Motion Detector

Motion Detector with Luminosity Sensor

ZN1IO-DETEC / ZN1IO-DETEC-N / ZN1IO-DETEC-P / ZN1IO-DETEC-X

User Manual Version: [0.3]_a

www.zennio.com

Contents

List of examples

DOCUMENT UPDATES

Version	Changes	Page(s)
[0.3]_a	New option to send multiple objects on Detection / No Detection and Occupied / Not Occupied events.	15, 26
	Optimisation and clarification of the Forced States function.	23
	Optimisation of the presence detection algorithm. Distinction between door opening and door closing.	26 - 31
	Presence detection: new day/night object.	27, 29
	Presence detection: new occupancy state object.	28
	Presence detection: customisable trigger values.	28, 31

1 INTRODUCTION

1.1 ZENNIO MOTION DETECTOR

The **Zennio Motion Detector** is an optional accessory that, connected to any Zennio device equipped with inputs, permits detecting motion (e.g., moving subjects) in the environment of the room where it is installed, by means of the built-in infrared technology.

In addition to the binary motion detection (yes / no), a function for **measuring the luminosity** has been incorporated, which makes it a very versatile accessory.

Finally, it brings the option to notify the KNX bus about errors in the connection or in the detector itself, through the **Short-Circuit** / **Open Circuit** functions.

Important: *the ZN1IO-DETECT-X model does not include the function of the luminosity measurement. The rest of functionality is equivalent to the model ZN1IO-DETEC-P.*

Note: *on certain Zennio devices the luminosity measuring function has not been implemented in software and therefore will not be available even if the model of the detector connected to the device (e.g., ZN1IO-DETEC-P) does incorporate such function. In those cases, the corresponding device user manual (available at www.zennio.com) will notify about it. Please note that the present user manual assumes it to be available, and therefore some options will not apply to certain devices.*

Figure 1. Zennio Motion Detector

1.2 MODELS AND SUPPORTED DEVICES

The Zennio Motion Detector is marketed as an optional accessory for the variety of Zennio devices equipped with inputs and whose application program is specifically compatible with the Zennio motion detector.

The user manuals of the Zennio devices specifically make reference to this functionality, so please refer to them to confirm whether a particular device is compatible with the motion detector or not. Moreover, the behaviour and the options of the motion detection function may differ slightly **depending on the Zennio device** it is connected to. Therefore, the user manual of the motion detector has been particularised for each device. To access the documentation, it is highly recommended to make use of the links provided at the Zennio homepage (www.zennio.com) within the specific section of the particular device being configured.

On the other hand, the Zennio motion detector is marketed under different reference numbers:

 Type A:

- *ZN1IO-DETEC,*

- *ZN1IO-DETEC-N.*

Type A and Type B (switchable):

- *ZN1IO-DETEC-P.*

- *ZN1IO-DETEC-X (identical to the above one, but with no luminosity sensor).*

Note that ZN1IO-DETEC-P and ZN1IO-DETEC-X can work under two different voltage modes, noted as "Type A" and "Type B". This is possible by means of a specific mode selection micro-switch (see section 2.1), which should be configured according to the particular Zennio device the detector is being connected to.

For information about the type (A or B) that applies to each Zennio device, please refer to its **Datasheet** (available at www.zennio.com) or contact the Zennio Technical Support.

2 INSTALLATION

2.1 GENERAL INSTALLATION

The Zennio Motion Detector needs to be connected to any of the available input ports of the Zennio device (please refer to the specific user manual of the target device for further details).

Once connected to the device, the detector needs no further electrical power; it will be ready to function once the Zennio device gets programmed.
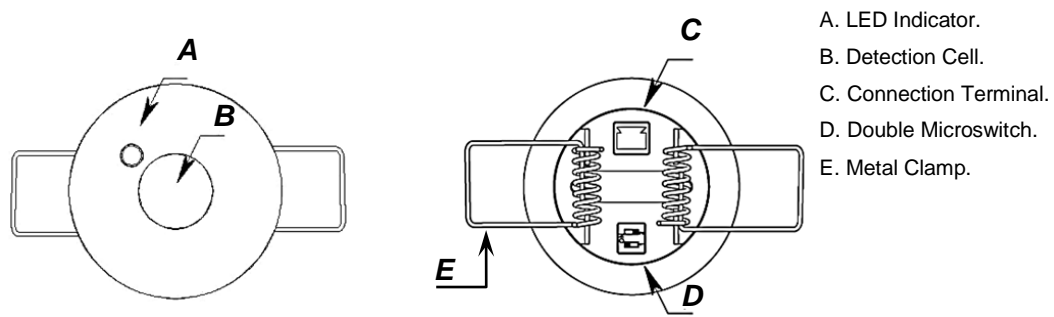

A. LED Indicator.
B. Detection Cell.
C. Connection Terminal.
D. Double Microswitch.
E. Metal Clamp.

Figure 2. Element Diagram (models ZN1IO-DETEC and ZN1IO-DETEC-N)

A. LED Indicator.
B. Detection Cell.
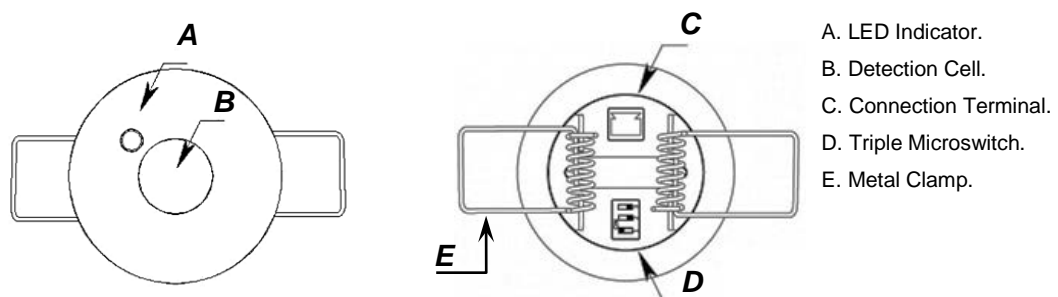C. Connection Terminal.
D. Triple Microswitch.
E. Metal Clamp.

Figure 3. Element Diagram (models ZN1IO-DETEC-P and ZN1IO-DETEC-X)

Figure 2 and Figure 3 show the main elements of the device:

- **LED Indicator** (A): LED light that will blink in red colour when motion is detected.

  ➢ The **luminosity** levels are obtained from the light that strikes this hole.

- **Detection Cell** (B): cell for the **motion** detection.

- **Connection Terminal** (C): slot for inserting the device connecting cables.

- **Double/Triple Micro-switch** (D):

  ➢ Models ZN1IO-DETEC and ZN1IO-DETECT-N

    These two switches enable or disable the luminosity measurement (switch #1) and the blinking of the LED on the detection of motion (switch #2).

