PRODUCT MANUAL

QUAD

Analog/Digital Sensor

ZN1IO-4IAD

Program version: 5.0

Manual edition: a

INDEX

Document updates 3
1. Introduction 4
1.1. QUAD 4
1.2. Installation 5
2. Configuration 6
2.1. General 6
2.2. Security binary inputs 7
2.3. Thermostats 9
3. ETS Parameterization 10
3.1. Default configuration 10
3.2. General window 11
3.2.1. Binary entry 11
a) Binary entry: Push button 11
b) Binary entry: Switch/Sensor 15
3.2.2. Temperature probe 18
3.2.3. Movement detector 20
3.3. Thermostats 21
Annex I. Communication objects 22

DOCUMENT UPDATES

Version	Modifications	Page(s)
5.0a	Changes regarding the version 4.0 of the application program: • Added Security functionality against Breakdown/Sabotage in the switch/sensor binary inputs	-
	New section *2.2. Security binary inputs*	7
	Renaming of the parameter and object "**Block**" by "**Lock**"	-

1. INTRODUCTION

1.1. QUAD

QUAD is a **Zennio** analog/digital sensor with 4 inputs, which can be individually configured as:

- **Binary entry**
- **Temperature probe**
- **Movement detector**

The inputs configured as binary entries could be connected to a push button or a switch/sensor, potential-free.

Moreover, for the binary inputs connected to a switch/sensor, it will be possible to enable a **security** function for them, in order to detect possible breakdown and/or sabotage situations in the installation. See section 2.2 for further information.

To the inputs configured as temperature probe one must connect a temperature probe (like the Zennio model **ZN1AC-NTC68**) to measure the room temperature.

The inputs configured as movement detector must be connected to the Zennio movement sensor (**ZN1IO-DETEC**).

Moreover, QUAD has the possibility to enable and configure **up to 4 different thermostats**.

Figure 1.1. A/D Sensor QUAD

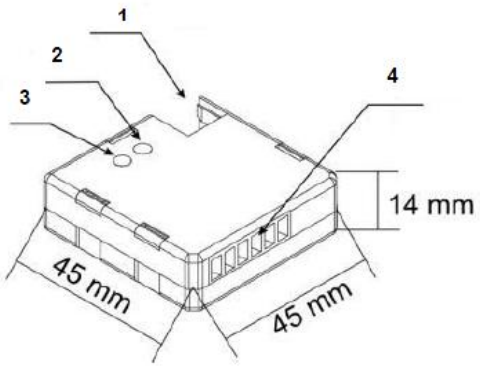
1.2. INSTALLATION

QUAD sensor connects to the KNX bus via the bus connecting terminals.

The inputs are connected to QUAD through the terminal block with screws included with the device.

Once the sensor is provided with power supply from the KNX bus, both the physical address and the associated application program can be downloaded.

In the figure 1.2, the connection scheme of QUAD is shown:



1.- KNX connector

2.- Programming LED

3.- Programming button

4.- Inputs connection

Figure 1.2. QUAD connections scheme

It is described below the functionality of its main elements:

- **Programming button:** a short press on this button set the sensor in programming mode, and the associated LED (2) lights red. If this button is held while plugging the device into de KNX bus, QUAD goes into secure mode. The LED blinks red.

- **Inputs connection:** the terminal block with screws must be plugged here (see figure 1.3) to allow connecting the different QUAD inputs (from 1 to 4). For a proper functioning, the terminals of the elements to be connected to the QUAD inputs (push button, switch, temperature probe or movement sensor) must be plugged to the corresponding input (connection points 1 to 4) and to any of the two common connection points, identified with the letter "C" in the terminal block, which are internally connected, so either of them can be used.

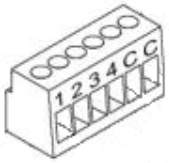


Figure 1.3. Inputs connection terminal block

To obtain a more detailed information about the technical features of QUAD, as well as security and installation information, please read the sensor **Datasheet**, included in the original package of the device and also available at**:** http://www.zennio.com.

2. CONFIGURATION

2.1. GENERAL

The analog/digital sensor QUAD is a multifunctional device that allows implementing every of its 4 inputs with several configurations:

- **Binary entry**
- **Temperature Sensor**
- **Motion detector**

Depending on the selected configuration, it is necessary to connect different external elements to the QUAD inputs: push buttons or switches, temperature probes or movement sensors (Zennio model ZN1IO-DETECT).

The inputs configured as binary entries may be push buttons or switch/sensors (depending on the connected element) and for every type, there are different customizable fields, which will be explained in section 3 of this manual.

In the inputs configured as temperature probe, several values related to the measurement and temperature sending can be established.

In the inputs configured as movement detector, up to 3 different detection channels may be enabled, to be configured for acting to a movement detection or no detection, as well as for establishing different timings. As result of the movement detection or not detection, the channel will transmit the corresponding information through the KNX bus, only when the channel is enabled (not locked). The movement detector has also a luminosity sensor that allows configuring the sending of the movement detection or not detection, constrained by the detected luminosity level, according to a previously parameterized levels.

2.2. SECURITY BINARY INPUTS

The binary inputs connected to a switch/sensor offer the possibility to enable a **security** function for them, in order to detect possible breakdown and/or sabotage situations in the installation. This security function is based on the insertion of a **resistor at the end of the line** connected.

Important Note: *The resistor values that can be installed are shown in Table 2.1.*

Ohmic value (Ω)	Maximum Tolerance (%)	Recommended Power (W)
2200 Ω	±10%	¼ W
2700 Ω	±10%	¼ W
3300 Ω	±10%	¼ W
4700 Ω	±10%	¼ W
10000 Ω	±10%	¼ W

Table 2.1. Allowed values for the end of line resistor

Two cases of use can be distinguished.

