# MINiBOX 20

## 2-Output Multifunction Actuator

**ZIO-MN20**

Application programme version: [1.3]
User manual edition: [1.3]_a

www.zennio.com

# CONTENTS

# DOCUMENT UPDATES

| Version | Changes | Page(s) |
|---------|---------|---------|
| [1.3]_a | **Changes in the application program**:<br><br>• Optimisation of the individual outputs, logic functions, shutter channel y heartbeat modules. | - |
|  | Minor text changes. | - |

# 1. INTRODUCTION

## 1.1  MINiBOX 20

MINiBOX 20 from Zennio is a versatile and small-size KNX actuator featuring a wide variety of functions and four relay outputs for a variety of applications, such as controlling shutter drives.

The most outstanding features are:

- **2 relay outputs**, configurable as:

  - 1 shutter channels (with or without slats), or

  - Up to 2 individual ON/OFF outputs,

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

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

- **Manual operation / supervision** of the 2 relay outputs through the on-board pushbuttons and LEDs.

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

## 1.2  INSTALLATION

MINiBOX 20 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 programme may be downloaded.

This device does not need any additional external power since it is entirely powered through the KNX bus.
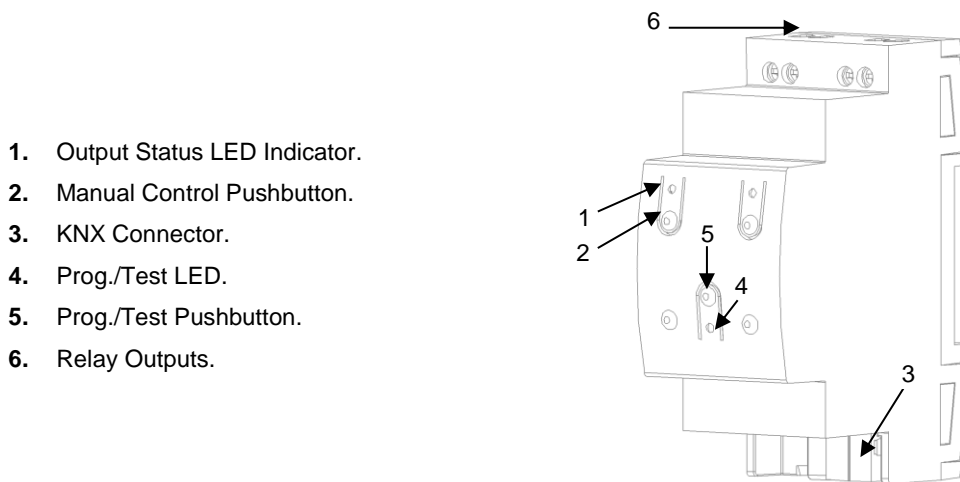
**1.** Output Status LED Indicator.
**2.** Manual Control Pushbutton.
**3.** KNX Connector.
**4.** Prog./Test LED.
**5.** Prog./Test Pushbutton.
**6.** Relay Outputs.

**Figure 1.** MINiBOX 20. Elements.

The main elements of the device are described next.

- **Prog./Test Pushbutton (5)**: a short press on this button sets the device into the programming mode, making the associated LED (5) 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 (6)**: 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.

To get detailed information about the technical features of this device, as well as on the installation and security procedures, please refer to the corresponding **Datasheet**, bundled with the original package 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 Prog./Test 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 output channels should switch to a particular state and whether the device should send certain objects to the bus after the power recovery. Please refer to 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, while the individual outputs will switch to the specific state configured in ETS (if any).

