![Zennio logo]

Presentia C

Presence Detector with Luminosity Sensor for Ceiling Mounting

ZPD-C30L

www.zennio.com

USER MANUAL

CONTENTS

1 INTRODUCTION

1.1 PRESENTIA C

Presentia C from Zennio is a device that aims at, among other functions, the detection of presence, the measurement and control of the room luminosity and the detection of occupancy within the room where it has been installed. It has been designed for ceiling or false ceiling mounting by means of the bundled accessories.

The most outstanding features of Presentia C are:

- **4 sensors** with configurable sensitivities.

- **4 LEDs** to indicate motion.

- **Presence detection**:

  - 6x presence detection virtual channels.

  - Luminosity-dependent presence detection (optional).

  - Periodic and delayed sendings (binary, scene, HVAC, percentage).

- **Occupancy detection**:

  - 1x occupancy detection channel.

  - Master / slave configuration.

  - Trigger upon door opening or closing.

  - Periodic and delayed sendings (binary, scene, HVAC, percentage).

- **Luminosity measurement**:

  - Configurable correction factor and offset.

  - Periodic sending or upon value change.

- 2 **constant light control** channels with configurable setpoints.

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

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

- **Day / night** configuration.

1.2 INSTALLATION

Presentia C 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.

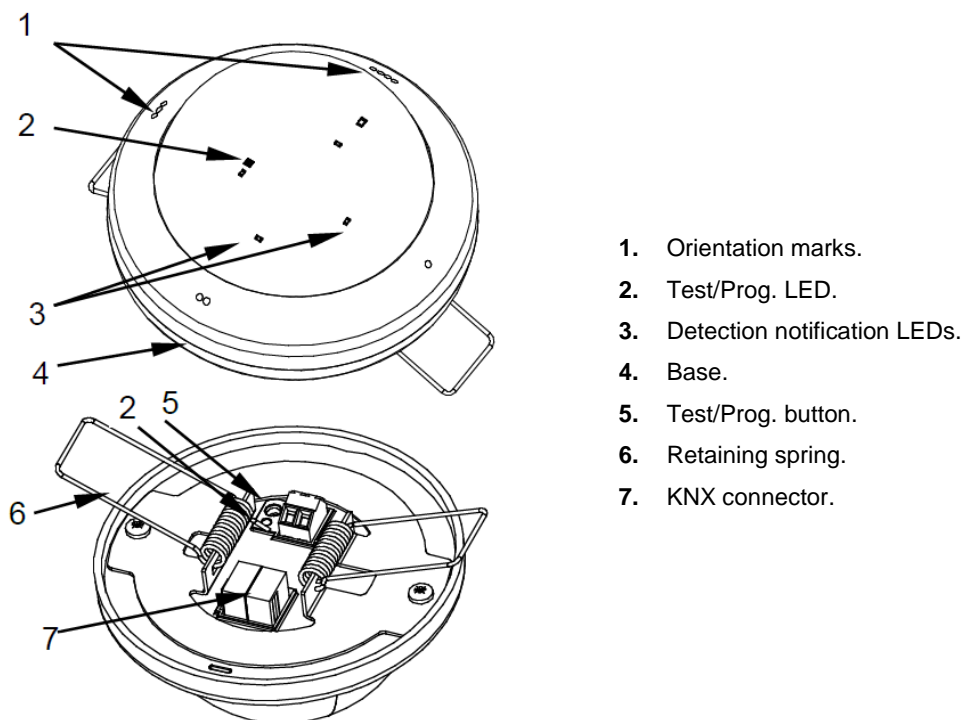This device does not need any additional external power since it is entirely powered through the KNX bus.

1. Orientation marks.
2. Test/Prog. LED.
3. Detection notification LEDs.
4. Base.
5. Test/Prog. button.
6. Retaining spring.
7. KNX connector.

Figure 1. Presentia C. Elements

The main elements of the device are described next.

- **Programming button (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 the* **safe mode**. *In such case, the LED will blink in red every 0.5 seconds.*

- **Detection notification LEDs (3)**: each of them emits a light flash whenever the sensor associated to that zone observes motion.

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 motion sensors are ready.

Depending on the configuration, some specific actions will also be performed during the start-up. For example, the integrator can set whether the detection channels should start up enabled or disabled. Please consult the next sections of this document forfurther 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.

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 window 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.
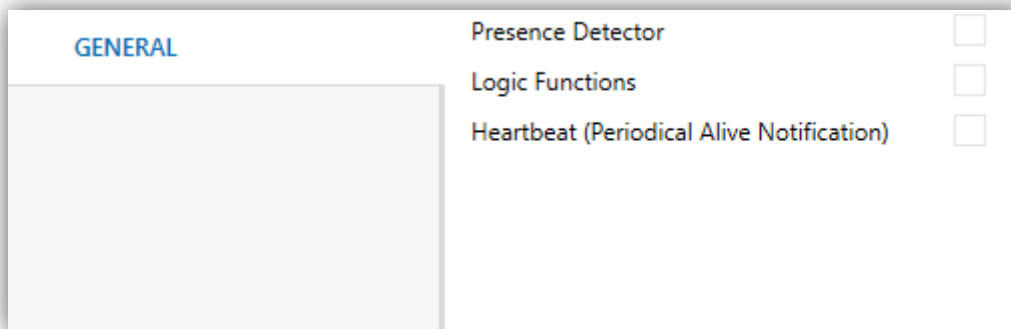

Figure 2. General screen

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

- **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).
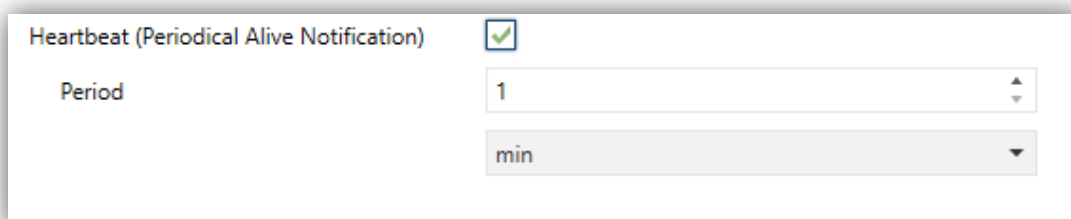

Figure 3. Heartbeat (periodic alive notification).