- **Installations with a switch/sensor normally closed:** The chosen resistor must be connected **in serial** in the installation line and as close as possible to the switch/sensor, at the terminals (if possible) and whose access is not easy from the outside. See figure 2.1.
  QUAD offers the possibility to detect and inform about the possible exception situations that may be produced, accidentally or intentionally, in this type of installations: **shortcircuit** or **open circuit** in the line.
  If a shortcircuit happens in the line, QUAD will activate the **Alarm object** of the corresponding input and will send the activation value periodically until this event ends. In case an open circuit happens in the line, QUAD will interpret this as an edge and will send to the KNX bus the corresponding value, according to the parameterization.

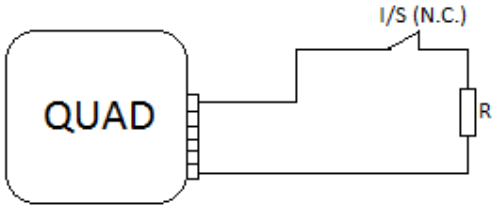


Figure 2.1. Normally closed switch/sensor. Serial resistor

- **Installations with a switch/sensor normally opened:** The chosen resistor must be connected **in parallel** in the installation line and as close as possible to the switch/sensor, at the terminals (if possible) and whose access is not easy from the outside. See figure 2.2.
  QUAD offers the possibility to detect and inform about the possible exception situations that may be produced, accidentally or intentionally, in this type of installations: **open circuit** or **shortcircuit** in the line.
  If an open circuit happens in the line, QUAD will activate the **Alarm object** of the corresponding input and will send the activation value periodically until this event ends. In case a shortcircuit happens in the line, QUAD will interpret this as an edge and will send to the KNX bus the corresponding value, according to the parameterization.

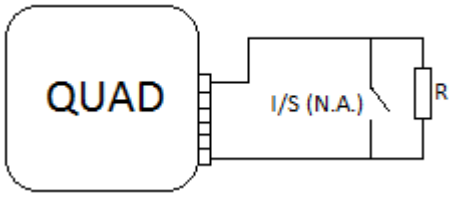


Figure 2.2. Normally opened switch/sensor. Parallel resistor

Thanks to this security function, QUAD is also able to analyze the status of the voltage levels of the installation. If QUAD detects unstable levels, for instance, due to links of other lines, it will activate the **Alarm object** of the corresponding input and will send through it the activation value periodically, until this event ends.

QUAD allows connecting several switches/sensors to a same input, only if they are of the same type, i.e., normally opened or normally closed.

If one wants to carry out this type of connection to a binary input of QUAD configured as switch/sensor with security, it is important to take into account that one **only need to connect one resistor at the end of the line** (it will be only connected to one of the switches/sensors installed).

2.3. THERMOSTATS

QUAD allows enabling and configuring in an **independent way** the functionality of up to 4 thermostats. I.e., the number of enabled thermostats does not depend on the number and type of the configured inputs.

The functioning and ETS configuration of the Zennio thermostat are described in the specific documentation "Zennio Building Thermostat", available at: http://www.zennio.com.

All these configuration options are explained in detail in the section 3. ETS Parameterization.

3. ETS PARAMETERIZATION

For starting to parameterize the QUAD sensor it is necessary, once the ETS program has been opened, importing the data base of the product (version 5.0 of the application program).

Next, the device is added to the project where desired. Click the right mouse button on the device and select "Edit parameters" for starting with the configuration.

In the following sections there is a detailed explanation about each of the different functionalities of QUAD in ETS.

3.1. DEFAULT CONFIGURATION

When entering for the first time to the parameters edition of QUAD, the following window will be shown:

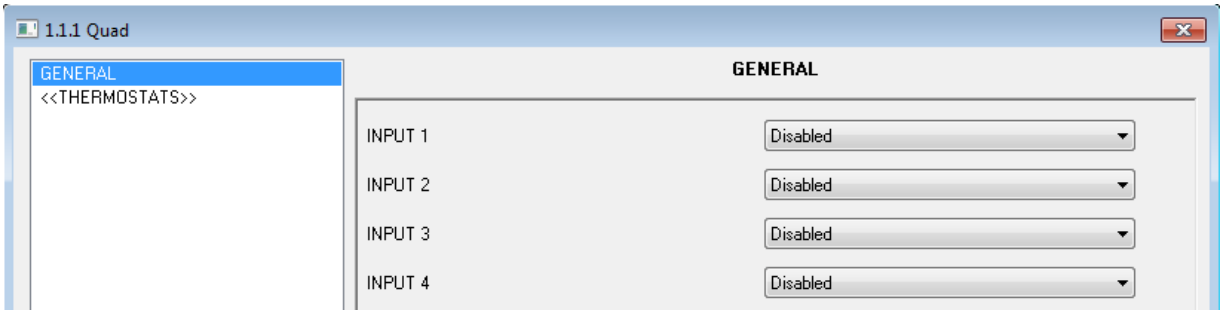


Figure 3.1. Parameterization screen by default

As it can be seen in the figure 3.1, the four inputs of the sensor are disabled by default. It will be necessary to enable and configure them one by one. The thermostat configuration window also appears. The 4 available thermostats appear disables by default, so it will bee necessary to enable and configure one to one all the thermostats required.

There are not communication objects enabled by default. These will appear when configuring the different available options.

3.2. GENERAL WINDOW

In this window every of the 4 QUAD inputs can be enabled and configured.

Depending on the type of input selected, different parameterization windows will be enabled.