1 2	**1 2**	**1 2**	**1 2**
LUM On	**LUM** Off	**LUM** On	**LUM** Off
LED On	**LED** Off	**LED** Off	**LED** On

    Figure 4. Positions of the Double Micro-Switch

  ➢ Models ZN1IO-DETEC-P and ZN1IO-DETEC-X

    These switches allow enabling or disabling the luminosity measurement (switch #1; only functional in model ZN1IO-DETEC-P) and the blinking of the LED on the detection of motion (switch #3). Switch #2, on the other hand, allows selecting the Type A or Type B working modes, depending on the Zennio device the detector is being connected to.
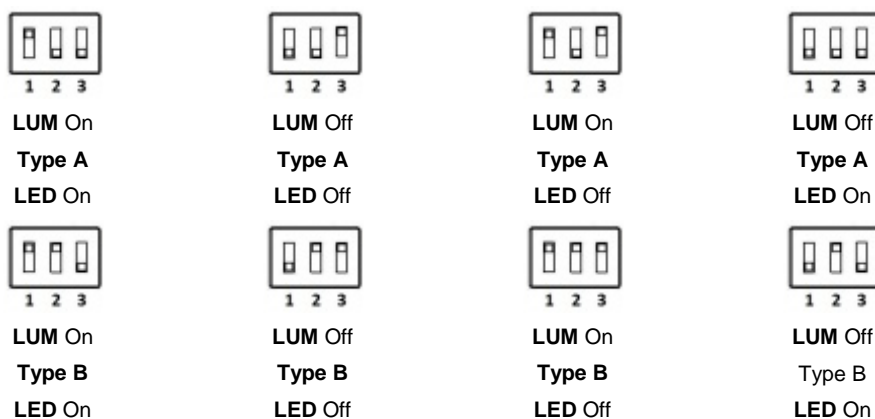
1 2 3	**1 2 3**	**1 2 3**	**1 2 3**
LUM On	**LUM** Off	**LUM** On	**LUM** Off
Type A	**Type A**	**Type A**	**Type A**
LED On	**LED** Off	**LED** Off	**LED** On
1 2 3	**1 2 3**	**1 2 3**	**1 2 3**
LUM On	**LUM** Off	**LUM** On	**LUM** Off
Type B	**Type B**	**Type B**	Type B
LED On	**LED** Off	**LED** Off	**LED** On

    Figure 5. Positions of the Triple Micro-Switch

    Important: *please refer to section 0 for information about the differences between all the models and their compatibility with the different Zennio devices*.

- **Metal Clamps** (E): spring clamps that help secure the detector after the installation.

It is possible to accommodate the motion detector in the **false ceiling** of the room by drilling a 40-mm diameter corona and inserting the detector (previously connected to the device) with its metal clamps folded (see section 2.2).

For installation tips and suggestions, it is also important to review the "**Installation**" technical note, available at the Zennio homepage (www.zennio.com).
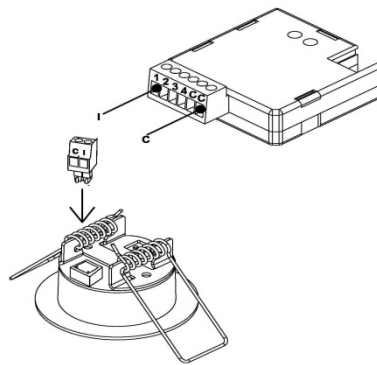
2.2 CONNECTING THE DETECTOR TO THE DEVICE

Figure 6. Connection to the Device

During the connection of the detector to the device, the terminal labeled as "**I**" on the detector side needs to meet the desired input slot (1, 2, etc.) on the side of the target device, while the terminal labeled as "**C**" on the detector plug should be connected to the common input slot (identified as well as "C") in the target device.

2.3 CONNECTING MULTIPLE PARALLEL DETECTORS

It is possible to connect multiple parallel detectors (typically two, but may be more depending on the model) to **the same input port** of the device, so that a **wider detection area** is covered, while both detectors still work (and are configured) as if they were only one detector, which does not apply when two input ports are used.

Important: *please refer to the datasheet of the motion detector for the **maximum number** of them that can be connected in parallel to the same input.*

Figure 7 illustrates this type of assembly, with one of the two wires of each detector (the one labeled as "**I**") connected to a particular input of the device and the remaining two wires (labeled as "**C**") connected to the common input port.

However, under this set-up it is mandatory that the luminosity measuring function **is not active in more than one detector** (see section 2.1). Otherwise the measurements sent by the detectors may interfere each other.
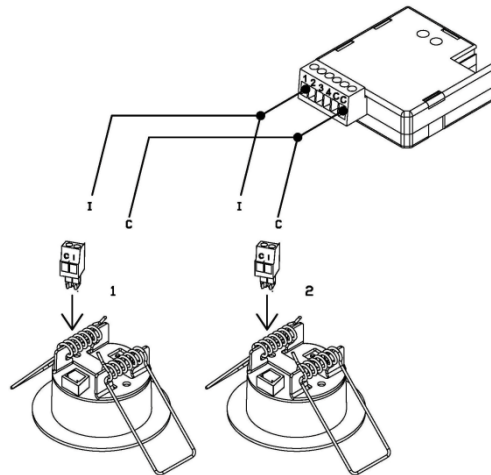

Figure 7. Connecting Two Parallel Detectors

Note: *Zennio devices compatible with the motion detector typically provide the option to enable multiple virtual detection channels, which makes it possible to simultaneously implement a variety of reactions and behaviours, although all of them associated to a sole input value. The number of the available virtual channels is not related at all with how many detectors are connected. Moreover, from the hardware point of view, multiple detectors connected to the same input are considered as only one detector.*

For detailed information about the technical features of the motion detector, as well as on security and installation procedures, please refer to the device **Datasheet**, bundled within the device packaging and also available at www.zennio.com

For installation tips and suggestions, it is also important to review the "**Installation**" technical note, available at the same address.

Important: *the next sections of this manual refer to different functions related to the measurement of the luminosity. Please ignore all them for model ZN1IO-DETEC-X.*

3 ETS PARAMETERISATION

3.1 INPUT

Once the corresponding input of the device has been configured to work as a motion detector (please refer to the user manual of the device itself for further details), the "Configuration" tab shows up. Figure 8 illustrates this.

Moreover, a set of communication objects are available by default:

- **[Ix] Short Circuit Error**: 1-bit object that will notify the KNX bus (by sending a "1" every 30 seconds) about short-circuit events in the detector wiring or in the detector itself. Once solved, the value "0" will be sent (once) through the same object.

- **[Ix] Open Circuit Error**: 1-bit object that will notify the KNX bus (by sending the value "1" every 30 seconds) about open-circuit events in the detector wiring or in the detector itself. Once solved, the value "0" will be sent (once) through the same object.

- **[Motion Sensor] Scene Input**: 1-byte object through which it will be possible to receive scene values (0 – 63, both included) from the bus.

