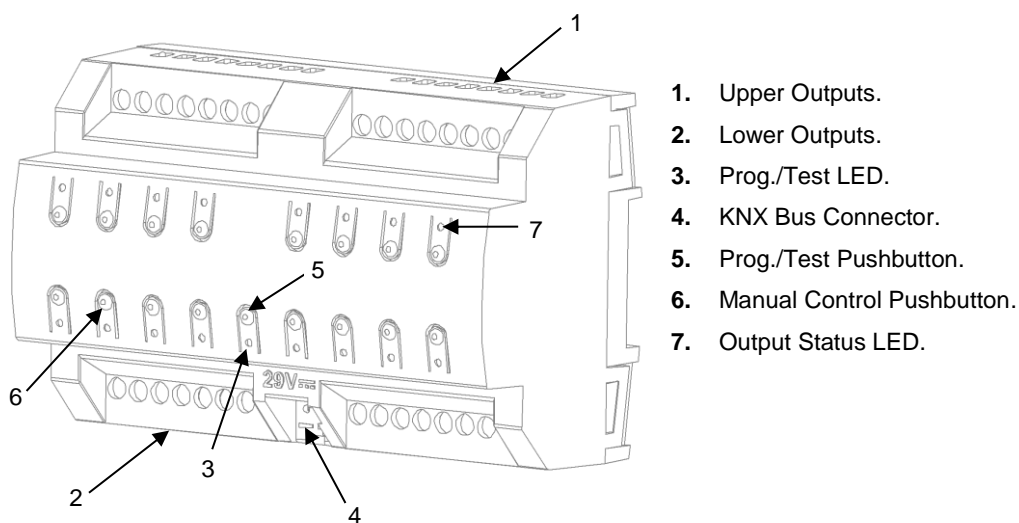


1. Upper Outputs.
2. Lower Outputs.
3. Prog./Test LED.
4. KNX Bus Connector.
5. Prog./Test Pushbutton.
6. Manual Control Pushbutton.
7. Output Status LED.

**Figure 1.** MAXinBOX SHUTTER 8CH

**Note**: *the above figure is entirely analogous for MAXinBOX SHUTTER 4CH.*

The main elements of the device are described next.

- **Test/Prog. Pushbutton (5)**: a short press on this button sets the device into the programming mode, making the associated LED (3) light in red.

  **Note**: *if this button is held while plugging the device into the KNX bus, the device will enter into **safe mode**. In such case, the LED will blink in red every 0.5 seconds.*

- **Outputs (1 and 2)**: output ports for the insertion of the stripped cables of the systems being controlled by the actuator (see section 2.2). Please secure the connection by means of the on-board screws.

- **Manual control pushbuttons (6)**: pushbuttons for a direct control of the shutter channels during the set-up process. See section 2.2.1.

To get detailed information about the technical features of this device, as well as on the installation process and on security procedures, please refer to the corresponding **Datasheet**, bundled with the original packaging of the device and also available at www.zennio.com.

## 1.3  START-UP AND POWER LOSS

During the start-up of the device, the Test/Prog. LED will blink in blue colour for a few seconds before the device is ready. External orders will not be executed during this time, but afterwards.

Depending on the configuration, some specific actions will also be performed during the start-up. For example, the integrator can set whether the shutter channels should change to a particular position and whether the device should send certain objects to the bus after the power recovery. Please consult the next sections of this document for further details.

On the other hand, when a bus power failure takes place, the device will interrupt any pending actions, and will save its state so it can be recovered once the power supply is restored.

For safety reasons, all **shutter channels** will be stopped (i.e., the relays will open) if a power loss takes place.