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 PRESENCE DETECTOR

2.2.1 CONFIGURATION

Presentia C incorporates six virtual, independent presence detection channels, two more for constant light control and one for occupancy detection.

- **Presence detection** consists in sending objects to the bus whenever the device observes a moving body (or no longer observes it) in the environment of the room where it has been installed.

- **Constant light control** consists in sending KNX orders to the dimmer device that controls the in-room luminaries so the ambient light level remains constant even if other light sources are present.

- **Occupancy detection** is an algorithm that allows determining, through a multiple sensor configuration, whether a particular space is under occupation no matter if the occupant moves or not (i.e., no matter if the device is detecting presence in the room or not).

The general configuration of the presence detection system allows enabling or disabling each of these virtual *channels*.

It also allows setting different luminosity setpoints or object types for **daytime and night time** as well as activating or deactivating the **motion indicator LEDs**.

On the other hand, Presentia C allows measuring the **luminosity** in the roomy and, among other functions, sending the value to the KNX bus. By applying certain correction parameters it is possible to compensate factors that may distort the measurements obtained by the sensor. To that end, two variables are defined:

- **Correction Factor**: ratio between the measured luminosity and the actual luminosity.

- **Offset**: deviation between measured and actual luminosities that remains constant.

Note: *to obtain the above values, it is advisable to compare the measurements of both a Presentia C installed on the ceiling and a luxometer on the work surface, i.e., the surface where the actual luminosity is required to be monitored. In case there are light*

sources of different types within the facility, it is advisable to perform this calculation for each of them and afterwards obtain the average values. For further details, please refer to the documentation available at http://www.zennio.com.

Example:

- **Correction factor** = 0.5

- **Offset** = 100 luxes

- **Luminosidad sent to the bus** = 100 + (2 * Luminosity measured by the sensor).

Finally, it is also possible to set custom **sensitivity** values for the motion sensors and to perform certain adjustments regarding the luminosity measurement. Factors such as the motion speed and direction together with the distance to the device may have a positive or negative effect the motion detection.

Distance		Speed		Direction	
Longer	**Shorter**	**Greater**	**Lower**	**Front**	**Tangential**
Negative	Positive	Positive	Negative	Negative	Positive

Table 1. Factors conditioning the motion detection

Therefore, **low sensitivity** may prevent the detection of slower and front movements, especially at longer distances from the sensor. On the other hand, **higher sensitivity** may saturate the detection, especially at shorter distances.

Presentia C allows configuring the sensitivity of each of the four motion sensors (one per detection sector) independently.

ETS PARAMETERISATION

From the Configuration tab, it is possible to enable or disable the different functions from Presentia C related to the motion detection, to the luminosity measurement and to the sensors sensitivity.

Figure 4. Presence detector. Configuration

- **Presence Detection 1-6**: activates or deactivates the different detection channels available. Each channel operates independently, which allows setting parallel criteria (in terms of delays, thresholds and actions) for a fixed amount of hardware sensors. See section 2.2.2.

- **Constant Lighting Control 1-2**: activates or deactivates the constant lighting control channels. See section 2.2.3.

- **Occupancy Detection**: activates or deactivates the occupancy detection function. See section 2.2.4.

- **Day/Night**: activating this functionality makes certain parameters referring to detection sendings and luminosity setpoints become double (see later sections). One binary object ("**Day/Night**") will also become visible to allow reception of the day ↔ night transitions.

Figure 5. Day/night configuration

- **Detection LEDs**: activates or deactivates the LEDs that indicate motion detection. If **Day/Night** has been enabled, the "Enabled only in Day" option will also be available.

With regard to LUMINOSITY, the following parameters are available:

- **Internal Sensor**: parameters to allow compensation of the luminosity values measured by the internal sensors.

  ➤ **Correction Factor**: ratio between 1 (default) and 0.

  ➤ **Offset**: value between -200 and 200 lux (default is 0).

- **Luminosity Sending**: activates or deactivates an automatic sending of the luminosity level in lux through the "**Luminosity – Internal Sensor**" object. If enabled, two additional parameters will show:

  ➤ **Period**: period to send the luminosity value to the bus, between 0 and 255 seconds. If set to 0, the periodic sending is disabled.

  ➤ **Luminosity Change to Send**: sets a minimum increment (in luxes) so when two consecutive readings of the luminosity measurement differ in more than such value, an extra sending to the bus will take place. If set to "0", this option remains disabled

Figure 6. Luminosity sending

With regard to SENSITIVITY, the following parameters are available:

- **Sensor N**: sensitivity for sensor number N. The default value is 90% (maximum sensitivity).

- **Sensitivity Objects**: shows or hides the "**Sensor N Sensitivity**" to allow changeovers of the values set in parameters.

The project topology shows the following objects as well:

- "**Scene Input**" and "**Scene Output**": objects to receive or send scene values from or towards the KNX bus.

2.2.2 PRESENCE DETECTION CHANNEL

Presentia C allows enabling up to six presence detection channels, as well as setting a different configuration (e.g., the type of the objects to be sent) for each.

2.2.2.1 CONFIGURATION

It is possible to specify **which of the four existing sensors** will be taken into account for each detection channel, together with the following options:

- Motion Detection Conditions.

- Operating Mode.

- Enablement.

- Forced State.

- External Motion Sensor.

- External Switch.

- Type (Light, HVAC, Presence).

These options are detailed next.

Motion Detection Conditions

The channel will switch to the "Detection" state when a motion detection signal is received from the sensors and to the "No Detection" state when such signal is no longer received.

- When the channel switches to the "Detection" state, a time count will begin. This count will be restarted whenever a motion detection signal is received.