3.2.1. BINARY ENTRY

When this type of entry is selected, it will be possible to connect push buttons or switches/sensors to QUAD. Depending on the kind of binary entry selected, different configuration options will be shown.

a) Binary entry: Push button

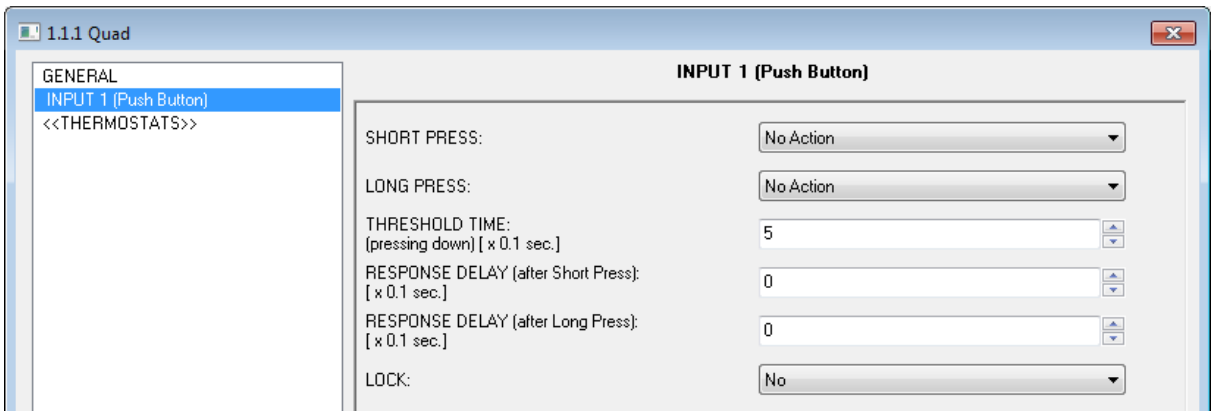


Figure 3.2. Binary entry: Push button

From the default window (figure 3.2) it will be possible to customize the working of the push button connected to QUAD, by configuring the following options:

- **Short press:** it allows selecting the option that will be carried out when pressing shortly the push button connected to the QUAD input, choosing among these:

  - **No action.** Nothing (no action is performed).

  - **Sending of 0/1.** It appears a new window to select (in the "Response" field) which value is sent to the KNX bus after a short press:

    - **"0":** the 1-bit communication object "[Ix][Short Press] 0" is enabled to send the value "0" to the bus.

- **"1":** the 1-bit communication object "[Ix][Short Press] 1" is enabled to send the value "1" to the bus.

- **"Switching 0/1":** the 1-bit communication object "[Ix][Short Press] Switching" is enabled to carry out the switched sending of the values "0" and "1" to the bus.

The transmission of the selected value can be carried out in a cyclical way, i.e., it is possible to parameterize a periodical sending to the KNX bus, in the field "Cyclical response sending".

Figure 3.3. Sending of 0/1

➢ **Shutter control.** This function allows sending to the KNX bus a 1-bit object to control the shutters. In the field "Response" one can choose the action to carry out with a short press:

- **Up:** the 1-bit communication object "[Ix][Short Press] Move Up Shutter" is enabled and it will send to the KNX bus the move up order (sending of value "0").

- **Down:** the 1-bit communication object "[Ix][Short Press] Move Down Shutter" is enabled and it will send to the KNX bus the move down order (sending of value "1").

- **Up/Down (switched):** the 1-bit communication object "[Ix][Short Press] Move Up/Down Shutter" is enabled and it will send to the KNX bus an alternate switching of the orders move up and move down shutter (sending the values "0" and "1", respectively). This option allows the handling of the shutter through just one input.

- **Stop/Step Up:** the 1-bit communication object "[Ix][Short Press] Stop/Step Up Shutter" is enabled and it will send to the KNX bus the stop shutter order. In case the shutters have lamellas, this object allows moving them one step up. In both cases, the value sent to the KNX bus is "0".

- **Stop/Step Down:** the 1-bit communication object "[Ix][Short Press] Stop/Step Down Shutter" is enabled and it will send to the KNX bus the stop shutter order. In case the shutters have lamellas, this object allows moving them one step down. In both cases, the value sent to the KNX bus is "1".

- **Stop/Switched Step:** the 1-bit communication object "[Ix][Short Press] Stop/Step Shutter (switched)" is enabled and it will send to the KNX bus the stop shutter order. In case the shutters have lamellas, this object allows switching their movement, with steps up and down. The value "0" or "1" will be sent to the KNX bus.

➢ **Dimmer control.** This function allows sending to the KNX bus communication objects to control a light Dimmer device. In the field "Response" one can choose the action to carry out with a short press:

- **Light ON:** the 1-bit communication object "[Ix][Short Press] Dimmer ON" is enabled and it will send to the KNX bus the Dimmer ON order (sending of value "1").

- **Light OFF:** the 1-bit communication object "[Ix][Short Press] Dimmer OFF" is enabled and it will send to the KNX bus the Dimmer OFF order (sending of value "0").

- **Light ON/OFF (switched):** the 1-bit communication object "[Ix][Short Press] Dimmer ON/OFF" is enabled and it will send to the KNX bus an alternate switching of the orders Dimmer ON and OFF (sending the values "1" and "0", respectively).

- **Brighter:** the 4-bit communication object "[Ix][Short Press] Brighter" is enabled to allow increasing the brightness level of the Dimmer with every short press, taking into account the parameterized dimming step (see table 3.1). A first short press starts increasing the brightness level, step by step. A second short press stops this increase.

- **Darker:** the 4-bit communication object "[Ix][Short Press] Darker" is enabled to allow decreasing the brightness level of the Dimmer with every short press, taking into account the parameterized dimming step (see table 3.1). A first short press starts decreasing the brightness level, step by step. A second short press stops this decrease.

- **Brighter/Darker (switched):** the 4-bit communication object "[Ix][Short Press] Brighter/Darker" is enabled and it will send to the KNX bus an alternate switching of the orders brighter/darker, according to the parameterized dimming step (see table

3.1). A first short press starts increasing (or decreasing) the brightness level. A second short press, stops the dimming. The next short press will carry out the opposite action (brighter/darker) to the first press. Another press, will stop the dimming and so on.