# 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.

### ETS PARAMETERISATION

The only parameterisable screen available by default is General. From this screen it is possible to activate/deactivate all the required functionality.
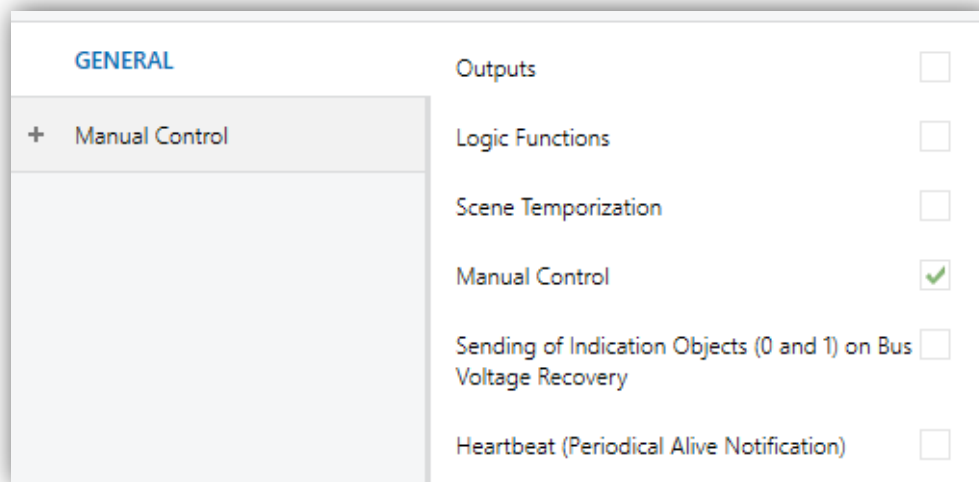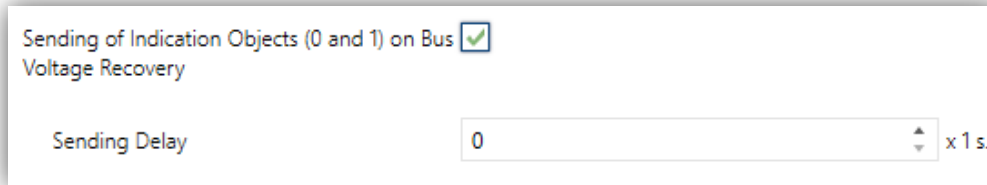


**Figure 2.** General screen

- Once activated, **Outputs, Logic Functions, Scene Temporisation** and **Manual Control** bring additional tabs to the menu on the left. These functions and their parameters will be explained in later sections of this document.

  The **Manual Control** function is enabled by default, and so is the corresponding configuration tab.

- **Sending of Indication Objects (0 and 1) on Bus Voltage Recovery**: this parameter lets the integrator activate two new communication objects ("**Reset 0**" and "**Reset 1**"), which will be sent to the KNX bus with values "0"

and "1" respectively whenever the device begins operation (for example, after a bus power failure). It is possible to parameterise a certain **delay** to this sending (0 to 255 seconds).



**Figure 3.** Sending of Indication objects on bus voltage recovery.

- **Heartbeat (Periodic Alive Notification)**: incorporates a one-bit object to the project ("**[Heartbeat] Object to Send '1'**") that will be sent periodically with a value of "1" to notify that the device is still working (*still alive*).



**Figure 4**. Heartbeat (Periodical Alive Notification).

**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 match the period set.*

## 2.2 OUTPUTS

MAXinBOX SHUTTER 4CH / 8CH incorporates **8 or 16 relay outputs**, respectively, configurable as up to **4 or 8 independent shutter channels**, each of which will operate one motorised shutter system.

For detailed information about the functionality and the configuration of the parameters related to the shutter channels, please refer to the specific manual "**Shutters**", available in the MAXinBOX SHUTTER 4CH / 8CH product section at the Zennio homepage (www.zennio.com).

### 2.2.1 MANUAL CONTROL

MAXinBOX SHUTTER 4CH / 8CH allows commanding orders through the pushbuttons on the top of the device to move the shutter up or down. Two specific pushbuttons are provided per channel (i.e., one per relay output).

Manual operation can be done in two different ways, named as **Test On Mode** (for testing purposes during the configuration of the device) and **Test Off Mode** (for a normal use, anytime). Whether both, only one, or none of these modes are available needs to be parameterised in ETS. Moreover, it is possible to enable a specific binary object for locking and unlocking the manual control in runtime.

**Note**:

- *The **Test Off mode** will be active (unless it has been disabled by parameter) after a download or a reset with no need of a specific activation – the pushbuttons will respond to user presses from the start.*

- *On the contrary, switching to the **Test On mode** (unless disabled by parameter) needs to be done by long-pressing the Prog/Test button (for at least three seconds), until the LED is no longer red and turns yellow. From that moment, once the button is released, the LED light will remain green to confirm that the device has switched from the Test Off mode to the Test On mode. After that, an additional press will turn the LED yellow and then off, once the button is released. This way, the device leaves the Test On mode. Note that it will also leave this mode if a bus power failure takes place.*

## Test Off Mode

Under the Test Off Mode, the shutter channels can be controlled through both their communication objects and the actual pushbuttons located on the top of the device.