- After some time (**detection length**) without further motion detections, the channel will switch to the "No Detection" state.

- At that moment, a new time count (**blind time**) begins so that any motion detections that may take place before it expires will be ignored.

- Once the blind time expires, the device will adopt the "Detection" state as soon as a new motion detection takes place.

Operation Mode

Each of the channels can be configured to work either automatically or semi-automatically:

- **Automatic mode**: allows switching between the Detection and No Detection states not only through actual motion but also through a specific, external object.

- **Semiautomatic Mode**: one of the two states (configurable) will <u>only</u> be reachable through the external switch object.

In any case, writing over the external switch object will make the Detection or No Detection become active immediately.

Enablement

While a channel remains disabled, sendings to the KNX bus <u>related to that channel</u> will be interrupted, no matter if motion detections are actually taking place or not.

It is possible to enable and disable a channel through a **binary object** or through a **scene**.

Forced State

Values received through the forced state object will be interpreted by the device as a master detection (or no detection) order. In other words, **this object allows externally overwriting the motion detection signal**. The device response when this object is received is as follows:

- An *On* is received:

  ➢ If the channel was under "Detection", no actions will take place.

> If the channel was under "No Detection", it will switch to "Detection".

- An *Off* is received:

  > If the channel was under "Detection", it will switch to "No Detection".

  > If the channel was under "No Detection", no actions will take place.

After forcing a state, the channel will remain under it for a (configurable) **idle time**. Once this time elapses:

- If the forced state was "No Detection":

  The channel will remain under this state, without performing any sendings until further motion is detected.

- If the forced state was "Detection":

  > If the time elapsed since the last motion detection is greater than the detection length, the channel switches to "No Detection" and sends the corresponding object (unless the channel is found to be disabled).

  > If the time elapsed since the last motion detection has not reached the detection length yet, the channel remains under the "Detection" state without performing any sendings.

The execution of the forced state is unconditional and does not depend on whether the channel remains disabled nor on the state of the sensor. In other words, the forced state function has a higher priority over other functions.
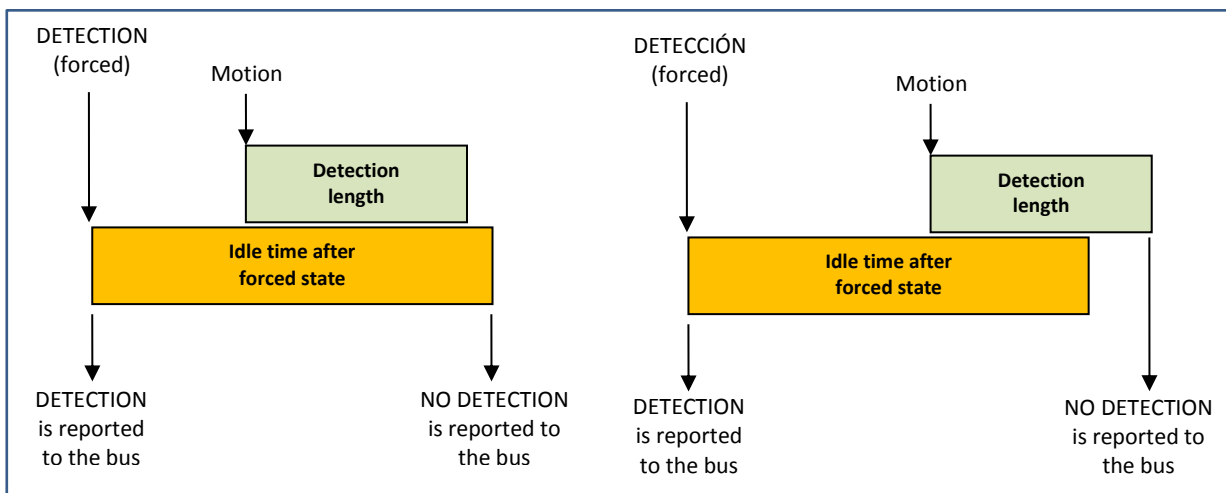

Figure 7. Forced states (example)

External motion detection

An object is available for the **detection of external motion**, so that other KNX devices can inform about motion detection and a joint response can be executed. Whenever this object receives one 1 or one 0, the channel will behave exactly the same as if the actual sensor had detected motion.

Therefore, two types of detectors can be defined:

- **Slave detectors**: it is possible to have several of them. They are in charge of transmitting the detection state to the master detector. To that end, all of them must remain enabled and share the same configuration. The following configuration is recommended:

  ➢ Detection: send one "1".

  ➢ No detection: no sending.

- **Master detector**: only one master detector must be present. It will be in charge of performing actions over the device under control. The following configuration is recommended:

  ➢ External detection object: enabled and linked to the detection objects from the slave detectors.

  ➢ Detection length: greater than the detection lengths of all the slave detectors.

  ➢ Avoiding forced states is encouraged.

Type

According to the required amount of functionality, each presence detection cannel may be configured as a:

- **Presence monitor**

  This is the simplest type. It provides the aforementioned functions.

- **Light control**

  This is a lighting-dependent channel:

  ➢ The Detection state is only reachable if the light level is lower than a certain value (**detection threshold**).

> ➢ The No Detection state will only be reachable if the light level is high enough (**no-detection threshold**) as to permit, once the luminaries are switched off, a light level greater than the **detection threshold**. The perceptible effect is that the luminaries under control remain off even if motion is being detected, since the detection threshold is still exceeded. To that end, Presentia C will automatically calculate the light level contribution of the luminaries under control and therefore the **no-detection threshold**, although a certain hysteresis can be parameterised.