Note: *If any of the three last dimming options is selected, it will necessary to define the dimming step to apply, i. e., the increase or decrease of the brightness given by the Dimmer in every step. See table 3.1 to know the available steps.*

Dimming step	Necessary pulsations for a complete regulation (0-100%)
(1) 100%	1
(2) 50%	2
(3) 25%	4
(4) 12.5%	8
(5) 6.25%	16
(6) 3.1%	32
(7) 1.5%	64

Table 3.1. Dimming step

➢ **Sending of a Scene.** This function allows sending to the KNX bus a 1-byte communication object to control the scenes. In the field "Response" one can choose the action to carry out with a short press:

- **Run Scene:** the 1-byte object "[Ix][Short Press] Run Scene" is enabled to send to the bus a value (between 1 and 64) that will execute the corresponding scene.
- **Save Scene**: the 1-byte object "[Ix][Short Press] Save Scene" is enabled. It allows saving the current scene, to be run after, when required.

Long press: it allows selecting the option that will be carried out when pressing for a long time the push button connected to the QUAD input. The configuration options are the same as in **Short Press**.

Threshold time: this parameter defines the minimum time (in tenths of second) that the push button must be pressed to turn a short press into a long press.

Response delay (after short press): this parameter sets the time (in tenths of second) to wait for the object associated to the short press to be sent to the KNX bus. I.e., with a short

press, QUAD will wait this parameterized delay to send to the bus the value of the corresponding object. To get an immediate sending (no delay), set the value 0 in this field.

- **Response delay (after long press):** this parameter sets the time (in tenths of second) to wait for the object associated to the long press to be sent to the KNX bus. I.e., with a long press, QUAD will wait this parameterized delay to send to the bus the value of the corresponding object. To get an immediate sending (no delay), set the value 0 in this field.

- **Lock:** the 1-bit object "[Ix] Lock" is enabled. It allows locking any action performed over the QUAD input, i.e., disabling its control. The functioning of this parameter is as follows: when receiving the value "1" through the lock object, QUAD locks the input,, ignoring any pressing or action over it. When receiving the value "0", the input is enabled again.
The actions or pressings performed during the lock status will not be taken into account when the input is unlocked.

b) Binary entry: Switch/Sensor

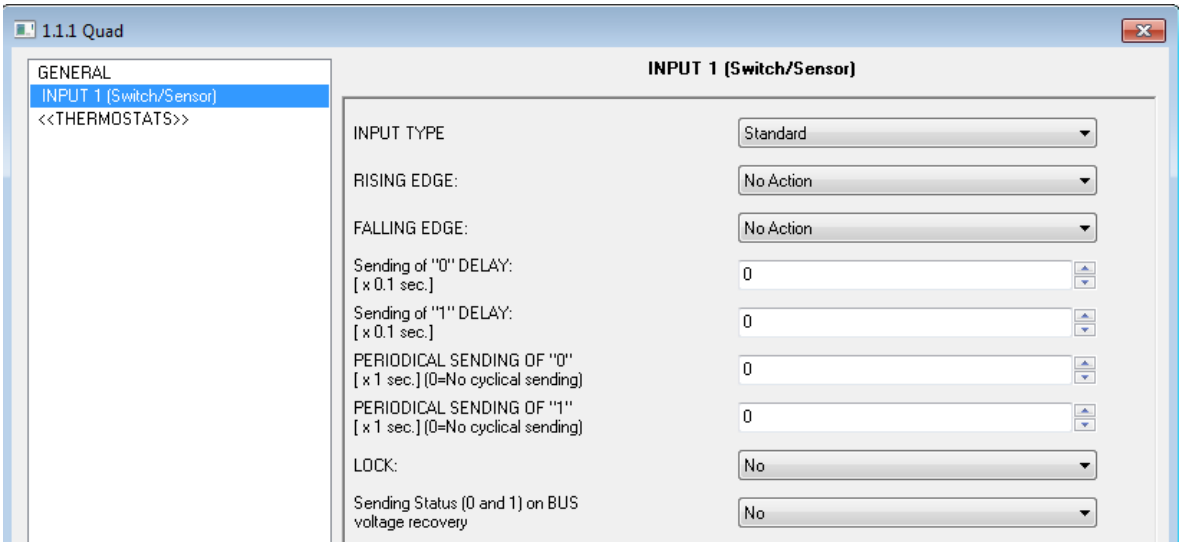


Figure 3.4. Binary entry: Switch/Sensor

From the default window (figure 3.4) it will be possible to customize the working of the switch or sensor connected to the QUAD input. The first parameter to configure is the Input type: **Standard** or with **Security** (with end of line resistor).

In case of selecting the switch/sensor input with security, the 1-bit communication object "[Ix] Alarm: Breakdown, Sabotage, Unstable line" is enabled, through which the value "1" will be sent every 30 seconds in case QUAD detects one of these situations: breakdown, sabotage or unstable line in any

of its inputs (when the event ends, QUAD will send the value "0" through this object). Moreover, the following related parameters are shown:

- **Switch/sensor type:** where to select the usual working mode of the switch/sensor connected to the QUAD input:
  - **N.O. (Parallel resistor):** Normally opened switch/sensor.
  - **N.C. (Serial resistor):** Normally closed switch/sensor.
- **Resistor value:** to establish the ohmic value of the end of line resistor connected to the switch/sensor. The available options are: 2.2 KΩ, 2.7 KΩ, 3.3 KΩ, 4.7 KΩ and 10 KΩ.

To get further information about the working of this type of inputs, please consult section 2.2.