When one of these buttons is pressed, the shutter will behave as if an order had been received through the corresponding communication object, and will also send the status objects when required.

This behaviour depends on the length of the button press:

- A **long press** makes the shutter start moving (upwards or downwards, depending on the button being pressed). The LED will light in green until the end of the motion. If the button gets pressed being the shutter already at the top or bottom positions, nothing will happen (the LED will not light).

- A **short press** will make the shutter drive stop (if in motion), as it normally does when a step/stop order is received from the KNX bus. In case of not being the shutter in motion, pressing the button does not cause any action, unless slats/lamellas have been parameterised – in such case, a step movement (up/down, depending on the button pressed) will take place.

Regarding the lock, timer, alarm and scene functions, the device will behave under the Test Off mode as usual. Button presses during this mode are entirely analogous to the reception of the corresponding orders from the KNX bus.

## Test On Mode

After entering the Test On mode, it will only be possible to control the shutters through the on-board pushbuttons. Orders received through communication objects will be ignored.

Pressing the button will make the shutter drive move upward or downward (depending on the button) until the button is released again, thus ignoring the position of the shutter and the parameterised times. The LED will light in green while the button is being hold.

For safety reasons, the device does not allow the activation of the two outputs of a shutter channel at the same time. If the button of one of the outputs is held while the other output is active, the device will first deactivate it and afterwards perform the required action on the output associated to the button pressed.

**Note**: *due to the actions that may have been performed during the Test On mode, the device will ignore the real status of the shutter when the user leaves such mode. Therefore, the Test On mode is useful to synchronise the device status and the actual status of the shutter..*

The lock, timer, alarm and scene functions as well as any orders received from the KNX bus will not have an effect over the shutter status while the device is under the Test On mode. The shutter status objects will not be sent to the bus, either.

On the contrary, the alarm and lock objects will be re-evaluated after leaving the Test On mode, so any changes that may have taken place in Test On will be considered when leaving.

**Important**: *the device is delivered from factory with both manual modes (Test Off and Test On) enabled by default.*

## ETS PARAMETERISATION

The **manual control** is configured from the Configuration tab, under Manual Control.

The only two parameters are:
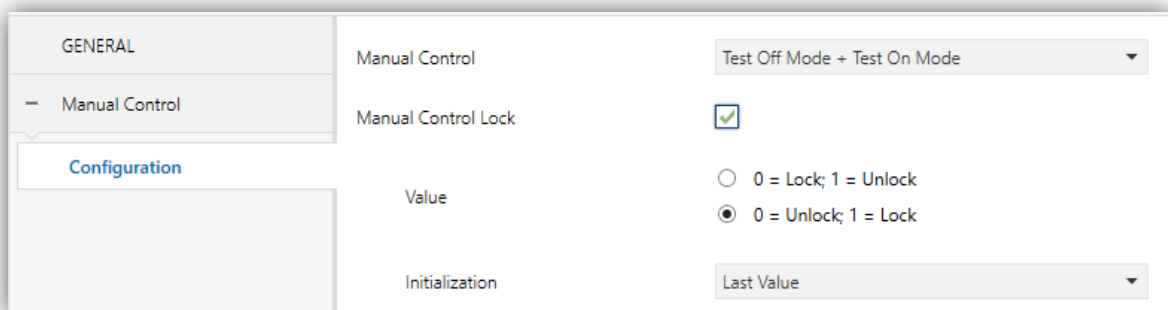


**Figure 5.** Manual Control

- **Manual Control**: options are "Disabled", "Only Test Mode Off", "Only Test Mode On" and "Both Test Mode Off and On" (default). Depending on the selection, the device will permit using the manual control under the Test Off, the Test On, or both modes. Note that, as stated before, using the Test Off mode does not require any special action, while switching to the Test On mode does require long-pressing the Prog/Test button.

- **Manual Control Lock**: unless the above parameter has been "<u>Disabled</u>", the Lock Manual Control parameter provides an optional procedure for locking the manual control in runtime. When this checkbox is enabled, object "**Manual Control Lock**" turns visible, as well as two more parameters:

  ➢ **Value**: defines whether the manual control lock/unlock should take place respectively upon the reception (through the aforementioned object) of values "0" and "1", or the opposite.