Example: *suppose a room with an artificial light contribution of 200 luxes, apart from the natural light contribution. The detection and no-detection thresholds are of 100 and 400 luxes.*

- *If the current light level is over 100 luxes, the luminaries will remain off even if motion is detected in the room. The additional light contribution is considered unnecessary.*

- *If the current light level is under 100 luxes, the luminaries are switched on as soon as motion is detected in the room since the light level is insufficient and therefore the artificial contribution becomes necessary.*

- *Being the luminaries on, once no further motion is detected they will be switched off. No light contribution is necessary as there is no motion.*

- *Being the luminaries on, if the light level becomes greater than 400 luxes, the luminaries will be switched off making the light level decrease to 200 luxes. No additional light contribution is necessary as such value already exceeds the detection threshold.*

- **Control de HVAC**

  Detection channels destined to control climate systems do not depend on luminosity. They provide a filter to allow switching on the unit only if motion is being detected on a continuing basis, which helps preventing the switch-on of the climate system in case of exceptional or temporary movements.
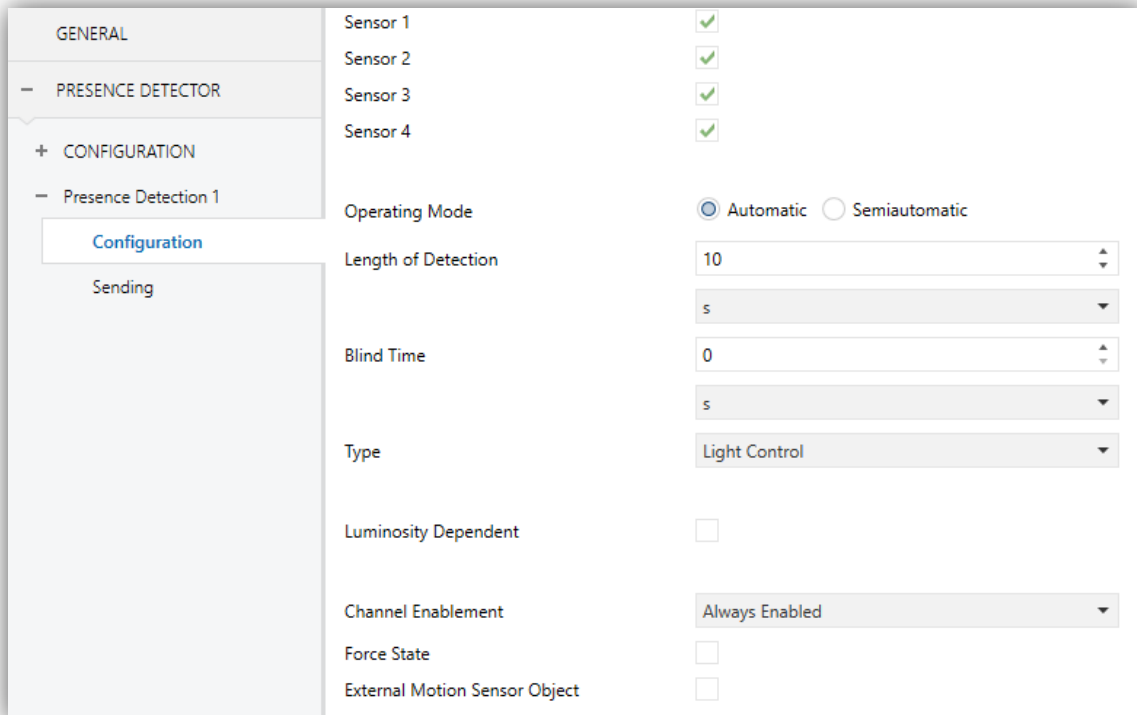
  This filter consists in a set of time windows, which allow switching to the Detection state only if motion has been detected in all of them.

Example: *a two 30-second window filter:*

- *Immediately after the first detection, a 30-second time count begins. If motion is detected in this time, the first window is considered overcome.*

- *Once the first 30-second time count finishes, a second window beings. If no motion is detected in this time, the process is aborted and does not start over until a new detection takes place. However, if detections do take place during the second window as well, then the Detection state will be adopted immediately.*

ETS PARAMETERISATION

Figure 8. Configuring a presence detection channel

The Configuration tab contains the following parameters:

- **Sensor n**: enables/disables sensor number n for the current channel. If unmarked, detections reported by such sensor will not be taken into account.

- **Operating Mode**: "Automatic" or "Semiautomatic". If the latter is selected, the following parameter will show up:

➤ **Acción via External Switch Object**: sets which event will only take place through the "**[PDx] External Switch**" object:

- "Detection": the Detection state can only be triggered through this object.

- "No Detection": the No Detection state can only be triggered through this object.

Depending on the above configuration, either both or only one of the two following parameters will be shown.

- **Detection Length**: sets the minimum period of time that must elapse without motion detections before the channel switches to the No Detection state (further detections will interrupt this time count). The parameterisable values are 1 to 255 seconds, 1 to 255 minutes and 1 to 18 hours. In addition, the "**[PDx] Length of Detection**" object will be available to allow overwriting this value (in seconds).

- **Blind Time**: sets the minimum period of time the channel will remain under the No Detection state, even if motion detections are taking place in the room, which will be ignored by the channel. The parameterisable values are 1 to 255 seconds, 1 to 255 minutes and 1 to 18 hours.

- **Type**: defines the specific functionality the channel will be oriented to, which will show or hide certain parameters to make the configuration easier:

➤ "Presence Monitoring": shows no additional parameters.

➤ "Light Control": adds the following parameters:

- **Luminosity Dependent**: determines whether the transitions to Detection or No Detection should depend on the luminosity or not. If enabled, the following parameters will also show up.

  o **Detection Threshold**: light level (in luxes) over which motion detections start to be ignored.

  o **No-Detection Threshold Hysteresis**: safety factor (in percentage) for the calculation of the No Detection threshold.

o **Luminosity Measurement Delay**: time period the device will wait for after the detection of motion, prior to calculating the increase in the light level it produces. This must be greater than the dimming time of the luminaries to ensure that the light level is already stable.