# 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

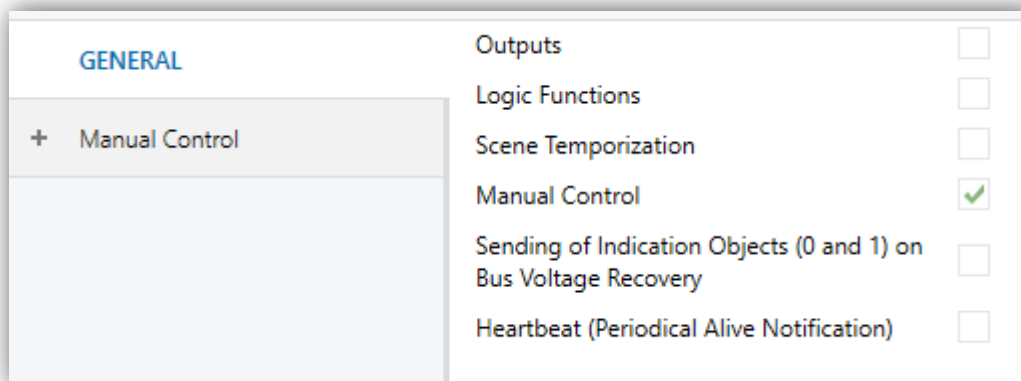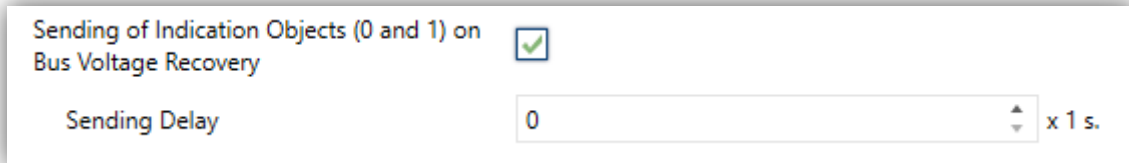From the General parameter screen it is possible to activate/deactivate all the required functionality.



**Figure 2.** General

- 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
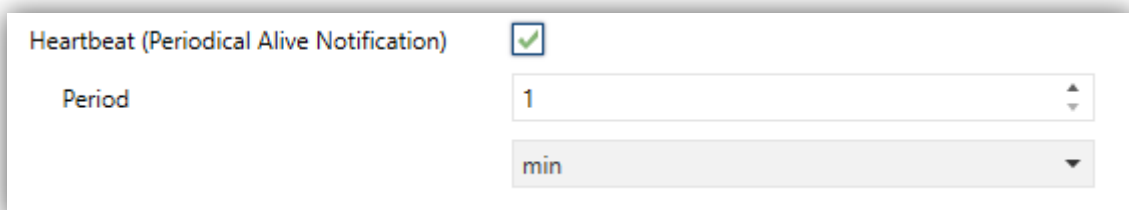
power failure). It is possible to parameterise a certain **delay** to this sending (0 to 255 seconds).



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

- **Heartbeat (Periodical 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

MINiBOX 20 incorporates **2 relay outputs**, configurable as:

- **Individual binary outputs**, which allows an independent control of loads.

- **One shutter channel**, which allow controlling the motion of shutters or blinds.

For detailed information about the functionality and the configuration of the related parameters, please refer to the following specific manuals, all of them available in the MINiBOX 20 product section at the Zennio homepage (www.zennio.com):

- **Individual outputs.**

- **Shutters.**

### 2.2.1  MANUAL CONTROL

MINiBOX 20 allows manually switching the state of its output relays through the respective pushbuttons on the top of the device. A specific pushbutton is therefore available per 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 can be accessed 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 outputs 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 output will behave as if an order had been received through the corresponding communication object, depending on whether the output is configured as an individual output or as a shutter channel.

- **Individual output**: a simple press (short or long) will make the output switch its on-off state, which will be reported to the KNX bus through the corresponding status object, if enabled.

- **Shutter Channel**: when the button is pressed, the device will act over the output according to the length of the button press and to the current state.

  - ➢ 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 parameterized – in such case, a step movement (up/down, depending on the button pressed) will take place. The status objects will be sent to the bus when corresponding.

- **Disabled output**: outputs disabled by parameter will not react to button presses under the Test Off mode.

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 outputs through the on-board pushbuttons. Orders received through communication objects will be ignored, with independence of the channel or the output they are addressed to.

Depending on whether the output has been parameterised as an individual output or as part of a shutter channel, the reactions to the button presses will differ.

- **Individual output**: short or long pressing the button will commute the on-off state of the relay.

- **Shutter channel**: 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.

