



Fan Coil QUATRO

ACTinBOX QUATRO Actuator

ZN1IO-AB40



Program version: 1.4

Manual edition: a

INDEX

1. Introduction	3
1.1. Fan Coil QUATRO.....	3
1.2. ACTinBOX QUATRO Actuator	3
1.3. Installation	4
1.4. Fan Coil connection	5
2. Fan Coil QUATRO configuration	6
2.1. Type of control	6
2.1.1. Control applied to the fan	6
2.1.2. Control applied to the valve.....	6
2.2. Functioning modes	12
2.2.1. Cooling mode	12
2.2.2. Heating mode.....	12
2.2.3. Both (Cooling and Heating)	12
3. ETS Parameterization.....	14
3.1. Default configuration	14
3.2. Mode	16
3.2.1. Cooling mode	16
3.2.2. Heating mode.....	16
3.2.3. Both (Cooling and Heating)	17
3.3. Type of control	17
3.3.1. Control applied to the fan	17
3.3.2. Control applied to the valve.....	20
3.4. Security switch-off.....	22
3.5. Initial configuration	22
3.6. Start-up status sending.....	23
3.7. Logical functions.....	24
Annex I. Communication objects.....	25

1. INTRODUCTION

1.1. FAN COIL QUATRO

Fan Coil QAUTRO is a **Zennio** application that uses the **ACTinBOX QUATRO** actuator to fulfil the climate control necessities in installations with an integrated Fan Coil unit within a domotic environment.

This application allows manual and automatic management of the valves and the fan speed.

Since the actuator has 4 outputs, it will be possible to control Fan Coil units of **2 pipes**.

1.2. ACTINBOX QUATRO ACTUATOR

ACTinBOX QUATRO is a KNX actuator able to control Fan Coil systems, which combines in a same device the following features:

- **4 multifunction binary outputs**, of 16A each, to be configured as:
 - Up to 2 shutter channels (with or without slats).
 - Up to 4 individual outputs.
- Multi operational **logical functions** module.
- **Manually operating** over the actuator outputs, by means of the Zennio infrared remote control.



Figure 1.1. ACTinBOX QUATRO Actuator

1.3. INSTALLATION

ACTinBOX QUATRO connects to the KNX bus through the included KNX connector.

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

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

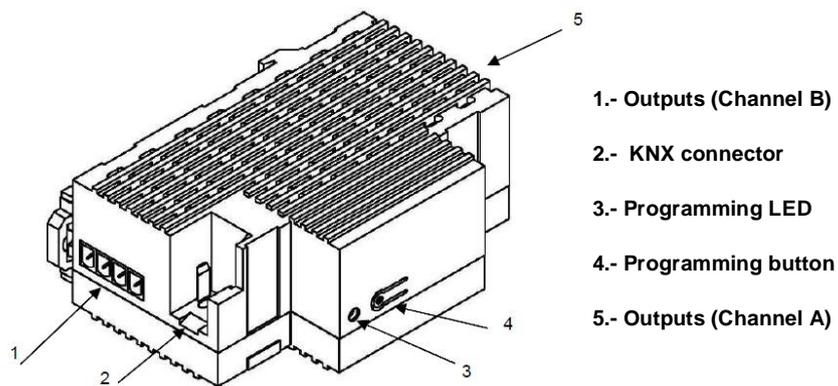


Figure 1.2. ACTinBOX QUATRO Elements scheme

It is described below the functionality of the main elements of the actuator:

 **Programming button (4):** a short press on this button set the actuator in programming mode, and the associated LED (3) lights red. (**Note:** *If this button is held while plugging the device into de KNX bus, ACTinBOX QUATRO goes into secure mode. The LED blinks red every 0.5 seconds*).

 **Outputs (1 and 5):** the terminal blocks with screws (included in the original package) must be plugged here to allow connecting the different systems to be controlled by the actuator. The terminal blocks wiring may be carried out before inserting them in the corresponding points.

1.4. FAN COIL CONNECTION

Next, a schematic of a typical connection used to control a Fan Coil in a KNX installation with the ACTinBOX QUATRO is shown:

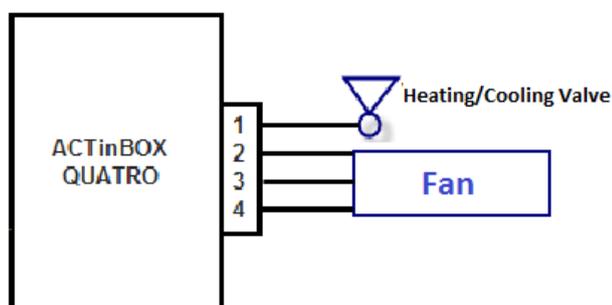


Figure 1.3. ACTinBOX QUATRO to Fan Coil connection

As shown in the figure 1.3, the correspondences of the actuator outputs to its connection with the Fan Coil are:

- QUATRO Output 1: Valve control (cooling/heating)
- QUATRO Output 2: Fan Speed 1
- QUATRO Output 3: Fan Speed 2
- QUATRO Output 4: Fan Speed 3

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

2. FAN COIL QUATRO CONFIGURATION

2.1. TYPE OF CONTROL

ACTinBOX QUATRO can control Fan Coil system in two different ways: control applied to the fan or control applied to the valve.

2.1.1. CONTROL APPLIED TO THE FAN

The fan will be in charge to control the Fan Coil functioning, through orders received from the KNX bus. The valve status will depend on the fan status. If it is on, the valve will be opened and if it is off, the valve will be closed.

Different fan percentages can be defined by parameter, associated to each of the fan speeds available. Speed 0 (Stopped), 1, 2 or 3. Depending on the received fan percentage, the fan will be set in the corresponding ventilation level (or speed), according to the percentages parametrized in ETS.

2.1.2. CONTROL APPLIED TO THE VALVE

In this case, the Fan Coil valve will be controlled via the orders received from the KNX bus through the associated communication objects. Next situations can be distinguished, depending on the parametrized mode:

-  **Cooling mode.** The valve will be controlled through the corresponding communication object ("PWM Cooling control").
-  **Heating mode.** The valve will be controlled through the corresponding communication object ("PWM Heating control").
-  **Both modes.** In the cooling mode, the valve will follow the orders received through the communication object "PWM Cooling control". In the heating mode, the valve will follow the orders received through the communication object "PWM Heating control". In both cases, changes produced in the object that does not belong to the actual mode will be ignored.