- **[Motion Sensor] Scene Input**: 1-byte object through which it will be possible to send scene values (0 – 63, both included) to the bus.

Note: *object names may differ slightly depending on the input port where the motion detector has been connected.*
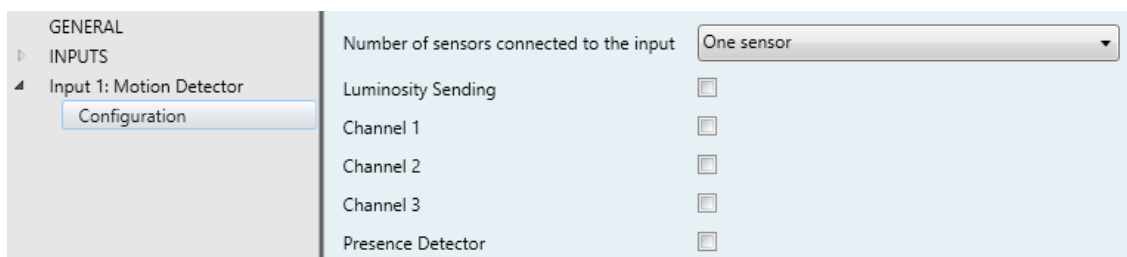

Figure 8. "Configuration" Tab

The following parameters are available in the Configuration tab:

.⁰● **Number of sensors connected to the input**: sets the number of sensors (one or two) connected to the same input, in order to adequately calibrate the luminosity and detection readings. Please see section 2.3 for more details.

.⁰● **Luminosity Sending**: activates or deactivates the option to periodically send the luminosity level to the KNX bus (in terms of percentage, through object "**[Ix] Luminosity**"). If checked, the following parameters show up:

  ➢ **Period**: period for sending the luminosity level to the bus, between 0 and 255 seconds. If set to 0, the periodical sending remains disabled.

  ➢ **Luminosity change to send**: sets a certain increment (in percentage) so that when two consecutive readings of the luminosity are found to differ by more than such value, an extra sending of the luminosity to the bus will take place. If set to "0", this type of transmission is disabled.

Number of sensors connected to the input	One sensor	
Luminosity Sending	☑	
Period (0 = Disabled)	15	x 1 s.
Luminosity change to send (0 = Disabled)	0	%

Figure 9. Luminosity Sending

As stated, the luminosity level is sent through the "**[Ix] Luminosity**" 1-byte object. The higher the light level detected in the room is, the greater the value of this object will be.

> **Note**: *bear in mind that during continuous motion detection, the value of the luminosity level may take a little more time to update, as both signals (detection and luminosity) share the same input port of the device.*
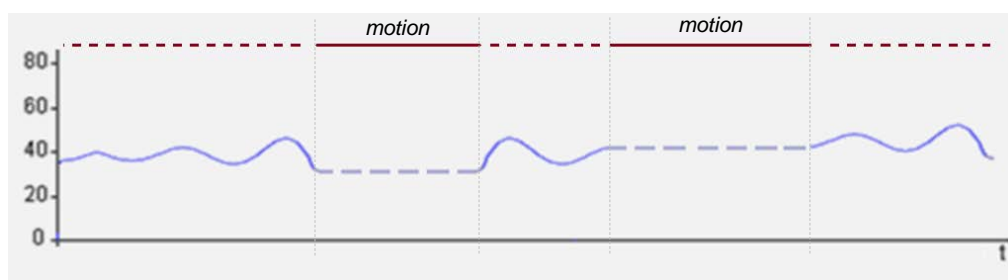

Figure 10. Luminosity Readings during Motion Detection Events.

- **Channels 1-X**: activates or deactivates the different virtual detection channels. Every channel will work independently, which makes it possible to set different parameters (delays, thresholds, etc.) and different parallel reactions to the readings sent by a sole physical detector.

- **Presence Detector**: activates or deactivates the presence detection function.

3.2 CHANNELS

Each input allows enabling **up to three motion detection channels**. These channels share the same *hardware status*, i.e., they read the same information about motion, luminosity and errors in the input. However, each of them can have a different configuration in relation to:

- Motion detection conditions,

- Values sent when entering the "Detection" and "No Detection" states.

- Channel lock.

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

It will be possible to define a **Detection Length** and a **Blind Time** to ensure the channel remains in the new state for at least a certain time, after which it will listen again to the signal received from the sensor. The following example illustrates this.

Example: Detection States.

The graph below shows the following succession of events:

- *At t_1, the sensor activates the motion signal. The channel switches then to the "Detection" state and sends a "1" through "**[Ix][Ci] Detection State**", making for example a lamp turn on.*

- *At t_2, motion is no longer detected; however the channel remains in "Detection" and starts counting the parameterised Detection Length time (T_1).*

- *At t_3, the channel switches to "No Detection" and sends a "0" (making the lamp turn off). After that, it starts counting the parameterised Blind Time (T_2).*

- *Although at t_4 the sensor starts reporting motion again, the channel does not switch to "Detection" until T_2 ends, that is, until t_5.*

- *At t_6 the sensor stops reporting motion, so the channel starts a new Detection Length count.*

- *At t_7, before T_1 concludes, motion is reported again, so the count for the Detection Length is interrupted, remaining the channel in "Detection" without having switched to "No Detection" between t_6 and t_7.*
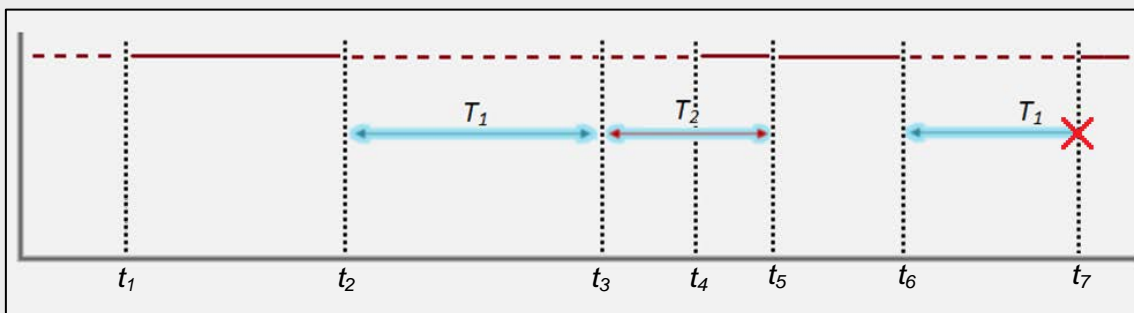
Figure 11. *Channel Detection States vs. Motion Signal*