  ➢ **Initialization**: sets how the manual control should remain after the device start-up (after an ETS download or a bus power failure): "<u>Unlocked</u>", "<u>Locked</u>" or "<u>Last Value</u>" (default; on the very first start-up, this will be unlocked).

## 2.3  LOGIC FUNCTIONS

This module makes it possible to perform numeric and binary operations to incoming values received from the KNX bus, and to send the results through other communication objects specifically enabled for this purpose.

MAXinBOX SHUTTER 4CH / 8CH can implement **up to 20 different and independent functions**, each of them entirely customisable and consisting in **up to 4 consecutive operations each one**.

The execution of each function can depend on a configurable **condition**, which will be evaluated every time the function is **triggered** through specific, parameterisable communication objects. The result after executing the operations of the function can also be evaluated according to certain **conditions** and afterwards sent (or not) to the KNX bus, which can be done every time the function is executed, periodically or only when the result differs from the last one.

Please refer to the specific "**Logic Functions in MAXinBOX SHUTTER**" user manual (available in the MAXinBOX SHUTTER 4CH / 8CH product section at the Zennio homepage, www.zennio.com) for detailed information about the functionality and the configuration of the related parameters.

## 2.4  SCENE TEMPORISATION

The scene temporisation allows **imposing delays over the scenes** of the shutter channels. These delays, defined in parameters, are applied on the execution of one or more scenes that may have been configured.

Please bear in mind that, as multiple delayed scenes can be configured for each shutter channel, in case of receiving an order to execute one of them **when a previous temporisation is still pending** in that channel, the channel will interrupt such temporisation and will only execute the delay and the action of the new scene.

### ETS PARAMETERISATION

Prior to setting the **scene temporisation**, it is necessary to have one or more scenes configured in some of the channels. When entering the Configuration window under Scene Temporization, all configured scenes will be listed, together with a few checkboxes to select which of them need to be temporised, as shown in Figure 6.
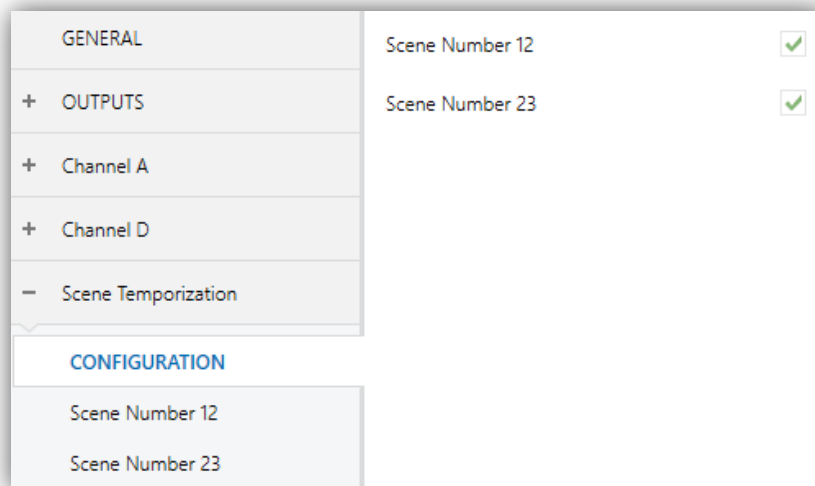