The fan status can be controlled independently (manual control) or in an automatic way by the program itself (automatic control):

Manual speed control. Ventilation level will be controlled independently, using any of the following methods:

- **Step control:** the ventilation level/speed is controlled through the 1-bit communication object "Fan Step control", by writing a "1" to increase one level or a "0" to decrease it. This control type can be non cyclical (see figure 2.1) or cyclical (figure 2.2).

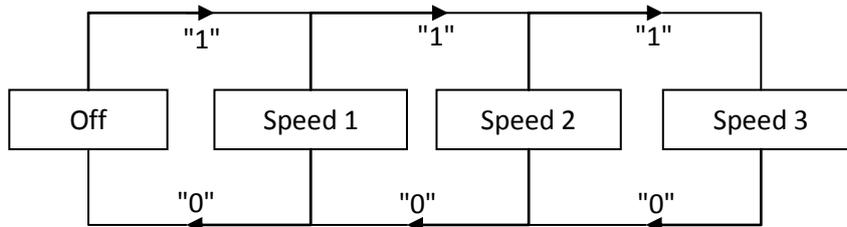


Figure 2.1. Non cyclical manual step control

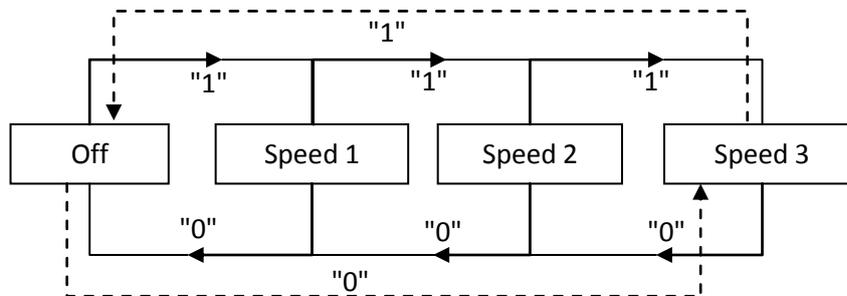


Figure 2.2. Cyclical manual step control

- **Single control:** if this method is chosen, eight 1-bit communication objects will be automatically enabled. The first four are associated to each ventilation level: "Fan: Speed x" (x = 0, 1, 2 or 3), and they only react when a "1" is sent through the object. If a "0" is received, no action is carried out (see table 2.1). The four last objects, "Fan: Speed x (Status)", are used to know the ventilation status. All of them have the value "0" except the one that makes reference to the actual ventilation level, whose value will be "1".

Object	Value	Action
Speed 0	0	Ignore
	1	Speed 0 (stopped)
Speed 1	0	Ignore
	1	Speed 1
Speed 2	0	Ignore
	1	Speed 2
Speed 3	0	Ignore
	1	Speed 3

Table 2.1. Actions of the manual control over the fan

➤ **Percentage control:** the fan speed will be controlled through the communication object "Fan Speed [1 byte]", by writing a percentage value on it. This value will correspond with a determinate ventilation level, according to the table 2.2:

Off	0%
Speed 1	1 - 25 %
Speed 2	26 - 50%
Speed 3	51 - 100%

Table 2.2. Percentages associated to each fan speed

 **Auto speed control.** When selecting this method, 4 communication objects will be provided:

- "Fan Auto": 1-bit object to activate or deactivate the automatic control by sending a "1" or "0", respectively.
- "Fan Auto (Status)": 1-bit object used to know, at any time, the automatic fan control status ("0" means deactivated and "1", activated).
- "Ambient temperature" and "Setpoint temperature": 2-byte communication objects, which will be used to set the fan speed automatically, as shown next.

When enabling the automatic control ("Fan Auto = 1"), the fan will be immediately set in any of the three speed levels available (or it will be switched off: Level/Speed 0), depending on the temperature criterion chosen (see figure 2.3).

To leave the automatic mode, just write a "0" in the object "Fan: Auto". The ventilation level of the fan will be kept until it receives a new order modifying it.

It is also possible to leave the automatic mode by manually changing the ventilation level (sending orders through the enabled object for manual control). The object "Fan Auto (Status)" will get the value "0" and the fan speed will be the corresponding one, according to the received order.

To get a correct configuration of the automatic mode, it will be necessary to define a set parameters in the corresponding parameterization window in ETS (see section 3.3.2):

➤ **t0, t1, t2:** these are the temperature differences, in tenths of degree, between the different fan speed levels (from 0 to 3). The application program will automatically estimate the difference between the ambient and the setpoint temperature ("Diff") and it will compare it with the values t0, t1 and t2. And, depending on the range this difference is, it will activate a ventilation speed or other:

- If $\text{Diff} \leq t_0$ --> Speed 0 (fan off/stopped).
- If $t_0 < \text{Diff} \leq t_0 + t_1$ --> Speed 1
- If $t_0 + t_1 < \text{Diff} \leq t_0 + t_1 + t_2$ --> Speed 2
- If $\text{Diff} > t_0 + t_1 + t_2$ --> Speed 3

✓ Example:

Suppose that the following differences between ventilation levels are parameterized:

$$t_0 = 1^\circ\text{C}; t_1 = 2^\circ\text{C}; t_2 = 2^\circ\text{C}$$

Setpoint temperature is 25°C and the ambient temperature has the following values:

1º) $T_{\text{amb}} = 25.5^\circ\text{C}$. Difference Setpoint-Ambient = Diff = 0.5°C . As this difference is lower than t_0 , Fan Coil QUATRO will switch off the fan.