When a channel is enabled, ETS displays a new tab ("**Channel i**"), which is divided into several parts.
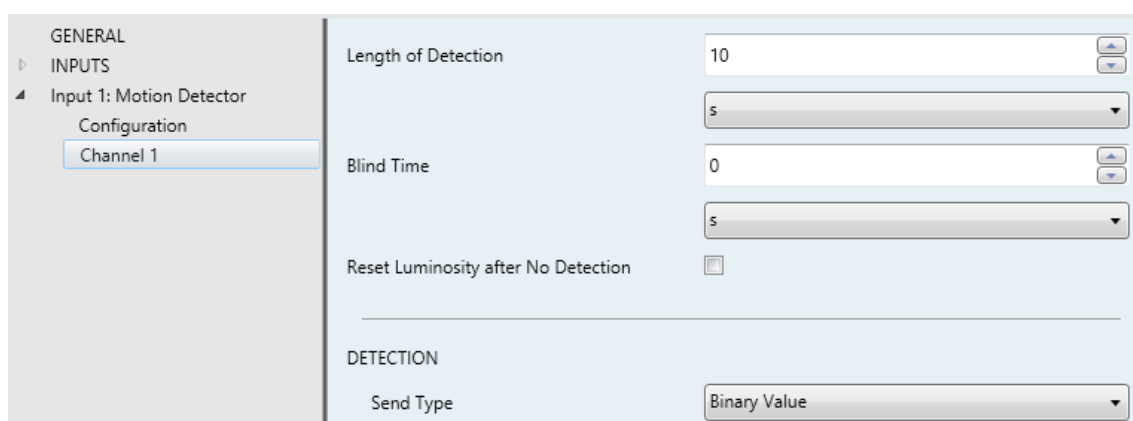

Figure 12. Channel Configuration

The upper parameters of this tab basically define the aforementioned delays:

- **Length of Detection**: sets the minimum time that should elapse without any motion before the channel switches to the "No Detection" state. Further motion events will interrupt this countdown. The range is 1 to 255 seconds, 1 to 255 minutes and 1 to 18 hours.

> **Example**: Detection Length.
>
> *One MAXinBOX 66 and one motion detector are being used to turn on/off a light source (which is connected to a KNX light dimmer) depending on whether motion is being detected or not in the room.*
>
> *If the detection length is set to 5 seconds, as soon as the detector finds moving subjects in the room, MAXinBOX 66 will send the "Detection" value to the dimmer, so that the light source turns on. After five seconds without detecting motion again (even if the subject is still in the room), MAXinBOX 66 will send the "No Detection" value to the dimmer, so the light source turns off.*
>
> *On the other hand, if a 60-second detection length is parameterised, the light will remain on <u>at least</u> for 60 seconds, as 60 seconds of no detection are necessary before sending the "No Detection" value. The greater the detection length is, the more evidence that the subject is not in the room is required before turning the lights off.*

- **Blind Time**: sets the time margin during which the channel will remain inactive whenever it enters the "No Detection" value. This ensures that the "No Detection" state is maintained at least during that time interval, even if there is motion in the room, which will be ignored by the channel. The range is 1 to 255 seconds, 1 to 255 minutes and 1 to 18 hours.

- **Reset Luminosity after No Detection**: enabling this parameter will set the "**[Ix] Luminosity**" object to 0% whenever the channel leaves the "Detection" state and enters the "No Detection" state. There is also the option to configure a delay.

  ➤ **Reset Delay**: imposes a delay before resetting the value of the luminosity object. The range is 0 to 60 seconds.
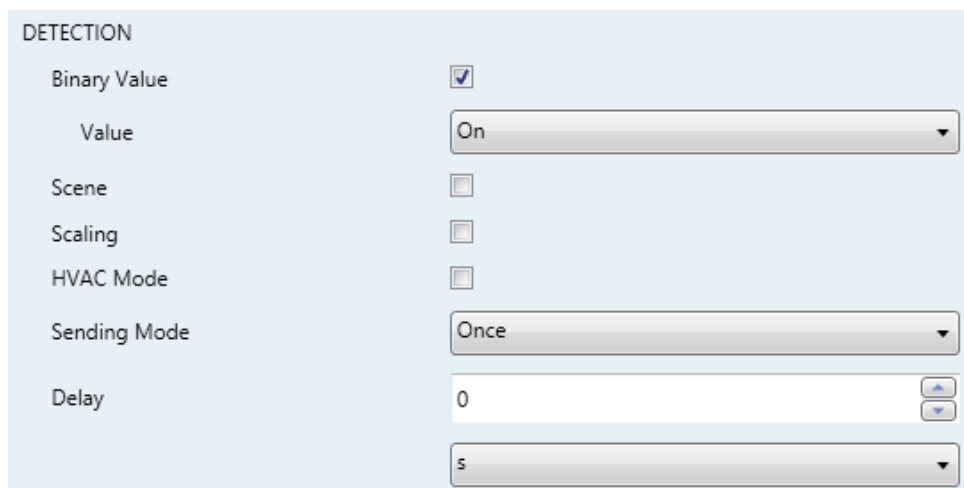
  This parameter is useful when combined with the **Luminosity-Constrained Detection** function. Therefore it is advisable to read the examples about such function in later sections of this manual.

3.2.1 DETECTION

These parameters configure what the device should send to the bus when the channel enters the Detection and No Detection states:

- A **binary** value,

- A **scene**,

- A **scaling** (percentage) value,

- An **HVAC** mode,

- Any combination of the above cases.

The same parameters are available for Detection and No Detection (however their defaults may be different).

Figure 13: "Channel I" – detection function

- **Binary Value**: enables or disables the sending of a binary value (either "<u>On</u>" or "<u>Off</u>", according to parameter **Value**) through object "**[Ix] [Ci] Detection Status (Binary)**".

- **Scene**: enables or disables the sending of a scene trigger value (1 to 64, according to parameter **Value**) through object "**[Ix] [Motion Detector] Scene Output**".

- **Scaling**: enables or disables the sending of a scene trigger value (1 to 64, according to parameter **Value**) through object "**[Ix] [Ci] Detection Status (Scaling)**".

- **HVAC Mode**: enables or disables the sending of an HVAC special-mode trigger value through object "**[Ix] [Ci] Detection Status (HVAC)**". The desired HVAC mode can be set through parameter **Value**, which provides the following options:

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

Table 1. HVAC Modes

- **Sending Mode**: sets whether the above values should be sent "Once" or be "Cyclic". In case of selecting the latter, one more parameter will show up:

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

- **Delay**: sets the time that the device should wait before sending the values to the bus, once the channel enters the Detection state. The available values are 0 to 255 seconds, 0 to 255 minutes and 0 to 18 hours.

3.2.2 NO DETECTION

The options are analogous to those for Detection (section 3.2.1). The values will be sent through the same objects in both cases (Detection and No Detection).

3.2.3 CHANNEL LOCK

While the channel remains locked, any further sendings to the bus related to that channel will be interrupted, no matter if motion detections take place or not. It is possible to start or end the lock state through a binary object ("**[Ix][Ci] Lock**") or a scene object ("**[Ix][Motion Detector] Scene Input**").

The parameters available for this function are:

Figure 14: Channel – lock function.