**Figure 6.** Scene Temporization

Enabling a certain scene number *n* brings a new tab with such name to the menu on the left, from which it is possible to configure the temporisation of that scene for each of the channels where it has been configured.
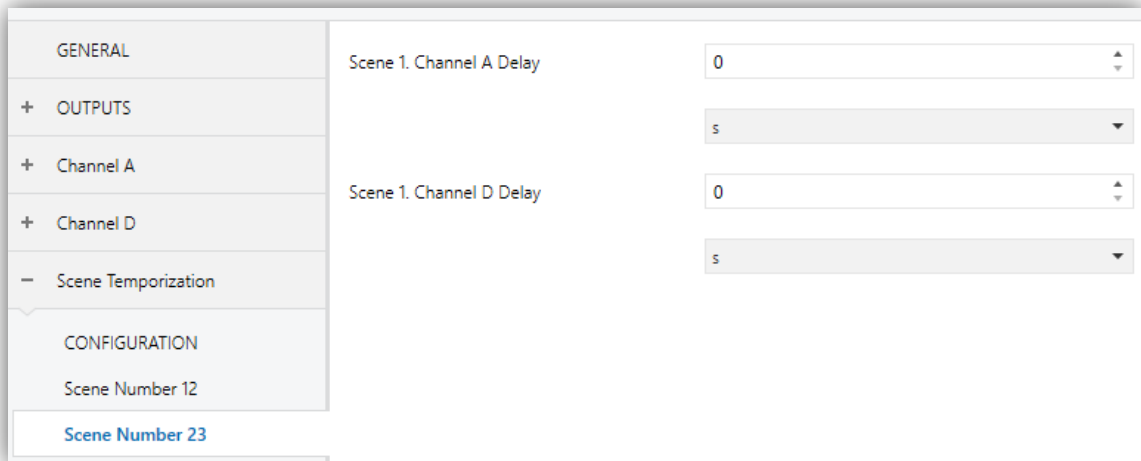
**Figure 7.** Configuration of Scene Temporization

Therefore, parameter "**Scene m. Channel z Delay**" defines the delay that will be applied to the action defined in channel z for the execution of scene m. The range of this delay is 0 to 3600 seconds, 0 to 1440 minutes or 0 to 24 hours.