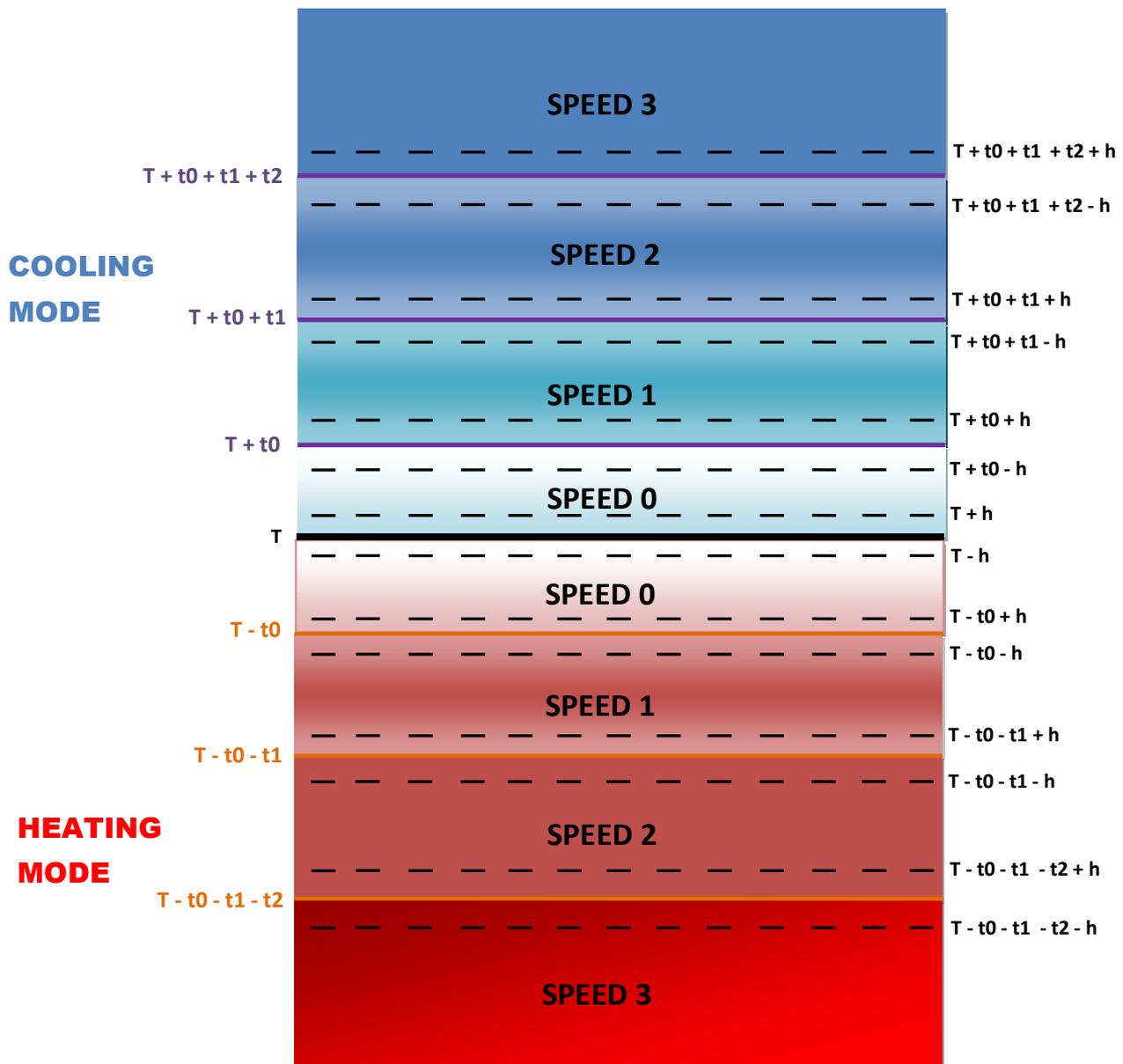
2º) $T_{\text{amb}} = 28^\circ\text{C}$. Diff = 3°C . This difference is equal to " t_0+t_1 ", so Fan Coil QUATRO will activate the fan at speed 1.

3º) $T_{\text{amb}} = 31^\circ\text{C}$. Diff = 6°C . This difference is higher than " $t_0+t_1+t_2$ ", so Fan Coil QUATRO will increase the fan speed to speed 3.

- **Hysteresis [x 0.1°C]:** parameter that allows a higher control over the temperature variations around the limit of the different fan speed levels.

Besides these parameters, it will be necessary to define the two temperature objects in a proper way, knowing that the **setpoint temperature** is the desired temperature for the room, whereas the **ambient temperature** is the real temperature of the room, which can be obtained through an external KNX device. These two temperatures will be compared and its result will determinate the fan speed level.

For a better understanding of this behaviour it is recommended to consult the figure 2.3, which represents the two modes on which Fan Coil can work (Cooling mode and Heating mode) and the temperature differences between the fan speed levels, as well as the hysteresis bands for each of them. Depending on the ambient temperature, the setpoint temperature, the parametrized temperature and hysteresis values, the automatic control of Fan Coil QUATRO will calculate automatically the level which is needed to set the fan speed to fulfil the requirements of the desired temperature.



T = Setpoint temperature t_0, t_1, t_2 = Temperature differences h = Hysteresis

Figure 2.3. Temperature bands for automatic control

2.2. FUNCTIONING MODES

Desires functioning mode can be chosen, as in every climate system, among next ones: **cooling**, **heating** or **both**.

2.2.1. COOLING MODE

It is the default Fan Coil mode. Depending on the chosen control type, different communication objects will be enabled:

- Control applied to the fan: the 1-bit object "Valve (Status)" is enabled, which allows knowing at any moment the valve status ("0" closed, "1", opened).
- Control applied to the valve: besides the valve status object, it is also enabled the 1-bit communication object "PWM Cooling control", to send the open/close orders to the valve, by sending a "1" or a "0", respectively.

2.2.2. HEATING MODE

The same way as before, depending on the parameterized control type, different communication objects will be enabled:

- Control applied to the fan: the 1-bit object "Valve (Status)" is enabled, which allows knowing at any moment the valve status ("0" closed, "1", opened).
- Control applied to the valve: besides the valve status object, it is also enabled the 1-bit communication object "PWM Heating control", to send the open/close orders to the valve, by sending a "1" or a "0", respectively. It will be also possible to configure a delay, in seconds, to switch off the fan, which allows taking the heat for longer, getting a higher energetic efficiency.

2.2.3. BOTH (COOLING AND HEATING)

In this case, and depending on the chosen mode, next communication objects will be provided for Fan Coil management in cooling and heating situations:

- Control applied to the fan: the 1-bit communication object "Mode" is enabled, which allows the switching between modes (writing a "1" for Heating and a "0" for Cooling), and

1-bit object “Mode (Status)”, to know the Fan Coil mode at all times (“1” for Heating and “0” for Cooling). Moreover, the “Valve (Status)” communication objects is available, to know the valve status, depending on the parameterized mode.