Both input types (standard and with security) share the following parameters:

- **Rising edge:** set the option to be carried out with a rising edge in the QUAD input, choosing among:
  - **No action:** no action is performed.
  - **0:** QUAD will send the value "0" to the KNX bus through the 1-bit object "[Ix][Sensor] Edge".
  - **1:** QUAD will send the value "1" to the KNX bus through the 1-bit object "[Ix][Sensor] Edge".
  - **Switching 0/1:** QUAD sends a switching of the values "0" and "1" to the KNX bus with a rising edge in the input, through the object "[Ix][Sensor] Edge"
- **Falling edge:** set the option to be carried out with a falling edge in the QUAD input. The available options are the same as for **Rising edge.**
- **Sending of "0" delay:** sets the time (in tenths of second) that QUAD waits, once received the corresponding order through the switch/sensor connected to its input, to send the value "0" through the object "[Ix][Sensor] Edge". In case the value “0” is sent with a rising edge, this delay will indicate the time the switch/sensor must be kept pressed for that value to be sent to the bus.

- **Sending of "1" delay:** sets the time (in tenths of second) that QUAD waits, once received the corresponding order through the switch/sensor connected to its input, to send the value "1" through the object "[Ix][Sensor] Edge". In case the value “1” is sent with a rising edge, this delay will indicate the time the switch/sensor must be kept pressed for that value to be sent to the bus.

- **Periodical sending of "0":** sets the sending period (in seconds) of the value "0" through the object "[Ix][Sensor] Edge". QUAD will send this value to the KNX bus in an undefined and cyclical way, according to the parameterized time. For a non-cyclical sending, set a 0 in this field.

- **Periodical sending of "1"**: sets the sending period (in seconds) of the value "1" through the object "[Ix][Sensor] Edge". QUAD will send this value to the KNX bus in an undefined and cyclical way, according to the parameterized time. For a non-cyclical sending, set a 0 in this field.

- **Lock:** the 1-bit object "[Ix] Lock" is enabled. It allows locking any action performed over the QUAD input, i.e., disabling its control. The functioning of this parameter is as follows: when receiving the value "1" through the lock object, QUAD locks the input and stops monitoring it for checking possible edge changes and possible alarm situations (for inputs with security). When receiving the value "0", the input is enabled again and the edge and alarm objects will be sent to the bus, only if their status has changed regarding the one they had before locking the input.

- **Sending Status on Bus voltage recovery:** if this function is enabled, the status of the QUAD input will be automatically sent to the bus (values "0" and "1", whichever is applicable) when recovering the voltage on the KNX bus, after the parameterized delay (in seconds). For the case of inputs with security, the Alarm status will be also sent.

Note: *If the alarm was activated, when recovering the bus power, only the alarm status will be sent to the bus.*

3.2.2. TEMPERATURE PROBE

When selecting this type of input, it will be possible to configure a set of parameters, related to the temperature sensor.

When configuring an input as temperature probe, in the topology window the communications objects "[Ix] Current Temperature" (2 bytes) and "[Ix] Probe Error" (1 bit) are shown. Through the first one, the temperature value measured by the probe connected to the QUAD input can be known. Through the second object, it is possible to know whether an error due to a wrong connection of the probe has occurred (value "1" in the object); when this error is corrected, the error object will take the value "0".

The default configuration window of the temperature sensor is shown in the figure 3.5:

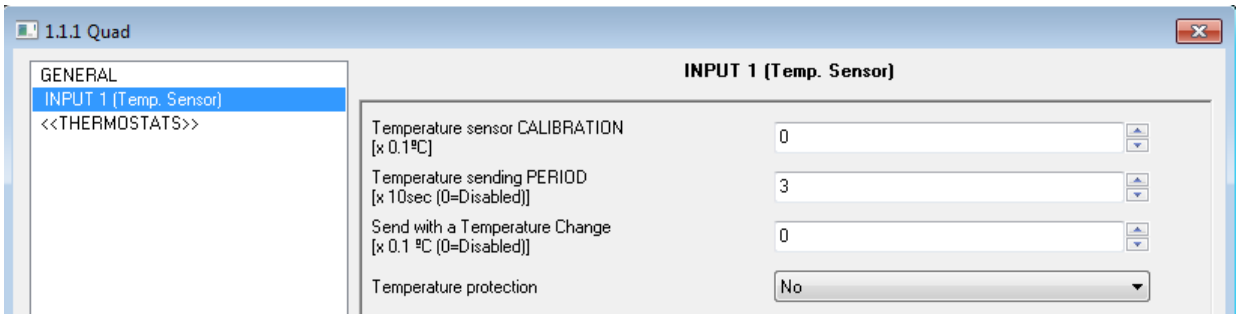


Figure 3.5. Temperature Sensor

Where to configure the following parameters:

- **Temperature sensor calibration:** this option allows calibrating (the setting the tenths of degree) a possible deviation between the measurement made by the sensor and the real temperature of the room.

- **Temperature sending period:** to select by parameter the time (in tens of second) to send cyclically to the KNX bus the current temperature measurement, through the communication object "[Ix] Current Temperature". The value 0 indicates that the periodical sending is disabled.

- **Send with a temperature change:** QUAD will send to the KNX bus the current temperature when it has changed (increased or decreased) with regard to the last measure, the amount of degrees specified in this parameter (from 0 to 200 tenths of degree). To disable this sending, please set the value 0 in this field.

- **Temperature protection:** to select an overheating, overcooling or both protection. Depending on the chosen protection, one or two communication objects will be enabled: "[Ix] Overheating" and "[Ix] Overcooling", which will indicate (with the value "1") if the corresponding temperature has been exceeded. It is necessary to define the overheating or the overcooling temperature (or both), in degrees as well as an hysteresis value (tenths of degree).

3.2.3. MOVEMENT DETECTOR

The analog/digital sensor QUAD allows configuring its inputs as movement detector, with up to 3 detection channels each.