- **Enable / Lock**: sets how to activate or deactivate the channel lock.

  - Always Unlocked: the channel will always remain active.

  - Lock using 1-bit Object: the channel will switch from active to inactive (or vice versa) according to the value received through the "**[Ix][Ci] Channel Lock**" object.
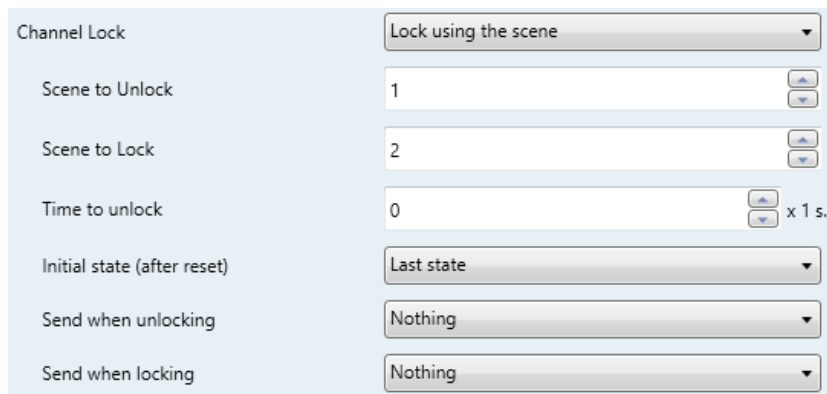

Figure 15. Lock using 1-bit object

Selecting this option also requires setting the following parameters:

  - **Configuration**: sets the value (0 / 1) that will switch the channel to unlocked and the value that will switch it to locked. The options are "0 = Unlock; 1 = Lock" and "0 = Lock; 1 = Unlock".

  - **Time to Unlock**: sets a delay between the reception of the value and the actual re-activation (unlock) of the channel. The available range is 0 to 255 seconds.

  - **Initial Status (after reset)**: sets the lock or unlock state of the channel at the device start-up or after a bus power failure: "Last state", "Unlocked", "Locked". On the very first start-up, the last state is assumed as unlocked.

o **Send when unlocking**: sets a value to be sent to the bus once the channel switches to the unlocked state, to notify that the detection process is being resumed. "Nothing", "No detection", "Detection". The last two correspond respectively to the values (binary, scene, etc.) set for "Detection" and "No Detection", as explained in 3.2.1.

o **Send when locking**: analogous to the above parameter; sets the value to be sent to the bus when the channel becomes locked. "Nothing", "No detection", "Detection".

➢ Lock using the Scene: the channel will switch between the unlocked and locked states according to the values received through the "**[Motion Sensor] Scene Input**" object.

Figure 16. Lock using the Scene

Selecting this option also requires setting the following parameters:

o **Scene to Unlock**: sets the scene number (between 1 and 64) that will activate the channel.

o **Scene to Lock**: sets the scene number (between 1 and 64) that will deactivate the channel.

o **Time to Unlock**, **Initial State**, **Send when Unlocking** and **Send when Locking**: these parameters are analogous to those already described for "Lock using 1-bit object".

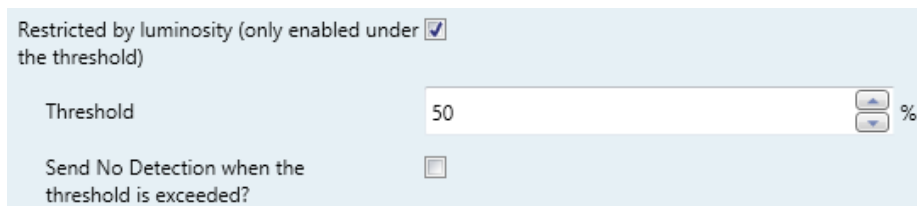## 3.2.4 RESTRICTING THE SENDINGS DEPENDING ON THE LUMINOSITY

The following parameters allow restricting the behaviour of the channel according to the luminosity readings:

- **Restricted by Luminosity**: if enabled, the value corresponding to "Detection" will only be transmitted to the bus in case the luminosity level during the detection is **lower than a certain threshold level**.

  <u>Note</u>: *the value corresponding to "No Detection" will always be sent, no matter if the luminosity level is over the threshold value or not.*

  Enabling this option requires setting the following parameters as well.

  ➢ **Threshold**: luminosity percentage over which the channel will stop sending detections.

  ➢ **Send No Detection when the Threshold is Exceeded?**: if enabled, the "No Detection" value will be sent to the KNX bus as soon as the luminosity is found over the threshold value.

**Figure 17.** Detection restricted by luminosity

<u>Important</u>: *parameterising inadequate threshold values may cause an undesired behaviour of the device.*

Example: Luminosity-Constrained Detection

*A source of artificial light needs to be switched on or off depending on whether there is motion or not in the room and on the available sunlight. Thus, a Luminosity-Constrained Detection is configured, with a threshold value of **50%** and with the "**Send no detection when the threshold is exceeded**" option active.*

1) *At night, the luminosity level stays at **10%**.*

2) *Motion is detected at 6:00h am. The light source **turns on** as luminosity < 50%, causing a rapid increase of the luminosity, which keeps rising due to the sunrise.*

3) *Further motion is detected every few seconds, so the light source remains on.*

4) *At 6:30h, after the dawn, the luminosity level is already about **60%**. Therefore, the artificial light source turns off ("No Detection" is sent as the threshold value has been exceeded) and therefore the level lowers to **55%**.*

5) *The light source remains off no matter if there are still moving subjects in the room, as the luminosity is in any case over the threshold value (**55%**).*

*It becomes clear that the essence of this example (where "No Detection" has been configured to be sent after exceeding the threshold value) consists in **setting a threshold value** which is greater than the luminosity level provided by the natural sunlight (here, 55%) and in ensuring the artificial light source itself does not cause (during the absence of the sunlight) a luminosity level greater than such value, either.*
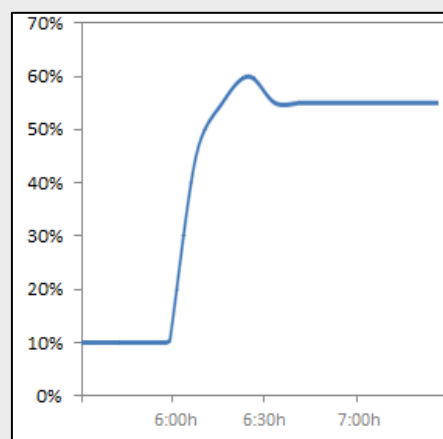
The already mentioned **Reset Luminosity after No Detection** function guarantees that, after a switch-off due to the sending of a no-detection, the device will notify any further detection in any case, even if it has not been able to verify the updated luminosity value yet. Please see the following example.

Example: Luminosity Reset.

A light source needs to be turned on or off depending on the motion detection and on whether it is daytime or night.