# 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.

(*) Only applies to MAXinBOX SHUTTER 8CH

| Number | Size | I/O | Flags | Data type (DPT) | Functional Range | Name | Function |
|---|---|---|---|---|---|---|---|
| 1 | 1 Bit | | C T - - - | DPT_Trigger | 0/1 | [Heartbeat] Object to Send '1' | Sending of '1' Periodically |
| 2 | 1 Bit | | C T - - - | DPT_Trigger | 0/1 | Reset 0 | Voltage Recovery -> Sending of 0 |
| 3 | 1 Bit | | C T - - - | DPT_Trigger | 0/1 | Reset 1 | Voltage Recovery -> Sending of 1 |
| 4 | 1 Bit | I | C - - W - | DPT_Enable | 0/1 | Lock Manual Control | 0 = Lock; 1 = Unlock |
| | 1 Bit | I | C - - W - | DPT_Enable | 0/1 | Lock Manual Control | 0 = Unlock; 1 = Lock |
| 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36 | 1 Bit | I | C - - W - | DPT_Bool | 0/1 | [LF] (1-Bit) Data Entry x | Binary Data Entry (0/1) |
| 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52 | 1 Byte | I | C - - W - | DPT_Value_1_Ucount | 0 - 255 | [LF] (1-Byte) Data Entry x | 1-Byte Data Entry (0-255) |
| 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68 | 2 Bytes | I | C - - W - | DPT_Value_2_Ucount | 0 - 65535 | [LF] (2-Byte) Data Entry x | 2-Byte Data Entry |
| | | | | DPT_Value_2_Count | -32768 - 32767 | | |
| | | | | DPT_Value_Temp | -273,00 - 670760,00 | | |
| 69, 70, 71, 72, 73, 74, 75, 76 | 4 Bytes | I | C - - W - | DPT_Value_4_Count | -2147483648 - 2147483647 | [LF] (4-Byte) Data Entry x | 4-Byte Data Entry |
| 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96 | 1 Bit | O | C T R - - | DPT_Bool | 0/1 | [LF] Function x - Result | (1-Bit) Boolean |
| | 1 Byte | O | C T R - - | DPT_Value_1_Ucount | 0 - 255 | [LF] Function x - Result | (1-Byte) Unsigned |
| | 2 Bytes | O | C T R - - | DPT_Value_2_Ucount | 0 - 65535 | [LF] Function x - Result | (2-Byte) Unsigned |
| | 4 Bytes | O | C T R - - | DPT_Value_4_Count | -2147483648 - 2147483647 | [LF] Function x - Result | (4-Byte) Signed |
| | 1 Byte | O | C T R - - | DPT_Scaling | 0% - 100% | [LF] Function x - Result | (1-Byte) Percentage |
| | 2 Bytes | O | C T R - - | DPT_Value_2_Count | -32768 - 32767 | [LF] Function x - Result | (2-Byte) Signed |
| | 2 Bytes | O | C T R - - | DPT_Value_Temp | -273,00 - 670760,00 | [LF] Function x - Result | (2-Byte) Float |
| 97 | 1 Byte | I | C - - W - | DPT_SceneControl | 0-63; 128-191 | [Shutter] Scenes | 0 - 63 (Execute 1 - 64); 128 - 191 (Save 1 - 64) |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 98, 126, 154, 182, 210(*), 238(*), 266(*), 294(*) | 1 Bit | I | C - - W - | DPT_UpDown | 0/1 | [Cx] Move | 0=Raise; 1=Lower |
| 99, 127, 155, 183, 211(*), 239(*), 267(*), 295(*) | 1 Bit | I | C - - W - | DPT_Step | 0/1 | [Cx] Stop/Step | 0=Stop/StepUp; 1=Stop/StepDown |
| | 1 Bit | I | C - - W - | DPT_Trigger | 0/1 | [Cx] Stop | 0=Stop; 1=Stop |
| 100, 128, 156, 184, 212(*), 240(*), 268(*), 296(*) | 1 Bit | I | C - - W - | DPT_Enable | 0/1 | [Cx] Lock | 0=Unlock; 1=Lock |
| 101, 129, 157, 185, 213(*), 241(*), 269(*), 297(*) | 1 Byte | I | C - - W - | DPT_Scaling | 0% - 100% | [Cx] Shutter Positioning | 0%=Top; 100%=Bottom |
| 102, 130, 158, 186, 214(*), 242(*), 270(*), 298(*) | 1 Byte | O | C T R - - | DPT_Scaling | 0% - 100% | [Cx] Shutter Position (Status) | 0%=Top; 100%=Bottom |
| 103, 131, 159, 187, 215(*), 243(*), 271(*), 299(*) | 1 Byte | I | C - - W - | DPT_Scaling | 0% - 100% | [Cx] Slats Positioning | 0%=Open; 100%=Closed |
| 104, 132, 160, 188, 216(*), 244(*), 272(*), 300(*) | 1 Byte | O | C T R - - | DPT_Scaling | 0% - 100% | [Cx] Slats Position (Status) | 0%=Open; 100%=Closed |
| 105, 133, 161, 189, 217(*), 245(*), 273(*), 301(*) | 1 Bit | O | C T R - - | DPT_Switch | 0/1 | [Cx] Rising Relay (Status) | 0=Open; 1=Closed |