  **Note**: *after leaving the Test On mode, the status objects will recover the values they had prior to entering Test On. As the device is never aware of the actual position of the shutter (as the shutter drive does not provide any feedback), these values may not show the real position. This can be solved by performing a complete move-up or move-down order, or by calibrating the shutter position in the Test On mode until it matches the status objects.*

- **Disabled output**: short and long presses will switch the state of the corresponding relay. In case this consists in closing the relay, then the remaining relays will open, for safety reasons.

The lock, timer, alarm and scene functions will not work while the device is under the Test On mode. Status objects will not be sent to the bus, either.

> **Important**: *the device is delivered from factory with all the output channels configured as disabled outputs, and with both manual modes (Test Off and Test On) enabled.*

## ETS PARAMETERISATION

The **Manual Control** is configured from the Configuration tab itself under Manual Control.
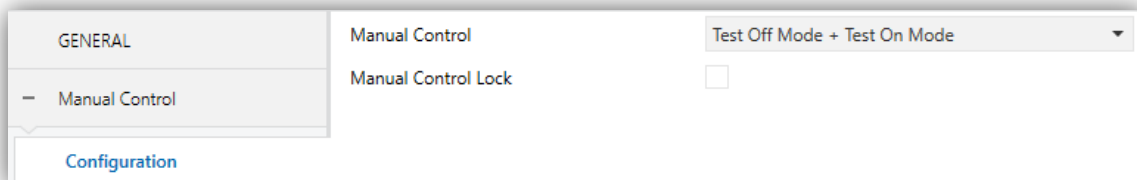
The only two parameters are:



**Figure 5.** Manual control screen

- **Manual Control**: options are "Disabled", "Only Test Mode Off", "Only Test Mode On" and "Test Mode Off + Test On Mode" (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.

- **Lock Manual Control**: unless the above parameter has been "Disabled", 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.

  - **Initialisation**: sets how the lock state of the manual control should remain after the device start-up (after an ETS download or a bus power failure): "Unlocked", "Locked" or "Last Value" (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.

MINiBOX 20 can implement **up to 10 different and independent functions**, each of them entirely customisable and consisting in **up to 4 consecutive operations each**.

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 "**Logic Functions**" user manual, available within the MINiBOX 20 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 outputs**. These delays are defined in parameters, and can be applied to 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 individual output / shutter channel, in case of receiving an order to execute one of them when a previous temporisation is still pending for that output / channel, such temporisation will be interrupted and only the delay and the action of the new scene will be executed.

### ETS PARAMETERISATION

Prior to setting the **scene temporisation**, it is necessary to have one or more scenes configured in some of the outputs. 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.
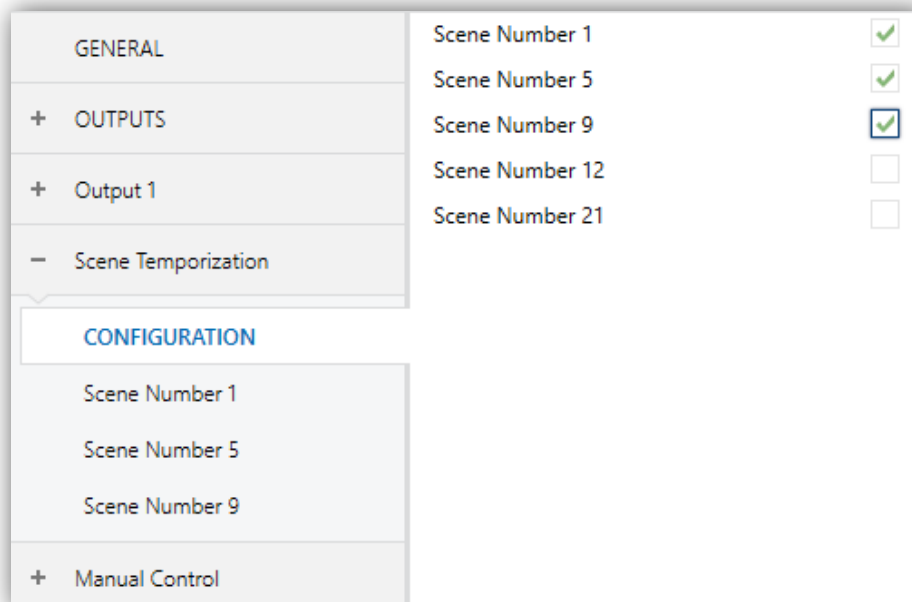