*Hence, a luminosity detection constrained to a **threshold of 30%** is configured, although "**Send no detection when the threshold is exceeded**" is disabled.*

1) *If somebody enters the room at night, the light source will turn on, thus making the luminosity raise to **70%**. Such value will be afterwards measured by the device. As the device was NOT set to send a "No Detection" after surpassing the threshold, the light source will remain on.*

2) *After a while with no motion, the "No Detection" is sent, which turns the light off.*

3) *If motion is again detected <u>immediately after</u> (before the device has had enough time to detect the darkness), the "Detection" will not be sent to the bus until the device can check the updated luminosity (see "Luminosity Sending", in 3.1).*

*To prevent the above situation, it is possible to parameterise a **luminosity reset** to 0% after the "No Detection".*

Example: Luminosity Reset (delayed).

Let the light source of the above example implement a soft (progressive) switch-off. As above, the switch-on and the switch-off are required to depend on the motion detection and on whether there is natural sunlight in the room or not.

*Again, a **threshold of 30%** is configured, <u>without a "No detection" sending after exceeding the threshold value</u>.*

4) *If somebody enters the room at night, the light source will turn on, making the luminosity raise to **70%**. Such value will be afterwards measured by the device. As*

---

the device was NOT set to send a "No Detection" after surpassing the threshold, the light source will remain on.

5) *After a while with no motion, the "No Detection" is sent, making the progressive switch-off of the light source start.*

6) *If* **Reset the luminosity after No Detection** *was set to "Yes", the device will assume a luminosity of 0% from that moment. However, as no motion is being detected, the device will receive further luminosity values (e.g., 60%) during the progressive switch-off, thus overwriting that 0%.*

7) *This may cause that a detection taking place before the end of the switch-off is not notified to the bus, as 60% > 30%. Therefore, the light will finally switch completely off, but* **the device will not be aware of the darkness** *until it receives, some instants later, the updated luminosity (see "Luminosity Sending" in 3.1).*

To prevent the above situation, it is possible to impose a **delay** *to the luminosity reset, so it remains at 0% once the progressive switch-off is over and therefore preventing that value from being overwritten.*

Note: *as stated in previous pages, the* **luminosity reset** *option only applies to transitions from the "Detection" to the "No Detection" states and when they are due to events of the sensor. Therefore, the luminosity is not reset in the following cases:*

- *After sending an extra "No Detection" due to exceeding the luminosity threshold (parameterisable behaviour).*

- *After sending a "No Detection" due to a transition from the "No Detection" state towards itself (i.e., if a detection had been reported by the sensor but the channel remained in the "No Detection" state because of a luminosity value greater than the threshold, the luminosity will not be reset even if "No Detection" is still reported to the bus).*

3.2.5 FORCED STATES

This parameter block refers to the configuration of the forced-state object. Values received through this object will be interpreted by the device as a master detection or no-detection signal (i.e., **this object permits externally overwriting the motion detection signal**).
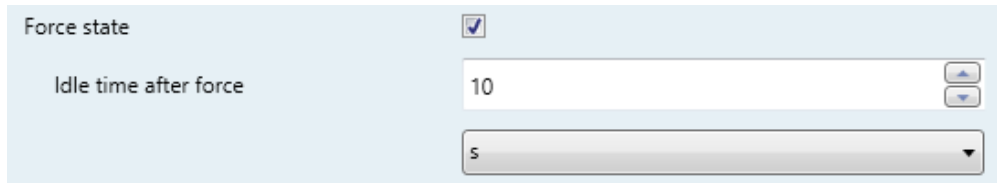
Force state	☑	
Idle time after force	10	
	s	▾

Figure 18: Forced States

The behaviour when a forced-state value is received from the bus is as follows:

An "on" is received:

- If the channel was previously in the "Detection" state, no action will be taken.

- If the channel was previously in the "No Detection" state, it will enter the "Detection" state. The detection delay (see 3.2.1) will also apply here, analogously as when it is the actual sensor what triggers the "Detection" state.

An "off" is received:

- If the channel was previously in the "Detection" state, it will enter the "No Detection" state. The configured no-detection delay (see 3.2.2) will also apply here, analogously as when it is the actual sensor what triggers the "No Detection" state.

- If the channel was previously in the "No Detection state", no action will be taken.

After forcing a state, the device remains under that state during a **configurable time**. Once this preiod expires:

- <u>If the forced state was "No Detection"</u>, the device will still remain under that state without performing any sendings, until motion is detected again.

- <u>If the forced state was "Detection"</u>,

> If motion did not take place during the last T seconds (being T the detection time), the device will switch to "No Detection" and perform the corresponding sending, unless the channel is found locked.

> If there was any motion during the last T seconds (being T the detection time), the device will remain under "Detection", without performing any sending to the b us.
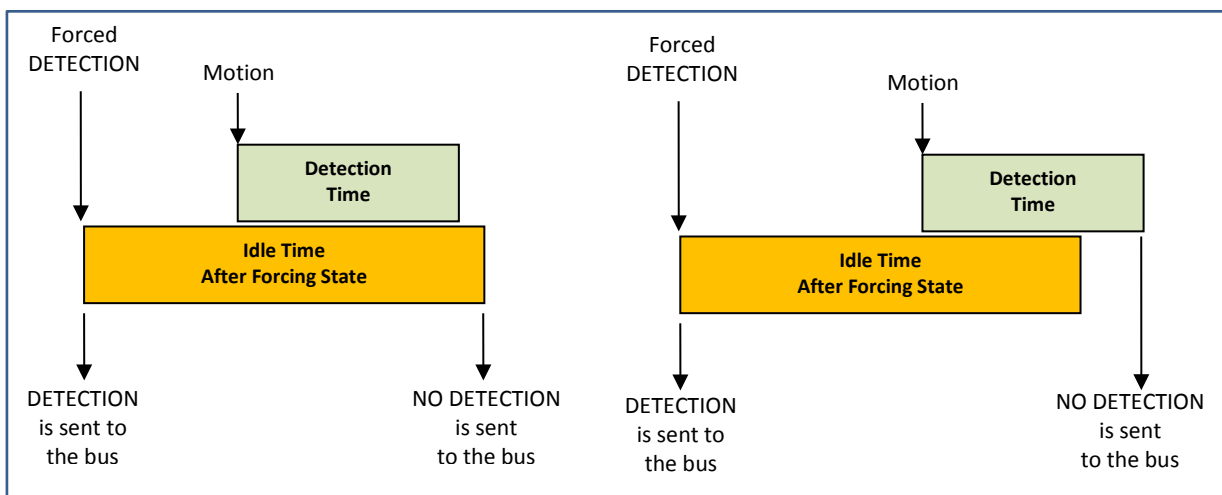

Figure 19: Forced States (example)

Forced states have **a higher priority** over any other functionality, and are executed unconditionally with independence of the lock state of the channel and of the state of the detector.