- Control applied to the valve: besides the objects described in control applied to the fan, two 1-bit communication objects to control the valve will be shown: “PWM Cooling Control” and “PWM Heating Control”. In this case it is also possible to configure a delay, in seconds, to switch off the fan.

3. ETS PARAMETERIZATION

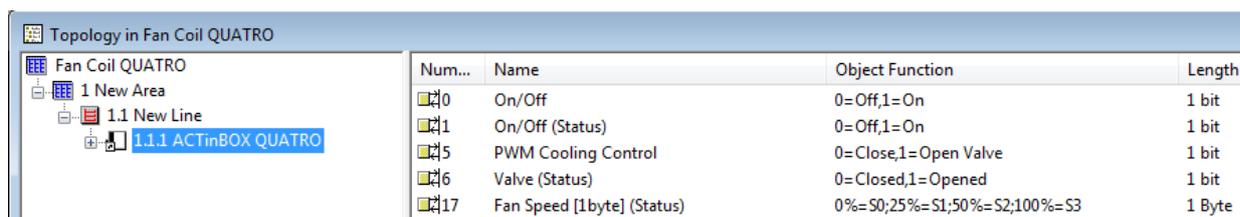
For starting to parameterize the Fan Coil for ACTinBOX QUATRO it is necessary, once the ETS program has been opened, importing the data base of the product (version 1.4 of the application program).

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 Fan Coil QUATRO in ETS.

3.1. DEFAULT CONFIGURATION

This section shows the default configuration from which the device parameterization starts.



Num...	Name	Object Function	Length
0	On/Off	0= Off,1= On	1 bit
1	On/Off (Status)	0= Off,1= On	1 bit
5	PWM Cooling Control	0= Close,1= Open Valve	1 bit
6	Valve (Status)	0= Closed,1= Opened	1 bit
17	Fan Speed [1byte] (Status)	0%= S0;25%= S1;50%= S2;100%= S3	1 Byte

Figure 3.1. Default topology

As it can be seen in the figure 3.1, Fan Coil QUATRO shows several default communication objects:

- "On/Off": 1-bit object to switch on or off the Fan Coil by sending a "1" or "0", respectively.
- "On/Off (Status)": 1-bit object to check the machine status (on/off) at any time.

Note: Take into account that, while the Fan Coil status is OFF, any change in the fan speed or any order over the valve will be ignored until the Fan Coil is switched on.

Fan coil for QUATRO is configured by default to apply a control over the valve and with the "Cooling" mode activated. For that reason, there will be present the following default communication objects:

- "PWM Cooling control": 1-bit object that allows opening or closing the valve by sending a "1" or "0", respectively.

- "Valve (Status)": 1-bit object to check at any time the status (opened/closed) of the valve.
- "Fan: Speed (Status)": 1-byte object to know at any moment the ventilation level, in percentage, according to the Table 3.1:

Off	0%
Speed 1	25%
Speed 2	50%
Speed 3	100%

Table 3.1. Fan speed-Fan percentage ratio

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

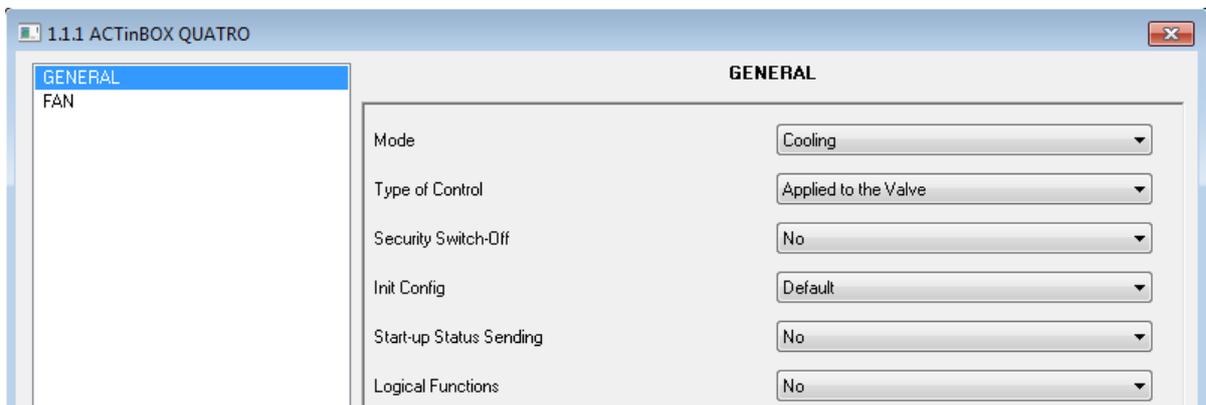


Figure 3.2. Configuration screen by default

In the next sections, the functionality of each parameter of the application Fan Coil QUATRO will be described.

3.2. MODE

Desired functioning mode can be chosen among next ones: cooling, heating or both.

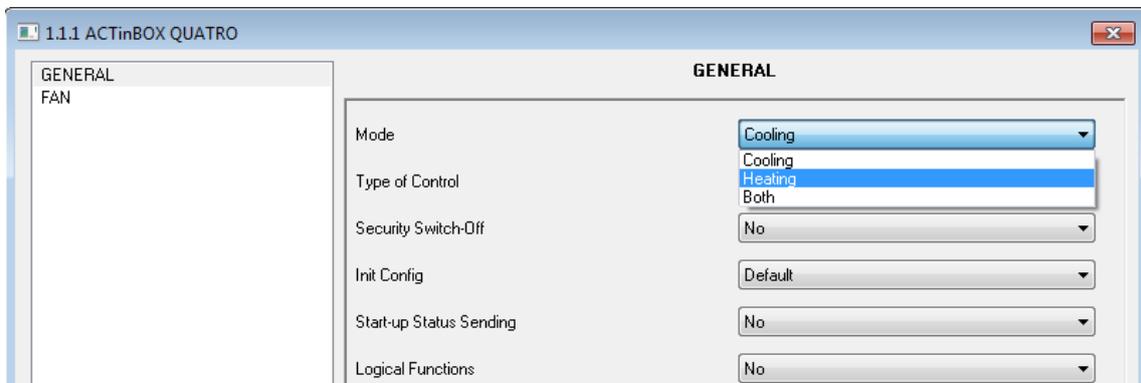


Figure 3.3. Mode selection

3.2.1. COOLING MODE

It is the default Fan Coil mode. Depending on the type of control parameterized, different communication objects to the Fan Coil control in cooling mode will be enabled, which functioning explanation can be read again in the section 2.2.1.