Note: *depending on whether the configured* **operating mode** *is automatic or semiautomatic and, in the latter case, on whether the action for the* **external switch object** *is Detection or No Detection, some of these three parameters will not be shown.*

Figure 9. Light control channel

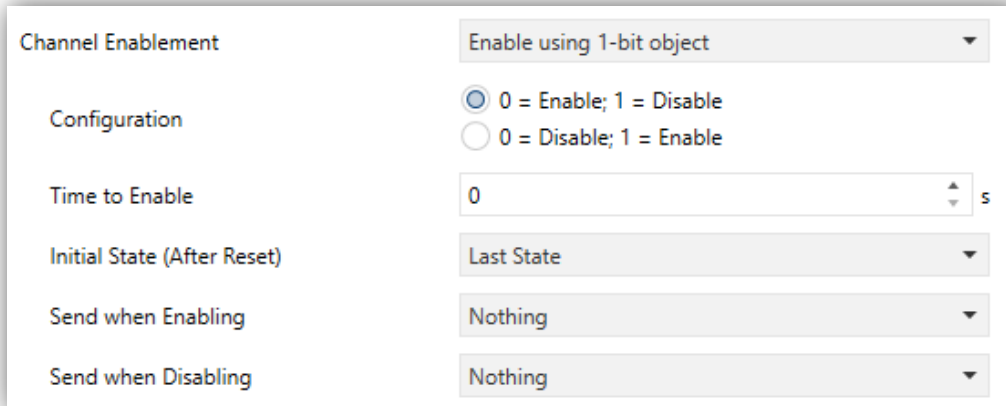➤ "HVAC Control": shows the following parameters:

- **Observation Windows**: number of time windows in which motion must be detected before switching to Detection.

- **Observation Time**: length (in seconds) of each observation window.

Figure 10. HVAC channel

✦ **Channel Enablement**: sets how the channel can be enabled or disabled:

➤ "Always Enabled": the channel remains always active.

➢ "Enable Using 1-Bit Object": the channel will switch from active to inactive (and vice versa) by receiving a binary value through the "**[PDx] Enable Channel**" object.

Figure 11. Enabling the channel by using a 1-bit object

Activating this option implies setting the following parameters as well:

- **Configuration**: sets which values (0 or 1) enable and disable the channel.

- **Time to Enable**: sets a delay for the re-activation (re-enablement) of the channel after the reception of the corresponding value. The range is 0 to 255 seconds.

- **Initial State (After Reset)**: determines the state to be acquired by the channel after the start-up of the device or after a bus power failure: "Last State", "Enabled" or "Disabled". After the very first start-up, the last state of the channel is assumed to be enabled.

- **Send When Enabling**: sets which value should be sent to the bus once the channel changes to enabled, in order to notify that the detection process is being resumed. "Nothing", "No Detection" or "Detection".

- **Send When Disabling**: analogous to the above parameter. It sets the value to be sent to the bus when the channel changes to disabled. "Nothing", "No Detection" and "Detection".

➢ "Enable Using a Scene": the channel will switch between enabled and disabled depending on the scene values received through the "**Scene Input**" object.

Figure 12. Enabling the channel by using scenes

Activating this option implies setting the following parameters as well:

- **Scene to Enable**: determines the scene number (between 1 and 64) that will activate the channel.

- **Scene to Disable**: determines the scene number (between 1 and 64) that will deactivate the channel.

- **Time to Enable**, **Initial State**, **Send When Enabling**, **Send When Disabling**: these parameters are similar to those for the "Enable Using 1-Bit Object" option, already described.

● **Force State**: enables or disables the "**[PDx] Force State**" binary communication object.

Figure 13. Forced state

➢ **Idle Time After Forced State**: sets the length of time the channel will remain under the forced state. The available range is 1 to 255 seconds, 1

to 255 minutes and 1 to 18 hours. Please note that forced states ignore the length of detection and the blind time that may have been configured.

- **External Motion Sensor Object**: allows enabling or disabling a specific communication object ("**[PDx] External Motion Detection**") to receive motion detections from other KNX devices, so that several devices can combine their states and perform a joint response.

2.2.2.2 SENDING

It is possible to define custom sendings and custom actions to be performed once the presence detection channel switches from Detection to No Detection (or vice versa), for both daytime and nighttime:

- **Object type**: the Detection and No Detection events can trigger one or several of the following objects, with a customisable value: binary object, scene, HVAC modes and percentage values (scaling).

- **Periodic sending**: objects can be sent to the bus either once or with a configurable period.

- **Delay**: objects will be sent either just after the state switchover or delayed.

ETS PARAMETERISATION

The Sending tab contains the following parameters, which are analogous for both Detection and No Detection.

- **Binary Value**: enables or disables the sending of a binary value (either "<u>On</u>" or "<u>Off</u>", as set in **Value**) through the "**[PDx] Output (binary)**" communication object.

- **Scene**: enables or disables the sending of a scene execution order (between 1 and 64, as set in **Value**) through the "**Scene Output**" communication object.

- **Scaling**: enables or disables the sending of a percentage value (as set in **Value**) through the "**[PDx] Output (Scaling)**" communication object.

- **Sending Mode**: sets whether the values will be sent "<u>Once</u>" or on a "<u>Cyclic</u>" basis. If the latter is selected, the following field will also be shown:

> ➢ **Period**: 1 to 255 seconds; 1 to 255 minutes; 1 to 18 hours.

- **Delay**: sets a delay between the detection state switchover and the actual sending to the bus. The available values are 0 to 255 seconds, 0 to 255 minutes and 0 to 18 hours.

Figure 14. Sending

- **HVAC Mode** (only if the channel type is "HVAC Control"; see section 2.2.2.1): enables or disables the sending of a special HVAC mode value (as set in **Value**) through the "**[PDx] Output (HVAC)**" object:

HVAC Mode	Object value
Comfort	1
Standby	2
Economy	3
Building Protection	4

Table 2. HVAC modes

Under a day and night dependent configuration (see section 2.2.1), all parameters regarding the object sending will be duplicated to allow different options depending on whether the detection state switchover takes place during daytime or nighttime.

2.2.3 CONSTANT LIGHTING CONTROL

The aim of the constant light regulation channels is maintaining the room light level as stable as possible around a certain **setpoint** by compensating natural light changes through artificial light.