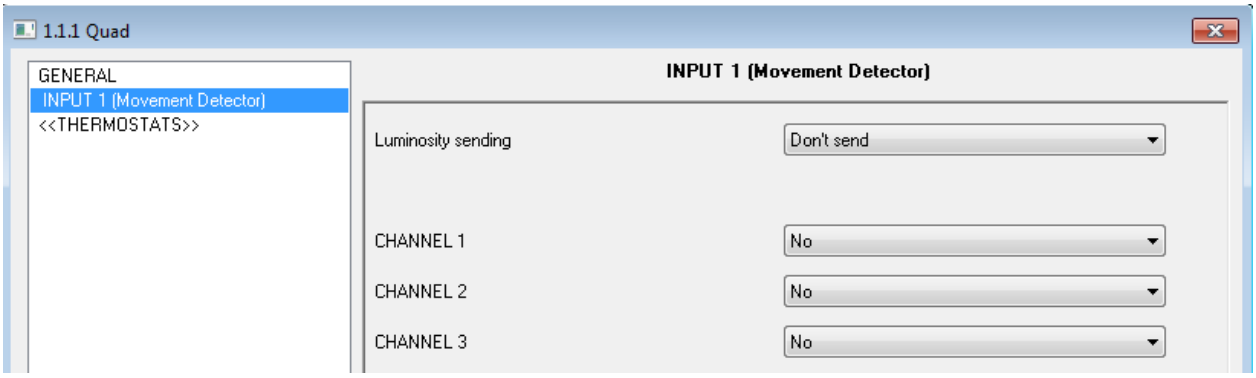


Figure 3.6. Movement detector configuration

After enabling the available channels, the next configuration window will be shown:

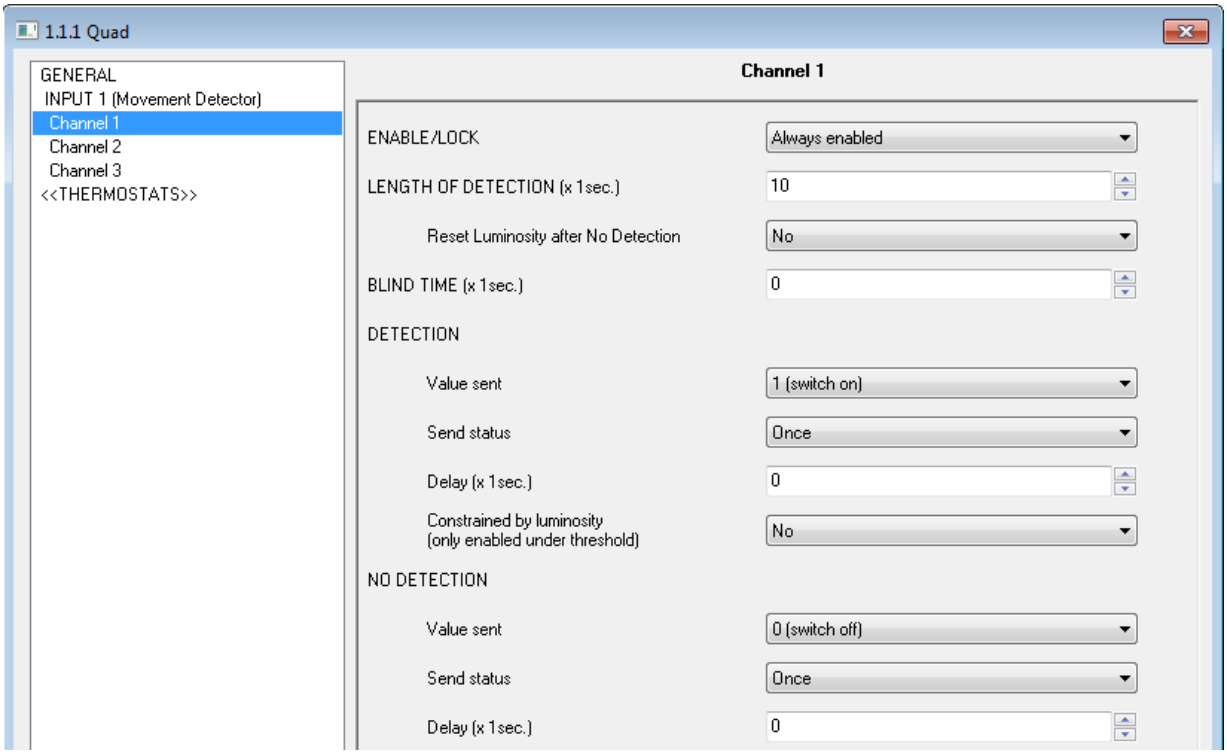


Figure 3.7. Channel configuration

For further information on the functioning and ETS parameterization of them, please consult the specific documentation "**Motion detector**", available at: http://www.zennio.com.

3.3. THERMOSTATS

As mentioned before, QUAD has the possibility to enable and configure up to 4 thermostats, in an independent way.

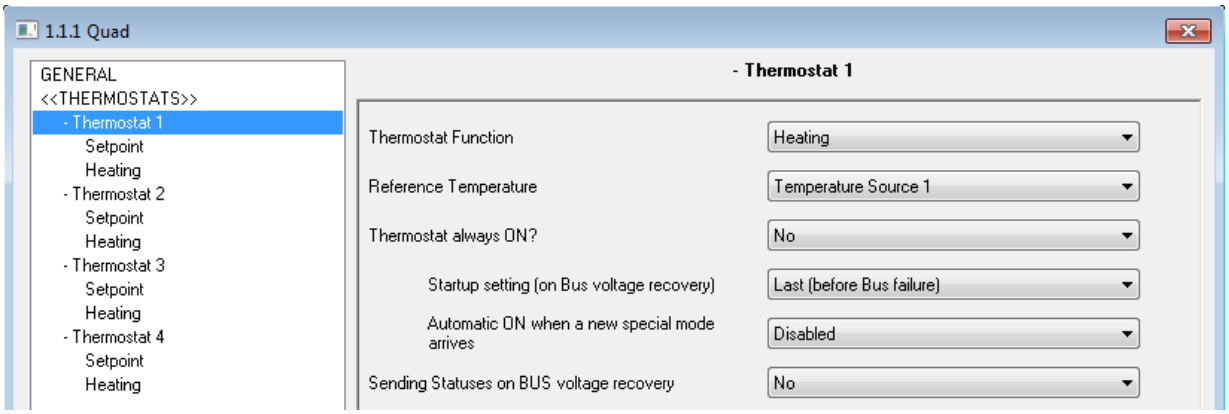


Figure 3.8. Thermostat 1 Configuration

To obtain theoretical information about the Zennio thermostat functioning, as well as information about its ETS configuration, please consult the specific documentation "**Zennio Building Thermostat** " available at http://www.zennio.com.