**Figure 6.** Scene temporisation

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 <u>for each of the outputs where it has been configured</u>.
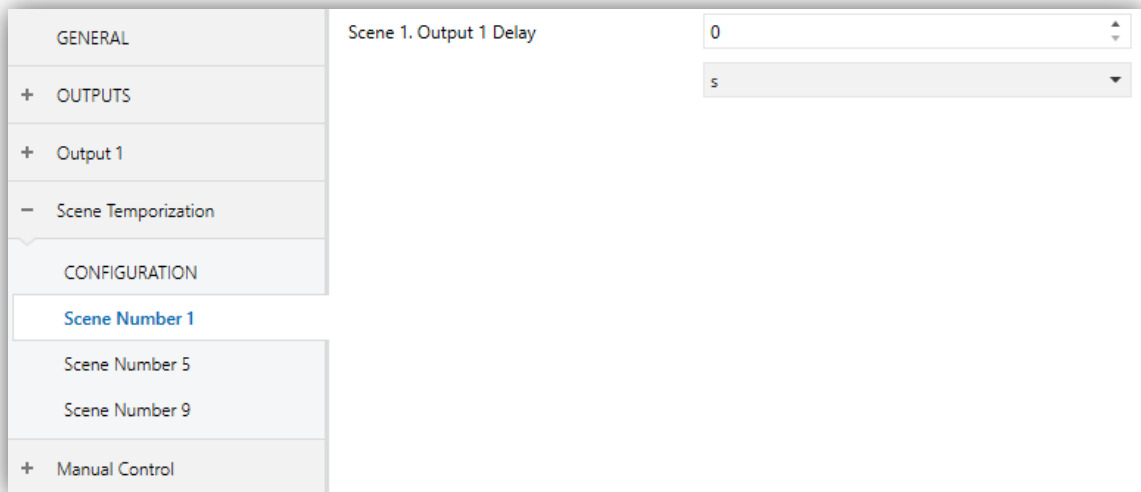


**Figure 7.** Configuration of the scene temporisation

Therefore, parameter "**Scene m. Z Delay**" defines the delay that will be applied to the action defined in Z for the execution of scene m (where Z may be a specific individual output or shutter channel). 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 programme itself.