- As long as **presence is being detected**, periodic light regulation values will be sent. The greater the difference between the setpoint and the actual light level is, the higher these values will be. In addition, after a detection it will be posible to maintain an **initial dimming level** during some time.

  Once the light level setpoint is met, light dimming will be interrupted. A tolerance band around the setpoint will also be parameterisable to prevent continuous changeovers in the artificial light contribution.

- After switching to **No Detection**, a dimming value of 0% will be sent.

It is possible to define a set of **dimming coefficients**, so the light increments are higher or lower. Please note that too high an increment may cause a light contribution greater than needed; to prevent this and the system from regulating over and over, it is encouraged to set the coefficients (P) shown in Table 2, given the luminosity contribution of the elements under control.

Example:

P = 15, setpoint = 450 luxes, current luminosity = 200 luxes.

- *Current dimming value = 0%.*

- *Increment = (450 – 200) * (15 / 100) = 37.5. This value divided by 255 gives 14.70%, which will be the new dimming value to be sent to the bus.*

- *Assuming this action increases the luminosity up to 280 luxes, the next increment will be (450 – 280) * (15 / 100) = 25.5, which equals to 10%. The new dimming value to be sent will be: 14.70% + 10% = 24.70%.*

Contribution (luxes)	Maximum P	Recommended P
100	255	128
200	128	64
300	85	43
400	64	32
500	51	26
600	43	21
700	36	18
800	32	16
900	28	14
1000	26	13
1100	23	12
1200	21	11
1300	20	10
1400	18	9
1500	17	9
1600	16	8
1700	15	8
1800	14	7
1900	13	7
2000	13	6

Table 3. Maximum and recommended coefficients given the luminosity contribution

ETS PARAMETERISATION

The Configuration tab shows the following parameters:

- **Sensor n**, **Operating Mode**, **Lenght of Detection**, **Blind Time**, **Channel Enablement**, **Force State** and **External Motion Sensor Object**: analogous to those already described for the presence detection channels (see section 2.2.2).

  Objects related with these functions are also named accordingly:

  ➢ "**[CLCx] External Switch**",

  ➢ "**[CLCx] Length of Detection**",

  ➢ "**[CLCx] Enable Channel**",

  ➢ "**[CLCx] External Motion Detection**",

  ➢ "**[CLCx] Force State**".

Figure 15. Constant lighting control

- **Setpoint**: target luminosity (in luxes), which can be modified afterwards through the "**[CLCx] Setpoint**" object. In case the day / night option has been enabled (see section 2.1), both parameters and objects will be duplicated to allow a different setpoint for each case.

- **Initial Dimming**: sets an initial dimming level to be sent after detection.

  - **Value**: level in percentage for the initial regulation

  - **Dwell Time in Initial Value**: time period (in seconds) the channel will maintain the initial dimming level. Once elapsed, the successive dimming values to be sent will be calculated depending on the error between the actual light level and the setpoint.

- **Cycle Time**: time period (in seconds) to calculate and send the successive regulation levels during the Detection state.

- **Tolerance**: margin or hysteresis (in luxes) to be applied over the setpoint.

- **P Coefficient for Upward Dimming**: value of the proportional coefficient for upward dimming orders. The higher its value, the greater the regulation increment for a fixed error.

- **P Coefficient for Downward Dimming**: analogous to the previous parameter, for upward dimming orders.

2.2.4 OCCUPANCY DETECTION CHANNEL

For a better understanding of the function of the occupancy detection channels it is important to note that the **detection of motion and presences** does not necessarily imply that the room is currently occupied by people, while in the same way the non-detection does not necessarily imply that the room is not occupied.

The occupancy detection in an algorithm which takes a set of variables (detection events, door opening and closing events) as an input to determine whether **there are individuals or not inside the room** at the moment.

2.2.4.1 CONFIGURATION

The occupancy detection entails combining several devices:

- Motion detectors (one as a **master** and the others as **slaves**),

- **Door opening and closing** sensors or indicators, for entrance and departure monitoring.

Presentia C can behave both as a master detector or as a slave detector.

Master

The master detector will be responsible for determining whether there is occupancy or not depending on the information it receives from the slave detectors (through the "**Occupancy: Slave Input**") and on its own information.

Transitions between one occupancy state and the other will trigger object sendings to the KNX bus, according to the configuration set in parameters.

The **occupancy detection algorithm** takes places as follows:

- When a **door opening** is received (through the trigger object): the "**Occupied**" state will be adopted (if not already active).

- When a **door closing** is received (through the **trigger object**):

  1. During a certain **waiting time**, the algorithm will ignore any motion detections reported by either the master or the slave detectors.

  2. Next, the master starts counting a certain **listening time**.

  3. If either the master itself or the slave detectors report motion, the "**Occupied**" state will be adopted (if not already active).

     a. If the listening time expires after none of the detectors has reported any motion detections, the "**Not Occupied**" state will be adopted.

     b. From that moment, any motion detection will activate the "**Occupied**" state (if not already active).

It is possible to associate certain **sendings** (a binary value, a scene, a percentage value or an HVAC mode) for the transitions between the "Occupied" and "Not Occupied" states and vice versa. A **status object** is also provided which will be sent whenever the occupancy state changes.

Occupancy detection can be enabled or disabled, analogously as in presence detection channels (see section 2.2.2), through the **enablement** and **scene input** communication objects. An extra sending of the aforementioned values (binary, scene, percentage or HVAC) can be performed during the transitions between the Enabled and Disabled, although this will not alter the value of the occupancy **status object**.

While the occupancy detection remains disabled, all the involved events (i.e., motion detections and trigger objects) are ignored.