ANNEX I. COMMUNICATION OBJECTS

SECTION	NUMBER	SIZE	IN/OUT	FLAGS	VALUES			NAME	DESCRIPTION
					RANGE	1st TIME	RESET		
BINARY ENTRIES	0-3	1 bit	I	W	0/1	0	Last	[Ix] Lock	1=Input disabled; 0=Input free
	4-7	1 bit	I	RWT	0/1	0	Last	[Ix] [Long Press] "0"	Short Pr. -> Sending of "0"
								[Ix] [Long Press] "1"	Short Pr. -> Sending of "1"
								[Ix] [Long Press] Switching	Short Pr. -> Switching "0/1"
								[Ix] [Long Press] Move Up Shutter	Short Pr. -> Sending of 0 (Up)
								[Ix] [Long Press] Move Down Shutter	Short Pr. -> Sending of 1 (Down)
								[Ix] [Long Press] Move Up/Down Shutter	Short Pr. -> Switching "0/1"
								[Ix] [Long Press] Stop/Step Up Shutter	Short Pr. -> Sending of "0"
								[Ix] [Long Press] Stop/Step Down Shutter	Short Pr. -> Sending of "1"
								[Ix] [Long Press] Stop/Step Shutter (switched)	Short Pr. -> Switching "0/1"
								[Ix] [Long Press] Dimmer ON	Short Pr. -> Sending of 1 (ON)
								[Ix] [Long Press] Dimmer OFF	Short Pr. -> Sending of 0 (OFF)
								[Ix] [Long Press] Dimmer ON/OFF	Short Pr. -> Switching "0/1"
								[Ix] [Sensor] Edge	Edge -> Sending of "0" or "1"
	8-11	4 bits	O	RT	0-15	0	Last	[Ix] [Long Press] Brighter	Sh.Pr.->Bright; Sh.Pr->Stop
								[Ix] [Long Press] Darker	Sh.Pr.->Dark; Sh.Pr->Stop
								[Ix] [Long Press] Brighter/Darker	Sh.Pr.->Bright/Dark; Sh.Pr->Stop
	12-15	1 byte	O	RT	0-63; 128-191	Indifferent	Indifferent	[Ix] [Long Press] Run Scene	Short Pr. -> Sending of 0-63

SECTION	NUMBER	SIZE	IN/OUT	FLAGS	VALUES			NAME	DESCRIPTION
					RANGE	1st TIME	RESET		
BINARY ENTRIES	12-15	1 byte	O	RT	0-63; 128-191	Indifferent	Indifferent	[Ix] [Long Press] Save Scene	Short Pr. -> Send. of 128-191
	16-19	1 bit	O	RWT	0/1	0	Last	[Ix] [Long Press] "0"	Long Pr. -> Sending of "0"
								[Ix] [Long Press] "1"	Long Pr. -> Sending of "1"
								[Ix] [Long Press] Switching	Long Pr. -> Switching "0/1"
								[Ix] [Long Press] Move Up Shutter	Long Pr. -> Sending of 0 (Up)
								[Ix] [Long Press] Move Down Shutter	Long Pr. -> Sending of 1 (Down)
								[Ix] [Long Press] Move Up/Down Shutter	Long Pr. -> Switching "0/1"
								[Ix] [Long Press] Stop/Step Up Shutter	Long Pr. -> Sending of "0"
								[Ix] [Long Press] Stop/Step Up Shutter	Long Pr. -> Sending of "1"
								[Ix] [Long Press] Stop/Step Shutter (switched)	Long Pr. -> Switching "0/1"
								[Ix] [Long Press] Dimmer ON	Long Pr. -> Sending of 1 (ON)
								[Ix] [Long Press] Dimmer OFF	Long Pr. -> Sending of 0 (OFF)
								[Ix] [Long Press] Dimmer ON/OFF	Long Pr. -> Switching "0/1"
	20-23	4 bits	O	RT	0-15	0	Last	[Ix] [Long Press] Brighter	Lg.Pr.->Bright; Lg.Pr->Stop
								[Ix] [Long Press] Darker	Lg.Pr.-Dark; Lg.Pr->Stop
								[Ix] [Long Press] Brighter/Darker	Lg.Pr.->Bright/Dark; Lg.Pr->Stop
	24-27	1 byte	O	RT	0-63; 128-191	Indifferent	Indifferent	[Ix] [Long Press] Run Scene	Long Pr. -> Sending of 0-63
								[Ix] [Long Press] Save Scene	Long Pr. -> Send. of 128-191