3.2.2. HEATING MODE

Depending on the type of control parameterized, different communication objects to the Fan Coil control in heating mode will be enabled. Its functioning explanation can be read again in the section 2.2.2.

In case of selecting a control applied to the valve, a new drop-down box will be shown: "Fan switch-off delay", to configure a delay, in seconds, after which the fan will switch off ("Fan Speed Status=0%") once sent the order to close the valve ("PWM Heating Control = 0").



3.2.3. BOTH (COOLING AND HEATING)

In this case, and depending on the chosen mode, several communication objects will be provided for Fan Coil management in cooling and heating situations (please read its functioning in the section 2.2.3).

In case of selecting a control applied to the valve, a new drop-down box will be shown: "Fan Switch-off delay (only heating)", to configure a delay, in seconds, after which the fan will switch off ("Fan Speed Status=0%") once sent the order to close the valve ("PWM Heating Control = 0").

3.3. TYPE OF CONTROL

To select one of these control methods: control applied to the fan or control applied to the valve.

3.3.1. CONTROL APPLIED TO THE FAN

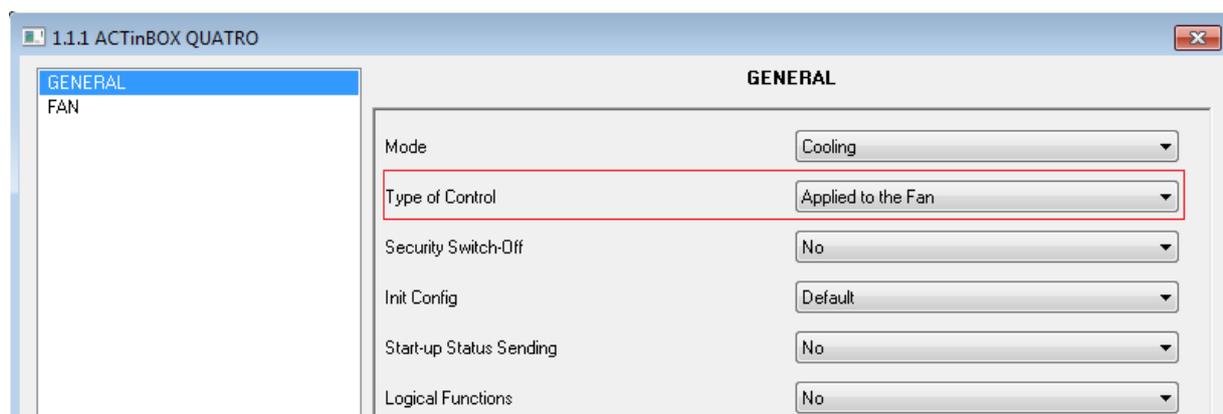


Figure 3.4. Type of control selection: applied to the fan

When selecting this type of control in the general parameterization window, the specific window for the fan configuration is the following:

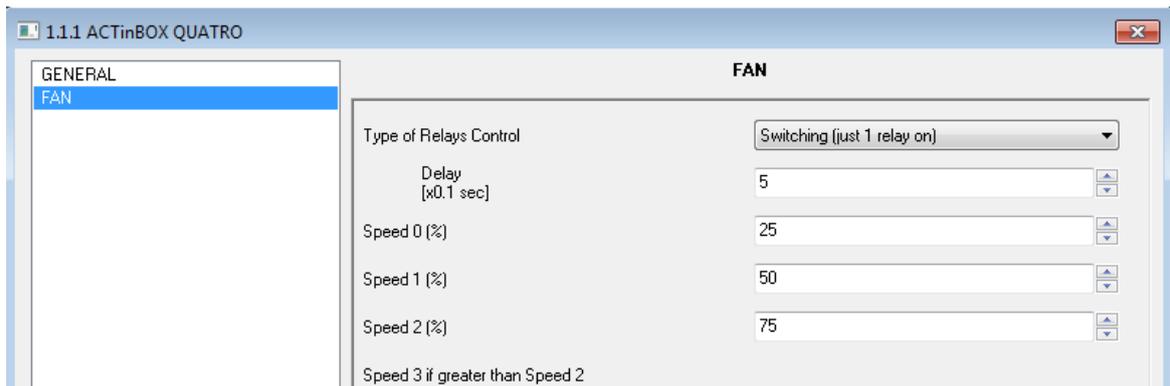


Figure 3.5. Control applied to the fan

Where to configure the following features:

🌐 **Type of relays control:** indicates the way the ventilation level of the fan will change, by closing the relays associated to each actuator output (from output 2 to 4) in charge of the fan speed control. Two options can be chosen:

- **Switching** (just 1 relay on): just one of the 3 outputs of the actuator will be activated for each ventilation level. See figure 3.6.

When selecting this control type over the actuator outputs, a new label is shown ("Delay") that allows the activation of the outputs after a parameterized delay (in tenths of seconds).