To prevent undesired occupancy detections (**false occupancy detections**) it is possible to send a scene-recording request object to the bus as soon as, being under the "Occupied" state, the trigger signal is received. If no further detections take place before the listening time elapses, the transition to the "Not Occupied" state will be notified as usually. But in such circumstances, if motion is detected **without having received the trigger value once again**, then the preceding situation will be considered as a "false no-occupation" (actually, the room was still occupied), so an order to **execute the previously recorded scene** will be sent to make the room be recover the state it had prior to the transition to the "Not Occupied" state, and the "**Occupied**" state will be adopted again (in this case, the parameterised **sendings** for such transition will not be sent, although the occupancy status object will be).

> **_Example_**_: false empty room detection._
>
> _Think of a room with two people sleeping inside ("Occupied" state). If the false occupancy detections are not avoided, when one of them leaves the room the "Not Occupied" state will be adopted. If afterwards the other occupant moves, the state will switch to "Occupied" and the corresponding order will be sent, making for instance the lights turn on. On the contrary, if this option is considered, when the remaining occupant moves the occupancy action will not be executed as no additional trigger values have been received (e.g., a new door opening notification)._
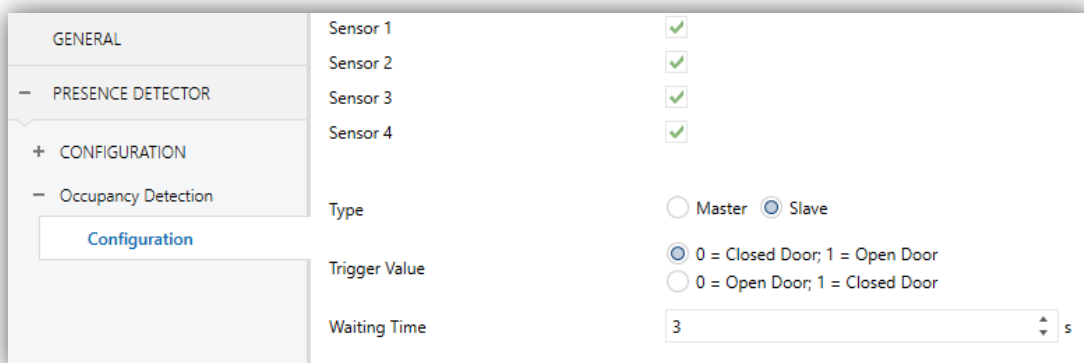
Slave

The slave Presentia C devices notify the master about detection and no-detection situations (by respectively sending a "1" or a "0" through the **slave output** object), although in this case switching from the detection state to the no-detection state is only possible after the reception of the trigger signal (e.g., a door closing), thus, also the slave detectors provide the **trigger** object.

After the reception of the trigger signal, the slave will allow a certain **waiting time** (configurable), the count of which will be restarted in case the trigger signal is received again. After the time count finishes, the slave switches to No Detection.

Once in No Detection, it will switch to Detection as soon as motion is detected.

ETS PARAMETERISATION

The Configuration tab shows the following parameters:

Figure 16. Occupancy detection (slave)

Common parameters for both master and slave:

- **Sensor n**: enables/disables sensor number n for the current channel. If unmarked, detections reported by such sensor will not be taken into account.

- **Tipo**: sets the occupancy detection channel type ("Master" or "Slave").

- **Trigger Value**: determines which values will indicate the door state, upon reception through "**Occupancy: trigger**": "0 = Closed Door; 1 = Open Door" and "0 = Open Door; 1 = Closed Door".

- **Waiting Time**: sets the waiting time before switching to No Occupancy, which is counted after the reception of the closed door trigger. Range: 0 to 65535 seconds. This time can be modified through object "**Occupancy: waiting time**".

  Note: *the waiting times in all slave detectors must be greater than (or equal to) the waiting time of the master detector to ensure a proper behaviour*.

Exclusive parameters for master detectors:

Figure 17. Occupancy detection (master)

- **Listening Time**: sets the value of the listening time, which is counted after the waiting time elapses. Range: 0 to 65535 seconds (2 seconds by default). This value can be modified afterwards through object "**Occupancy: Listening Time**".

- **Occupancy State Object**: enables or disables the "**Occupancy: Occupancy State**" binary object, which allows requesting the current occupancy state anytime (1 = occupied; 0 = not occupied).

- **False Non-Occupied Detection**: enables or disables the algorithm that prevents false detections of the Not Occupied state, which implies setting the **Scene Number** to be recorded and executed.

- **Enablement**: provides a procedure for temporary disabling the occupancy detection function. This procedure is analogous to that for the presence detection (see section 2.2.2.1).

2.2.4.2 SENDING

As for presence detection channels, occupancy detection channels allow sending different objects and values to the bus, which can be periodic or delayed (depending on the event type) and different for daytime and nighttime.

ETS PARAMETERISATION

The Configuration tab shows the following parameters:

Figure 18. Occupancy detection. Sending

The functionality and configuration are entirely analogous as those already explained in section 2.2.2.2, although in this case the **HVAC Mode** sending type will be always available.

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.

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

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 under the **Presentia C** product section at the Zennio homepage (www.zennio.com) for detailed information about the functionality and the configuration of the related parameters.