SECTION	NUMBER	SIZE	IN/OUT	FLAGS	VALUES			NAME	DESCRIPTION
					RANGE	1st TIME	RESET		
MOVEMENT DETECTOR	28-31	1 bit	O	T	0/1	0	0	[Ix] Short Circuit	1=Shortcircuit; 0=No Short circuit
	32-35	1 bit	O	T	0/1	0	0	[Ix] Open Circuit	1=Open Circuit; 0=No Open Circ.
	36-39	1 byte	O	RT	0-100%	Indifferent	Last	[Ix] Luminosity level	Luminosity of Input 6
	40-51	1 bit	I	W	0/1	1	1	[Ix][Ch.y] Channel enabling	1=Enable; 0=Disable
						0	0	[Ix][Ch.y] Channel lock	1=Lock; 0=Unlock
	52-63	1 bit	O	T	0/1	Parameter.	Parameter.	[Ix][Ch.y] Detection status	According to parameters
	64-75	1 byte	I	W	0-63	Indifferent	Indifferent	[Ix][Ch.y] Scene reception	0-63 (Run Scene 1-64)
	76-87	1 byte	O	T	0-63	Indifferent	Indifferent	[Ix][Ch.y] Scene sending	0-63 (Send scene 1-64)
TEMPERATURE PROBE	88-91	2 bytes	O	RT	-40ºC – 150ºC	Indifferent	Indifferent	[Ix] Current temperature	Temperature sensor value
	92-95	1 bit	O	RT	0/1	0	Last	[Ix] Overcooling	1=Overcooling; 0=No Overcooling
	96-99	1 bit	O	RT	0/1	0	Last	[Ix] Overheating	1=Overheating; 0=No Overheating
	100-103	1 bit	O	RT	0/1	Connection dependant	Connection dependant	[Ix] Probe Error	1=Error; 0=No Error
THERMOSTATS	104-111 (even)	2 bytes	I	W	-40ºC – 150ºC	25ºC	Last	[Tx] Temperature Source 1	External sensor measure
	104-111 (odd)	2 bytes	I	W	-40ºC – 150ºC	25ºC	Last	[Tx] Temperature Source 2	External sensor measure
	112-115	1 byte	I	W	1-4	Parameter.	Parameter.	[Tx] Special Mode:	1-byte HVAC mode
	116-131	1 bit	I	W	0/1	0	Last	[Tx] Special Mode: x (x= Comfort, Standby, Economy or Protection)	0=Nothing; 1=Trigger

SECTION	NUMBER	SIZE	IN/OUT	FLAGS	VALUES			NAME	DESCRIPTION
					RANGE	1st TIME	RESET		
THERMOSTATS	116-131	1 bit	I	W	0/1	0	Last	[Tx] Special Mode: x (x= Comfort, Standby, Economy or Protection)	0=Off; 1=On
	132-135	1 bit	I	W	0/1	0	Last	[Tx] Window status (input)	0=Closed; 1=Opened
	136-139	1 bit	I	W	0/1	0	0	[Tx] Comfort prolongation	0=Nothing; 1=Timed Comfort
	140-143	1 byte	O	RT	0-255	Parameter.	Last	"[Tx] Special Mode (Status)	1-byte HVAC mode
	144-147	2 bytes	I	W	-20ºC – 150ºC	Parameter.	Last	[Tx] Basic setpoint	Reference setpoint:
		2 bytes	I	W	-20ºC – 150ºC	Parameter.	Last	[Tx] Setpoint	Thermostat Setpoint input
	148-151	1 bit	I	W	0/1	0	Indifferent	[Tx] Setpoint (Step)	0=-0.5ºC;1=+0.5ºC
	152-155	2 bytes	I	W	-10ºC, 10ºC	0	Last	[Tx] Setpoint (Offset)	Float value
	156-159	2 bytes	O	RT	-20ºC – 150ºC	25ºC	Last	[Tx] Setpoint (Status)	Current setpoint
	160-163	2 bytes	O	RT	-20ºC – 150ºC	Parameter.	Last	[Tx] Basic Setpoint (Status)	Current basic setpoint
	164-167	2 bytes	O	RT	-10ºC, 10ºC	0	Last	[Tx] Setpoint (Offset Status)	Current setpoint offset
	168-171	1 bit	I	W	0/1	0	Indifferent	[Ix] Setpoint reset	Reset setpoint to default
		1 bit	I	W	0/1	0	Indifferent	[Tx] Offset Reset	Reset offset
	172-175	1 bit	I	W	0/1	Parameter.	Last	[Tx] Mode	0=Cooling; 1=Heating
	176-179	1 bit	O	RT	0/1	Parameter.	Last	[Tx] Mode (Status)	0=Cooling; 1=Heating
	180-183	1 bit	I	W	0/1	Parameter.	Parameter.	[Tx] ON/OFF	0=Off; 1=On
	184-187	1 bit	O	RT	0/1	Parameter.	Parameter.	[Tx] ON/OFF (Status)	0=Off; 1=On

SECTION	NUMBER	SIZE	IN/OUT	FLAGS	VALUES			NAME	DESCRIPTION
					RANGE	1st TIME	RESET		
THERMOSTATS	188, 190, 192, 194	1 bit	O	RT	0/1	0	Last	[Tx] Control variable (Cool)	PI control (PWM)
								[Tx] Control variable (Cool)	2-point control
	189, 191, 193, 195	1 bit	O	RT	0/1	0	Last	[Tx] Control variable (Heat)	PI control (PWM)
								[Tx] Control variable (Heat)	2-point control
	196-203 (even)	1 byte	O	RT	0-100%	0	Last	[Tx] Control variable (Cool)	PI control (Continuous)
	196-203 (odd)	1 byte	O	RT	0-100%	0	Last	[Tx] Control variable (Heat)	PI control (Continuous)
	204, 206, 208, 210	1 bit	O	RT	0/1	0	Last	[Tx] Additional Cool	Temp >= (Setpoint+Band) => "1"
	205, 207, 209, 211	1 bit	O	RT	0/1	0	Last	[Tx] Additional Heat	Temp <= (Setpoint-Band) => "1"
BREAKDOWN, SABOTAGE, UNSTABLE LINE ALARM	212-215	1 bit	I/O	RWT	0/1	Detection dependant	Last	[Ix] Alarm: Breakdown, Sabotage, Unstable line	("1" -> Active; "0"->No Active)

ZENNIO TECHNICAL DOCUMENTATION

BECOME USER!

http://zennioenglish.zendesk.com

TECHNICAL SUPPORT