The parameters and objects involved in this functionality are:

- **Force state**: enables or disables the "**[Ix][Ci] Force state**" one-bit object.

  > **Idle Time after Force**: sets the time interval during which the channel will remain in the forced state. The available values are 1 to 255 seconds, 1 to 255 minutes and 1 to 18 hours. Note that forced states ignore the length of detection and blind time that may have been configured (see section 3.2).
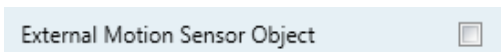
3.2.6 EXTERNAL MOTION DETECTION

Figure 20. External Motion Detection

This parameter lets enabling or disabling a specific communication object ("**[Ix] External Motion Detection**") for the reception of motion detections from external KNX

devices, so that various devices can combine their states and perform a joint response.

When one "1" is received through this object, the channel will act the same way as if the sensor itself had detected motion: the detection object will be sent to the bus and the detection length will start counting.

Keep in mind that if the remote device is not configured to **periodically re-send** that "1" to this object, the local detection channel will leave the "Detection" state as soon as the detection length time expires.

Note: *one external detection object is provided per input configured as a motion detector. Therefore, it will affect all channels enabled for such input.*

3.3 PRESENCE DETECTION

Apart from the motion detection channels, it is possible to enable the presence detection function. The difference between *motion* and *presence* is important:

- The detection of **motion** does not necessarily imply **presence** of humans (there may be objects moving in the room).

- The non-detection of motion does not necessarily imply **non-presence** of humans (they may be sleeping, for example).

The domotic system can hardly deal with this distinction. However, it is possible to monitor subjects entering / leaving the room (and therefore determining whether there is **human presence inside or not**) by making use of a set of motion detectors (one of them acting as a **master**, and the others as **slaves**) and of state sensors at the doors of the room.

To better illustrate how this functionality works, a practical example has been included at the end of this section (3.2.3).

Figure 21: "Presence Detector" tab

Once the "**Presence Detector**" has been enabled in the "Configuration" tab, a new screen containing the corresponding parameters will show.

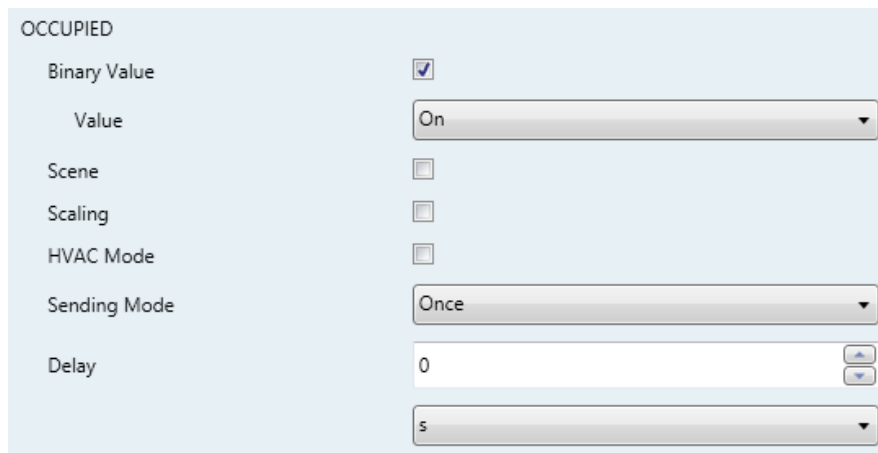The main parameter to be set is the role type assigned to this detector:

- **Type**: sets the detector as a master or a slave.

The remaining parameters are explained in sections 3.3.1 and 3.3.2.

3.3.1 MASTER

The master detector will bring together the information received from the slaves (through object "**[Ix] Presence: Slave Input**") and its own detections, and therefore determine whether there is presence ("**Occupied**") or not ("**Not Occupied**").

The transitions between the two states will imply the sending of a certain value to the KNX bus, which is configured in parameters:

Figure 22: Sendings for the "Occupied" state.

These options are analogous to those in the detection function (section 3.2.1), although in this case the values will be sent through object "**[Ix] Presence State (Z)**", where "Z" depends on the send type selected (unless it has been set to "Scenes"; in that case, the value will be sent through "**[Motion Sensor] Scene Output**").

Note: *the above parameters allow a daytime configuration and an overnight configuration (through parameter **Day/Night**), as explained later in this section.*

The presence detection algorithm itself works as follows:

1. The room will begin in the "Not Occupied" state.

2. When a **door opening** is received (through the **trigger object**):

   - If motion is detected either by the master or the slaves, the "**Occupied**" state will be adopted (if not already active).

3. When a door closing is received (through the trigger object):

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

   - Next, the master starts counting the **listening time**.

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

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

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

4. Back to point 2).

This algorithm depends on the following parameters:

- **Trigger Value**: sets the specific values of object "**[Ix] Presence Trigger**" that will be associated to the two door states: "0 = Closed Door; 1 = Open Door" or "0 = Open Door; 1 = Closed Door".

- **Waiting Time**: sets the value of the waiting time, which starts counting once the closed-door trigger value is received. The range is 0 to 65535 seconds (2 seconds by default). This can be overwritten in runtime by sending the new value to object "**[Ix] Presence: Waiting Time**".

- **Listening Time**: sets the value of the listening time, which starts counting once the waiting time expires. The range is 0 to 65535 seconds (2 seconds by default). This can be overwritten in runtime by sending the new value to object "**[Ix] Presence: Listening Time**".

- **Occupancy State Object**: enables or disables the "**[Ix] Presence: Occupancy State**" binary object, which can be read anytime to obtain whether the current state is Occupied (value "1") or Not Occupied (value "0").

• **False Non-Occupied Detection**: enables or disables the False Non-Occupied detection algorithm (see section 3.3.1.2).

• **Day/Night**: brings the option to distinguish a daytime configuration and an overnight configuration for the sendings to be performed during the transitions from "Occupied" to "Not Occupied" or vice versa. When enabled, the corresponding parameters will be duplicated, and the "**[Ix] Presence: Day/Night**" binary object will be added to the project topology, so the day/night transitions can be received.

  ➢ **Configuration**: sets which values will be associated to the beginning of the day and the beginning of the night: "0 = Day; 1 = Night" or "0 = Night; 1 = Day".

• **Presence Detection Lock**: provides a procedure for temporarily disabling the presence detection function. See section 3.3.1.1.

3.3.1.1 PRESENCE DETECTION LOCK

The presence detection can be locked by means of a communication object, analogously as it happens with the motion detection channels (see section 3.2.3). The lock and unlock trigger values can be either binary (through object "**[Ix] Presence: Detection**") or scene values (through object "**[Motion Detector] Scene Input**"). Please refer to section 3.2.3, as the behaviour of this function and the parameters involved are analogous to those detailed there.

While the presence detection remains locked, it will ignore any event it may be related to, such as motion detections or receptions of the trigger object.

Figure 23: Sendings for the "Occupied" state.