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		C T - - -	DPT_Trigger	0/1	[Heartbeat] Object to Send '1'	Sending of '1' Periodically
2	1 Byte	I	C - - W -	DPT_SceneControl	0-63; 128-191	Scene Input	Scene Value
3	1 Byte		C T - - -	DPT_SceneControl	0-63; 128-191	Scene Output	Scene Value
4	2 Bytes	O	C T R - -	DPT_Value_Lux	-671088.64 - 670760.96	Luminosity - Internal Sensor	0-2000 Luxes
5	2 Bytes	O	C T R - -	DPT_Value_Lux	-671088.64 - 670760.96	Luminosity - Additional Sensor	0-2000 Luxes
6	1 Bit	I	C - - W -	DPT_Switch	0/1	Day/Night	0 = Day; 1 = Night
	1 Bit	I	C - - W -	DPT_Switch	0/1	Day/Night	0 = Night; 1 = Day
7	1 Byte	O	C T R - -	DPT_Scaling	0% - 100%	Occupancy: Output (Scaling)	0-100%
8	1 Byte	O	C T R - -	DPT_HVACMode	1=Comfort 2=Standby 3=Economy 4=Building Protection	Occupancy: Output (HVAC)	Auto, Comfort, Standby, Economy, Building Protection
9	1 Bit	O	C T R - -	DPT_Occupancy	0/1	Occupancy: Output (Binary)	Binary Value
	1 Bit	O	C T R - -	DPT_Trigger	0/1	Occupancy: Slave Output	1 = Motion Detected
10	1 Bit	I	C - - W -	DPT_Window_Door	0/1	Occupancy: Trigger	Binary Value to Trigger the Occupancy Detection
11	1 Bit	I	C - - W -	DPT_Trigger	0/1	Occupancy: Slave Input	0 = 1 = Detection from slave device
12	2 Bytes	I	C - - W -	DPT_TimePeriodSec	0-65535	Occupancy: Waiting Time	0-65535 s.
13	2 Bytes	I	C - - W -	DPT_TimePeriodSec	1-65535	Occupancy: Listening Time	1-65535 s.
14	1 Bit	I	C - - W -	DPT_Enable	0/1	Occupancy: Enable	According to parameters
15	1 Bit	O	C T R - -	DPT_Occupancy	0/1	Occupancy: Occupancy State	0 = Not Occupied; 1 = Occupied
16, 17, 18, 19	1 Byte	I	C - - W	DPT_Scaling	0% - 100%	Sensor x Sensitivity	0-100%

			-				
20, 30, 40, 50, 60, 70	1 Bit	I	**C - - W -**	DPT_Trigger	0/1	[Cx] External Motion Detection	0 = 1 = Motion detected by an external sensor
21, 31, 41, 51, 61, 71	1 Byte	O	**C T R - -**	DPT_Scaling	0% - 100%	[Cx] Output (Scaling)	0-100%
22, 32, 42, 52, 62, 72	1 Byte	O	**C T R - -**	DPT_HVACMode	1=Comfort 2=Standby 3=Economy 4=Building Protection	[Cx] Output (HVAC)	Auto, Comfort, Standby, Economy, Building Protection
23, 33, 43, 53, 63, 73	1 Bit	O	**C T R - -**	DPT_Switch	0/1	[Cx] Output (Binary)	Binary Value
24, 34, 44, 54, 64, 74	1 Bit	I	**C - - W -**	DPT_Enable	0/1	[Cx] Enable Channel	According to parameters
25, 35, 45, 55, 65, 75	1 Bit	I	**C - - W -**	DPT_Switch	0/1	[Cx] Force State	0 = No Detection; 1 = Detection
26, 36, 46, 56, 66, 76	1 Bit	I	**C - - W -**	DPT_Switch	0/1	[Cx] External Switch	0 = No Detection; 1 = Detection
27, 37, 47, 57, 67, 77	2 Bytes	I	**C - - W -**	DPT_TimePeriodSec	1-65535	[Cx] Length of Detection	1-65535 s.
80, 89	1 Bit	I	**C - - W -**	DPT_Trigger	0/1	[CLCx] External Motion Detection	0 = 1 = Motion detected by an external sensor
81, 90	1 Bit	I	**C - - W -**	DPT_Enable	0/1	[CLCx] Enable Channel	According to parameters
82, 91	1 Bit	I	**C - - W -**	DPT_Switch	0/1	[CLCx] Force State	0 = No Detection; 1 = Detection
83, 92	1 Bit	I	**C - - W -**	DPT_Switch	0/1	[CLCx] External Switch	0 = No Detection; 1 = Detection
84, 93	2 Bytes	I	**C - - W -**	DPT_Value_Lux	-671088.64 - 670760.96	[CLCx] Setpoint	Setpoint = Value (1-2000)
	2 Bytes	I	**C - - W -**	DPT_Value_Lux	-671088.64 - 670760.96	[CLCx] Setpoint During Day	Setpoint = Value (1-2000)
85, 94	2 Bytes	I	**C - - W -**	DPT_Value_Lux	-671088.64 - 670760.96	[CLCx] Setpoint During Night	Setpoint = Value (1-2000)
86, 95	1 Byte	O	**C T R - -**	DPT_Scaling	0% - 100%	[CLCx] Dimming Value	Dimming Value (%)
87, 96	2 Bytes	I	**C - - W -**	DPT_TimePeriodSec	1-65535	[CLCx] Length of Detection	1-65535 s.
88, 97	2 Bytes	O	**C T R - -**	DPT_Value_Lux	-671088.64 - 670760.96	[CLCx] Luminosity	0-2000 Luxes
98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129	1 Bit	I	**C - - W -**	DPT_Bool	0/1	[LF] (1 bit) Data Entry x	Binary Data Entry (0/1)
130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141,	1 Byte	I	**C - - W -**	DPT_Value_1_Ucount	0 - 255	[LF] (1 byte) Data Entry x	1 byte Data Entry (0-255)

142, 143, 144, 145							
146, 147, 148, 149, 150, 151, 152, 153, 154, 155, 156, 157, 158, 159, 160, 161	2 Bytes	I	**C - - W -**	DPT_Value_2_Ucount	0 - 65535	[LF] (2 bytes) Data Entry x	2 bytes Data Entry
				DPT_Value_2_Count	-32768 - 32767		
				DPT_Value_Temp	-273,00 - 670760,00		
162, 163, 164, 165, 166, 167, 168, 169	4 Bytes	I	**C - - W -**	DPT_Value_4_Count	-2147483648 - 2147483647	[LF] (4 bytes) Data Entry x	4 bytes Data Entry
170, 171, 172, 173, 174, 175, 176, 177, 178, 179	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 bytes) Unsigned
	4 Bytes	O	**C T R - -**	DPT_Value_4_Count		[LF] Function x - Result	(4 bytes) 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 bytes) Signed
	2 Bytes	O	**C T R - -**	DPT_Value_Temp	-273.00 - 670760.00	[LF] Function x - Result	(2 bytes) Float