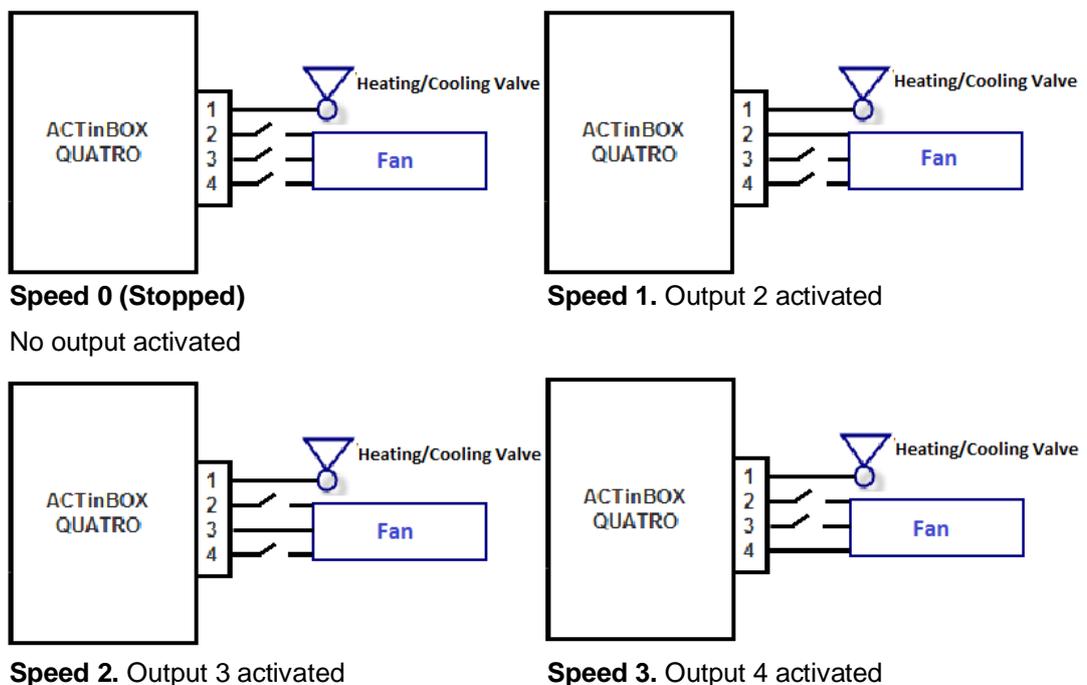


Figure 3.6. Ventilation speed control by switching

➤ **Accumulation** (some relays on): ventilation level is proportional to the number of outputs activated. See figure 3.7.

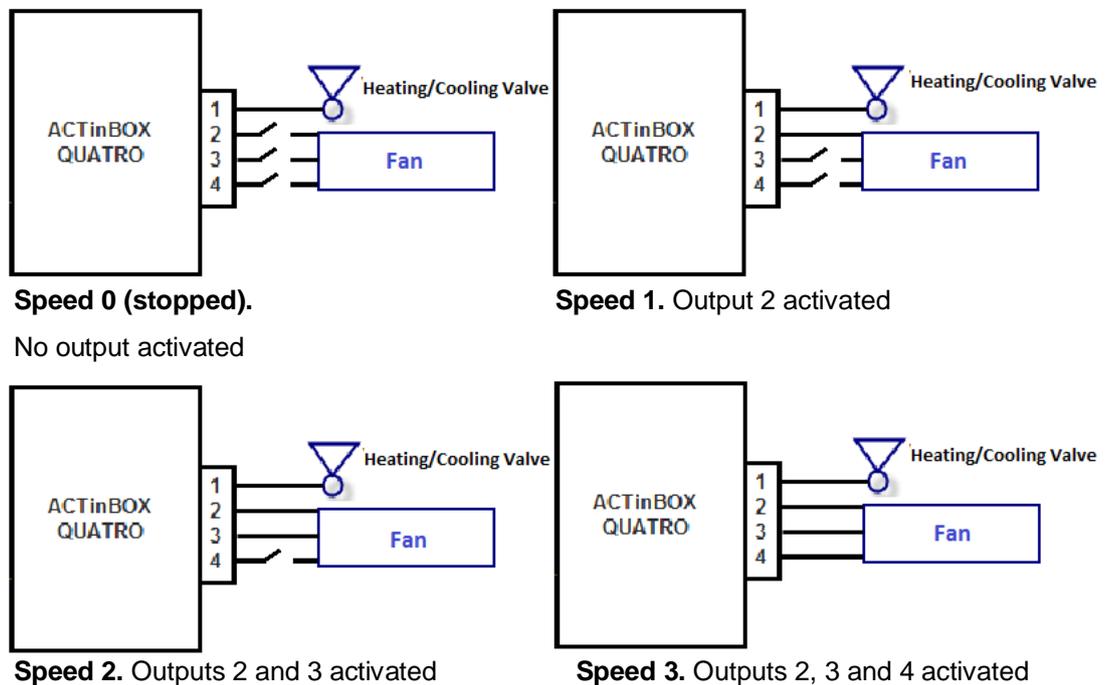


Figure 3.7. Ventilation speed control by accumulation

🌐 **Speed 0, 1, 2 and 3 (%)**: allows the definition of the ventilation percentages associated to each fan speed.

The desired fan speed is given to the fan using a 1-byte communication object “Continuous Fan Control”, enabled by default. Once this value is received, the Fan Coil system will set the fan speed in function of the ranges defined previously.

3.3.2. CONTROL APPLIED TO THE VALVE

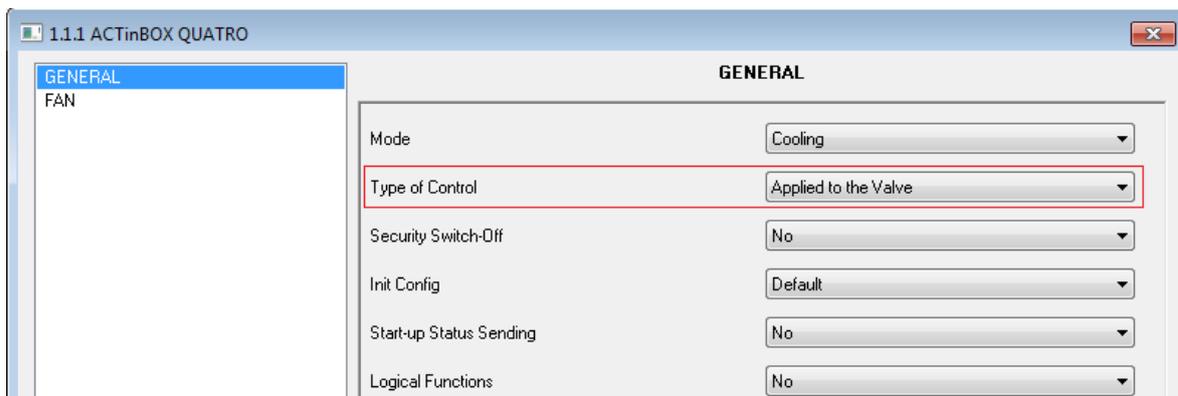


Figure 3.8. Type of control selection: applied to the valve

When selecting this type of control in the general parameterization window, the specific window for the fan configuration is the following:

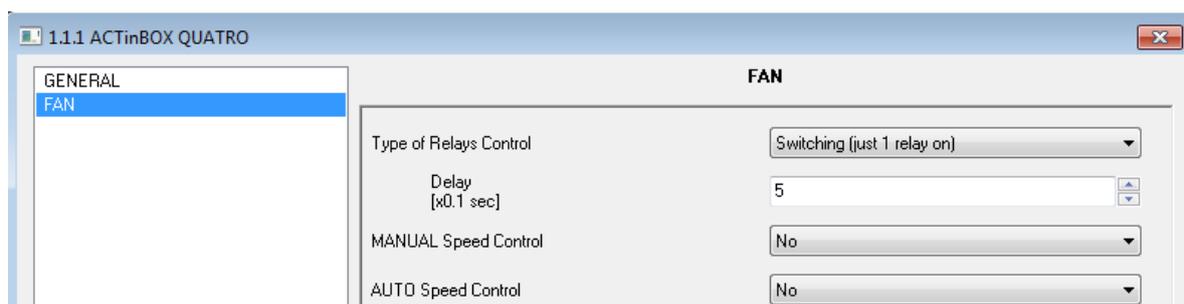


Figure 3.9. Control applied to the valve

In this case, the Fan Coil valve will be controlled through an external KNX device, via communication objects. Specifically, the opening or closing of the valve is commanded by two 1-bit communication objects: "PWM Cooling control" and "PWM Heating control", according to the configured working mode (cooling, heating or both). The value "1" on any of these objects will open the valve. A "0" will close the valve. Next situations can be distinguished, depending on the parametrized mode:

- Cooling mode. The valve will be controlled through the corresponding communication object ("PWM Cooling control").
- Heating mode. The valve will be controlled through the corresponding communication object ("PWM Heating control").

Both modes. In the cooling mode, the valve will follow the orders received through the communication object "PWM Cooling control". In the heating mode, the valve will follow the orders received through the communication object "PWM Heating control". In both cases, changes produced in the object which does not belong to the actual mode will be ignored.

For its part, the fan could be controlled manually or automatically.

The following parameters can be configured:

- Type of Relays control:** same functioning as the explained in the control applied to the fan (see section 3.3.1).
- Manual speed control:** Ventilation level will be controlled independently, using any of the following methods: **Step control**, **Single control** or **Percentage control**. The explanation of the functioning of every of these control methods can be read again in the section 2.1.2 of this manual.

MANUAL Speed Control	Yes
Step Control	Yes
Type	Non Cyclical
Single Control	Yes
Percentage Control	Yes

Figure 3.10. Configuration of the manual speed control methods

- Auto speed control:** the explanation of the functioning of this method can be read again in the section 2.1.2 of this manual.

AUTO Speed Control	Yes
Difference between Ambient and Setpoint	
Speed 0 (Diff0 <= t0) [t0 = x0.1 °C]	0
Speed 1 (t0 < Diff1 <= t0+t1) [t1 = x0.1 °C]	5
Speed 2 (t0+t1 < Diff2 <= t0+t1+t2) [t2 = x0.1 °C]	5
NOTE: Speed 3 (Diff 3 > Diff2)	
Hysteresis [x0.1 °C]	3

Figure 3.11. Auto speed control parameters

3.4. SECURITY SWITCH-OFF

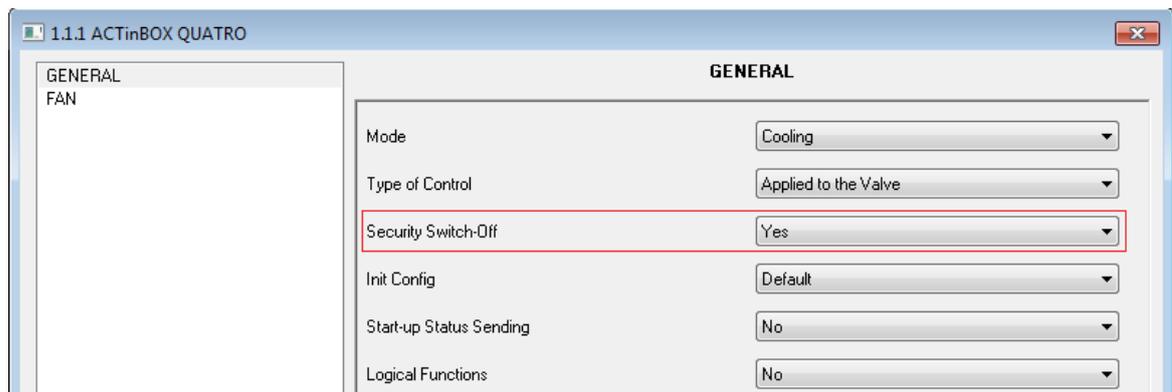


Figure 3.12. Security switch-off

The option Security switch-off allows the **deactivation of the outputs** of ACTinBOX QUATRO in charge of the Fan Coil management (valve ON/OFF and fan speed regulation), for security reasons, when a voltage supply drop occurs, causing the **immediate Fan Coil switched-off**.

Once the voltage supply is recovered, the Fan Coil status will be also recovered and the actuator outputs will be activated or deactivated, depending on their initial configuration (see section 3.5).

3.5. INITIAL CONFIGURATION

It is possible the configuration of the desired Fan coil status after a download from ETS, or after voltage supply drop. This configuration can be default or custom.