Note: *after leaving the lock state, the room is assumed to be "Not Occupied".*

3.3.1.2   FALSE "NOT OCCUPIED" DETECTIONS

To prevent undesired detections of no-presence, it is possible to send the KNX bus an order to **save the scene** as soon as the trigger value is received, provided that the current presence state is "Occupied". After the listening time, if no further detections have taken place, the presence detector will switch (as usual) to "Not Occupied", and send the bus the corresponding value.
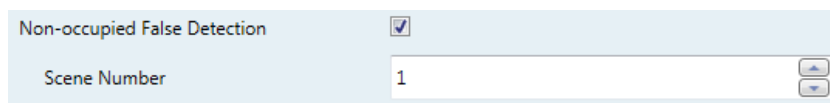
In these circumstances, if **motion is detected again without a new reception of the trigger value**, then the above situation will be considered as a false "Not Occupied" detection (as there was still presence in the room). An order will be sent to **play back the scene** previously saved (this will make the room recover the state it had prior to sending "Not Occupied") and the detector will switch back to "Occupied".

> ***Example:*** *false "non-occupied" detections.*
>
> *Suppose a room with two people sleeping inside (state "Occupied"). If the false non-occupied detections are being ignored, when one of them leaves, the room will switch to "Not occupied". Afterwards, when the other person makes a movement, the room will switch to "Occupied" and the corresponding order will be sent, so for example the lights will turn on. On the contrary, with this option in use, such action will not take place when the person in the room makes a movement, as the trigger value (i.e., the door being opened) has not been received again.*

The only parameters of this function are:

- **Non-Occupied False Detection**: enables or disables this functionality.

  - ➢ **Scene Number**: number of the scene (1 – 64) to be used.

Figure 24. False Detection of the Non-Occupied event
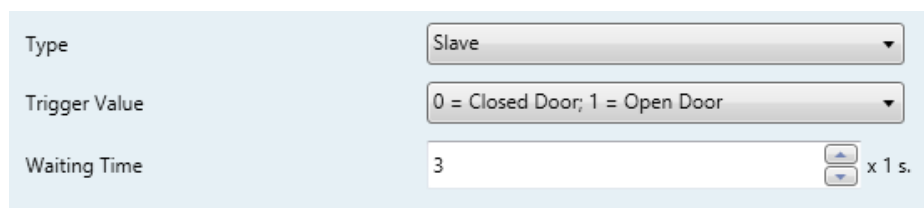
3.3.2 SLAVE

Slave detectors notify the master detector about detection and no-detection situations by sending one "1" or one "0", respectively, through the "**[Ix] Presence: Slave Output**" object. However the only way a slave can switch from the detection state to the no-

detection state is after the reception of a trigger signal (i.e., after a door closes). Therefore, slave detectors also make use of the "**[Ix] Presence Trigger**" object.

After the trigger signal is received, the slave will let a certain **waiting time** (configurable) elapse. The time count will be restarted if further triggers are received. Once the count ends, the slave will switch to no-detection.

Once in no detection, as soon as motion is detected the state will switch to detection.

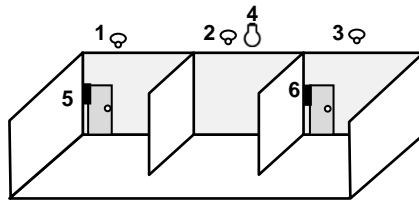Both the **trigger value** (0 / 1) and the **waiting time** can be configured through the homonymous parameters.

Type	Slave
Trigger Value	0 = Closed Door; 1 = Open Door
Waiting Time	3 x 1 s.

Figure 25. Slave detector

3.3.3 PRACTICAL EXAMPLE

1.	Slave Detector #1.
2.	Master Detector.
3.	Slave Detector #2.
4.	Lamp.
5.	Door Sensor #1.
6.	Door Sensor #2.

The above figure shows a scenario with two slave motion detectors, one master motion detector, two door sensors and one lamp, which is required to be controlled automatically, depending on the presence detection.

The involved objects for proper presence detection are:

A. "**[Ix] Presence: Slave Output**", from the device where the **slave detector #1** is connected to.

B. "**[Ix] Presence: Slave Output**", from the device where the **slave detector #2** is connected to

C. "**[Ix] Presence: Slave Input**", from the device where the **master detector** is connected to.

D. "**[Ix] Presence: Trigger**", from the device where the **master detector** is connected to.

E. "**[Ix] Presence: Trigger**", from the device where the **slave detector #1** is connected to.

F. "**[Ix] Presence: Trigger**", from the device where the **slave detector #2** is connected to.

G. "**[Ix] [Switch/Sensor] Edge**" (or equivalent object), from the device where the **left door** sensor is connected to.

H. "**[Ix] [Switch/Sensor] Edge**" (or equivalent object), from the device where the **right door** sensor is connected to.

I. "**[Ix] Presence State (Binary)**", from the device where the **master detector** is connected to.

J. The **on/off control object** of the dimmer that controls the lamp.

One group address (A1) is required for objects A through C, another one (A2) for objects D through H, and another one (A3) for objects I and J.

In the above situation, the following sequence of events can be reproduced:

- Being the room empty and therefore the presence detection in state "**not-occupied**", the door on the left opens and somebody enters. The device where the left door sensor is connected to will send the trigger signal through A2 and it will be received by the slave detector no. 1 which afterwards will notify detection through A1.

- The device where the master detector is connected to will receive both the trigger (A2) and the detection (A1) and therefore will activate the "**occupied**" state, which will make the lamp turn on (A3).

- Another person enters the room through the door on the right side. The corresponding sensor and the slave detector no. 2 will react the same as above. However, since the presence detection in the device of the master detector was already in "**occupied**", nothing will happen in the lamp.

- Afterwards, one of the two people leaves the room (while the other person remains inside, in the central section of the room). Therefore, the trigger will be sent and then the two slaves will switch to "no-detection".

- As long as any of the detectors (master or slaves) keeps detecting motion, the "**occupied**" state will be maintained and therefore the lamp will not be switched off.

- If the person that remains inside finally leaves the room (no matter through which door), the slave and master detectors will definitely remain in "no-detection". After the *listening time*, the presence detection will change to "**not-occupied**", so the lamp will be switched off.

Note that if the person that remains in the room is sleeping when the other one leaves, the "**not-occupied**" state will be adopted. Therefore, any movement made afterwards by the person who is sleeping will activate back the "occupied" state, which will make the lamp turn on.

To prevent this, the **False "Non-Occupied" Detection** function can be activated. This way, when the first person leaves, an order will be sent to the bus to record the current scene, which will be played back (instead of turning the lamp on) when the person that was left sleeping makes a movement (i.e., when switching to "occupied" due to the detection of motion but without a previous trigger from the door sensors).

Zennio

Passion for KNX innovation

Join and send us your inquiries
about Zennio devices:

http://zennioenglish.zendesk.com

KNX   CE   RoHS