| 106, 134, 162, 190, 218(*), 246(*), 274(*), 302(*) | 1 Bit | O | C T R - - | DPT_Switch | 0/1 | [Cx] Lowering Relay (Status) | 0=Open; 1=Closed |
| 107, 135, 163, 191, 219(*), 247(*), 275(*), 303(*) | 1 Bit | O | C T R - - | DPT_Switch | 0/1 | [Cx] Movement (Status) | 0=Stopped; 1=Moving |
| 108, 136, 164, 192, 220(*), 248(*), 276(*), 304(*) | 1 Bit | O | C T R - - | DPT_UpDown | 0/1 | [Cx] Movement Direction (Status) | 0=Upward; 1=Downward |
| 109, 137, 165, 193, 221(*), 249(*), 277(*), 305(*) | 1 Bit | I | C - - W - | DPT_Switch | 0/1 | [Cx] Auto: On/Off | 0=On; 1=Off |
| | 1 Bit | I | C - - W - | DPT_Switch | 0/1 | [Cx] Auto: On/Off | 0=Off; 1=On |
| 110, 138, 166, 194, 222(*), 250(*), 278(*), 306(*) | 1 Bit | O | C T R - - | DPT_Switch | 0/1 | [Cx] Auto: On/Off (Status) | 0=On; 1=Off |
| | 1 Bit | O | C T R - - | DPT_Switch | 0/1 | [Cx] Auto: On/Off (Status) | 0=Off; 1=On |
| 111, 139, 167, 195, 223(*), 251(*), 279(*), 307(*) | 1 Bit | I | C - - W - | DPT_UpDown | 0/1 | [Cx] Auto: Move | 0=Raise; 1=Lower |
| 112, 140, 168, 196, 224(*), 252(*), 280(*), 308(*) | 1 Bit | I | C - - W - | DPT_Step | 0/1 | [Cx] Auto: Stop/Step | 0=Stop/StepUp; 1=Stop/StepDown |
| | 1 Bit | I | C - - W - | DPT_Step | 0/1 | [Cx] Auto: Stop | 0=Stop; 1=Stop |
| 113, 141, 169, 197, 225(*), 253(*), 281(*), 309(*) | 1 Byte | I | C - - W - | DPT_Scaling | 0% - 100% | [Cx] Auto: Shutter Positioning | 0%=Top; 100%=Bottom |
| 114, 142, 170, 198, 226(*), 254(*), 282(*), 310(*) | 1 Byte | I | C - - W - | DPT_Scaling | 0% - 100% | [Cx] Auto: Slats Positioning | 0%=Open; 100%=Closed |
| 115, 143, 171, 199, 227(*), 255(*), 283(*), 311(*) | 1 Bit | I | C T - W U | DPT_Scene_AB | 0/1 | [Cx] Sunshine/Shadow | 0=Sunshine; 1=Shadow |
| | 1 Bit | I | C T - W U | DPT_Scene_AB | 0/1 | [Cx] Sunshine/Shadow | 0=Shadow; 1=Sunshine |
| 116, 144, 172, 200, 228(*), 256(*), 284(*), 312(*) | 1 Bit | I | C T - W U | DPT_Heat_Cool | 0/1 | [Cx] Cooling/Heating | 0=Heating; 1=Cooling |
| | 1 Bit | I | C T - W U | DPT_Heat_Cool | 0/1 | [Cx] Cooling/Heating | 0=Cooling; 1=Heating |
| 117, 145, 173, 201, 229(*), | 1 Bit | I | C T - W U | DPT_Occupancy | 0/1 | [Cx] Presence/No Presence | 0=Presence; 1=No Presence |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 257(*), 285(*), 313(*) | 1 Bit | I | C T - W U | DPT_Occupancy | 0/1 | [Cx] Presence/No Presence | 0=No Presence; 1=Presence |
| 118, 146, 174, 202, 230(*), 258(*), 286(*), 314(*) | 1 Bit | I | C - - W - | DPT_Alarm | 0/1 | [Cx] Alarm 1 | 0=No Alarm; 1=Alarm |
| | 1 Bit | I | C - - W - | DPT_Alarm | 0/1 | [Cx] Alarm 1 | 0=Alarm; 1=No Alarm |
| 119, 147, 175, 203, 231(*), 259(*), 287(*), 315(*) | 1 Bit | I | C - - W - | DPT_Alarm | 0/1 | [Cx] Alarm 2 | 0=No Alarm; 1=Alarm |
| | 1 Bit | I | C - - W - | DPT_Alarm | 0/1 | [Cx] Alarm 2 | 0=Alarm; 1=No Alarm |
| 120, 148, 176, 204, 232(*), 260(*), 288(*), 316(*) | 1 Bit | I | C - - W - | DPT_Ack | 0/1 | [Cx] Unfreeze Alarm | Alarm1=Alarm2=No Alarm + Unfreeze (1) => End Alarm |
| 121, 149, 177, 205, 233(*), 261(*), 289(*), 317(*) | 1 Bit | I | C - - W - | DPT_Scene_AB | 0/1 | [Cx] Move (Reversed) | 0=Lower; 1=Raise |
| 122, 150, 178, 206, 234(*), 262(*), 290(*), 318(*) | 1 Bit | I | C - - W - | DPT_Ack | 0/1 | [Cx] Direct Positioning 1 | 0=No Action; 1=Go to Position |
| 123, 151, 179, 207, 235(*), 263(*), 291(*), 319(*) | 1 Bit | I | C - - W - | DPT_Ack | 0/1 | [Cx] Direct Positioning 2 | 0=No Action; 1=Go to Position |
| 124, 152, 180, 208, 236(*), 264(*), 292(*), 320(*) | 1 Bit | I | C - - W - | DPT_Ack | 0/1 | [Cx] Direct Positioning 1 (Save) | 0=No Action; 1=Save Current Position |
| 125, 153, 181, 209, 237(*), 265(*), 293(*), 321(*) | 1 Bit | I | C - - W - | DPT_Ack | 0/1 | [Cx] Direct Positioning 2 (Save) | 0=No Action; 1=Save Current Position |

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*

KNX    C E    RoHS