| 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 0 |
| 3 | 1 Bit | | C T - - - | DPT_Trigger | 0/1 | Reset 1 | Voltage Recovery -> Sending 1 |
| 4 | 1 Bit | I | C - - W - | DPT_Enable | 0/1 | Manual Control Lock | 0 = Lock; 1 = Unlock |
| | 1 Bit | I | C - - W - | DPT_Enable | 0/1 | Manual Control Lock | 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 | 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 | | [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 | -65535 | [LF] Function x - Result | (2-Byte) Signed |
| | 2 Bytes | O | C T R - - | DPT_Value_Temp | -273.00 ºC - 670760.00 ºC | [LF] Function x - Result | (2-Byte) Float |
| 87 | 1 Byte | I | C - - W - | DPT_SceneControl | 0-63; 128-191 | [Outputs] Scenes | 0 – 63 (Execute 1 – 64); 128 – 191 (Save 1 – 64) |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 88, 96 | 1 Bit | I | **C - - W -** | DPT_BinaryValue | 0/1 | [Ox] On/Off | N.O. (0=Open Relay; 1=Close Relay) |
| | 1 Bit | I | **C - - W -** | DPT_BinaryValue | 0/1 | [Ox] On/Off | N.C. (0=Close Relay; 1= Open Relay) |
| 89, 97 | 1 Bit | O | **C T R - -** | DPT_BinaryValue | 0/1 | [Ox] On/Off (Status) | 0=Output Off; 1=Output On |
| 90, 98 | 1 Bit | I | **C - - W -** | DPT_Enable | 0/1 | [Ox] Lock | 0=Unlock; 1=Lock |
| 91, 99 | 1 Bit | I | **C - - W -** | DPT_Start | 0/1 | [Ox] Timer | 0=Switch Off; 1=Switch On |
| 92, 100 | 1 Bit | I | **C - - W -** | DPT_Start | 0/1 | [Ox] Flashing | 0=Stop; 1=Start |
| 93, 101 | 1 Bit | I | **C - - W -** | DPT_Alarm | 0/1 | [Ox] Alarm | 0=Normal; 1=Alarm |
| | 1 Bit | I | **C - - W -** | DPT_Alarm | 0/1 | [Ox] Alarm | 0=Alarm; 1=Normal |
| 94, 102 | 1 Bit | I | **C - - W -** | DPT_Ack | 0/1 | [Ox] Unfreeze Alarm | Alarm=0 + Unfreeze=1 => End Alarm |
| 95, 103 | 1 Bit | O | **C T R - -** | DPT_State | 0/1 | [Ox] Warning Time (Status) | 0=Normal; 1=Warning |
| 104 | 1 Byte | I | **C - - W -** | DPT_SceneControl | 0-63; 128-191 | [Shutter] Scenes | 0 - 63 (Execute 1 - 64); 128 - 191 (Save 1 - 64) |
| 105 | 1 Bit | I | **C - - W -** | DPT_UpDown | 0/1 | [Cx] Move | 0=Raise; 1=Lower |
| 106 | 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 |
| 107 | 1 Bit | I | **C - - W -** | DPT_Enable | 0/1 | [Cx] Lock | 0=Unlock; 1=Lock |
| 108 | 1 Bit | O | **C T R - -** | DPT_Switch | 0/1 | [Cx] Rising Relay (Status) | 0=Opened; 1=Closed |
| 109 | 1 Bit | O | **C T R - -** | DPT_Switch | 0/1 | [Cx] Lowering Relay (Status) | 0=Opened; 1=Closed |
| 110 | 1 Byte | O | **C T R - -** | DPT_Scaling | 0% - 100% | [Cx] Shutter Position (Status) | 0%=Top; 100%=Bottom |
| 111 | 1 Byte | O | **C T R - -** | DPT_Scaling | 0% - 100% | [Cx] Slats Position (Status) | 0%=Open; 100%=Closed |
| 112 | 1 Byte | I | **C - - W -** | DPT_Scaling | 0% - 100% | [Cx] Shutter Positioning | 0%=Top; 100%=Bottom |
| 113 | 1 Byte | I | **C - - W -** | DPT_Scaling | 0% - 100% | [Cx] Slats Positioning | 0%=Open; 100%=Closed |
| 114 | 1 Bit | I | **C - - W -** | DPT_Alarm | 0/1 | [Cx] Alarm | 0=No Alarm; 1=Alarm |
| | 1 Bit | I | **C - - W -** | DPT_Alarm | 0/1 | [Cx] Alarm | 0=Alarm; 1=No Alarm |
| 115 | 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 |
| 116 | 1 Bit | I | **C - - W -** | DPT_Ack | 0/1 | [Cx] Unfreeze Alarm | Alarm=0 + Unfreeze=1 => End Alarm |
| 117 | 1 Bit | I | **C - - W -** | DPT_Scene_AB | 0/1 | [Cx] Move (Reversed) | 0=Lower; 1=Raise |
| 118 | 1 Bit | I | **C - - W -** | DPT_Ack | 0/1 | [Cx] Direct Positioning | 0=No Action; 1=Go to Position |
| 119 | 1 Bit | I | **C - - W -** | DPT_Ack | 0/1 | [Cx] Direct Positioning 2 | 0=No Action; 1=Go to Position |
| 120 | 1 Bit | I | **C - - W -** | DPT_Ack | 0/1 | [Cx] Direct Positioning (Save) | 0=No Action; 1=Save Current Position |
| 121 | 1 Bit | I | **C - - W -** | DPT_Ack | 0/1 | [Cx] Direct Positioning 2 (Save) | 0=No Action; 1=Save Current Position |