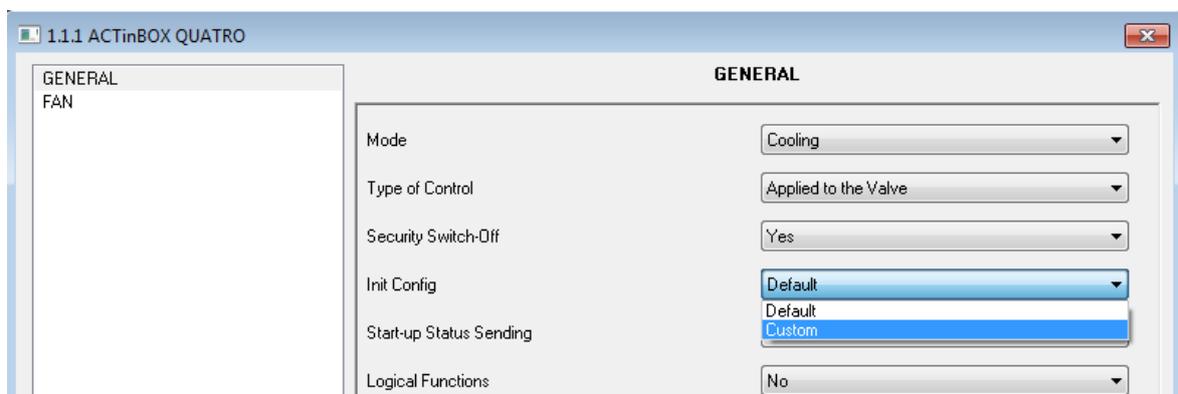
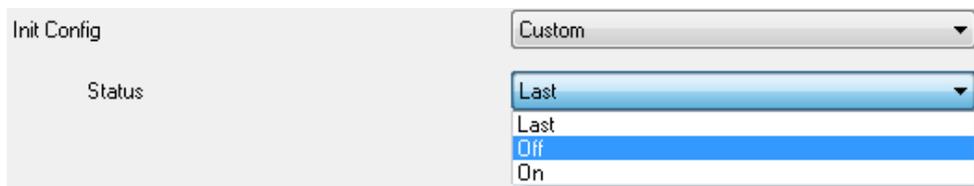


Figure 3.13. Fan Coil start-up configuration

If default configuration is chosen, Fan Coil will recover the last status before the voltage supply drop.

For custom configuration, next options are shown ("Status"):



The screenshot shows a configuration window with two dropdown menus. The first dropdown, labeled 'Init Config', is set to 'Custom'. The second dropdown, labeled 'Status', is set to 'Last' and is currently open, showing three options: 'Last', 'Off', and 'On'. The 'Last' option is highlighted in blue.

- **Last:** Fan Coil will recover the status before the voltage supply drop.
- **Off:** Fan Coil will be switched off after voltage supply recovering or after any download, independently its previous status, and it will be switched on only when a “1” is written in the correspond communication object “On/Off”.
- **On:** Fan Coil will be switched on automatically after the voltage supply recovering or after any download, independently its previous status.

In each case, the ventilation level after the voltage supply recovering depends on its last status; it means, if the fan was in a ventilation level manually established, when the voltage supply is back, it will be situated in this position again (unless Fan Coil initial status is “off”, in which case it will be necessary to switch on the system to recover the fan previous status). However, if the fan was in a level calculated by the application program automatically, it is possible that after the voltage recovering, temperature conditions have changed, so the ventilation level should change depending on the new requirements. Something similar happens with the valve status, which can vary if the temperature conditions change after the voltage failure.

3.6. START-UP STATUS SENDING

If this parameter is activated (“Yes”), the status communication objects of a determined parameterization (like “On/Off (Status)”, “Valve (Status)”, etc.) can be sent to the KNX bus in order to update the rest of devices connected in the installation. These objects could be sent with a parameterized delay (in seconds) or immediately (writing a 0 in the box “Delay”, shown after the activation of Start-up status sending).



The screenshot shows a configuration window with two input fields. The first field, labeled 'Start-up Status Sending', is a dropdown menu set to 'Yes'. The second field, labeled 'Delay [x1 sec]', is a text input box containing the number '3'.

3.7. LOGICAL FUNCTIONS

This section of the application Fan Coil QUATRO is meant to perform binary logic operations with incoming data from the KNX bus, to send the result through other communication objects specifically enabled in the actuator for this operation.

Up to 5 different logical functions can be enabled, independent of each other, which can carry out **up to 4 operations** each. To use any of them, it is necessary to enable them in the following ETS window, which appears when selecting “Yes” in the Logical Functions box of the Fan Coil QUATRO General window.

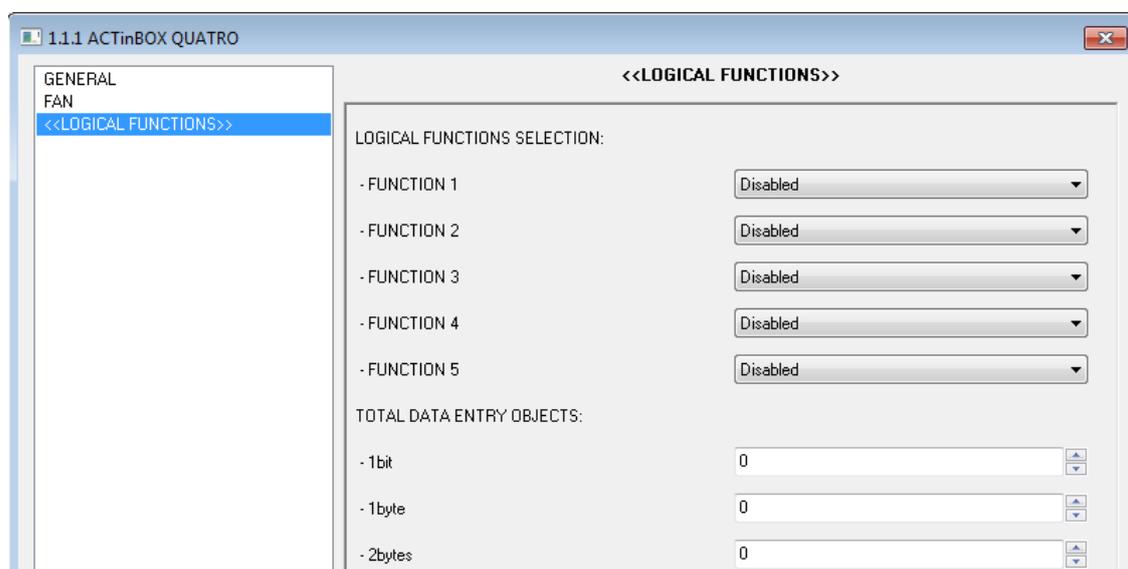


Figure 3.14. Logical functions

To obtain detailed information about the use and ETS parameterization of the logical functions, please consult the specific documentation "**Logical Functions X5**", available at: <http://www.zennio.com>.

ANNEX I. COMMUNICATION OBJECTS

SECTION	NUMBER	SIZE	IN/OUT	FLAGS	VALUES			NAME	DESCRIPTION
					RANGE	1st TIME	RESET		
GENERAL	0	1 bit	I	W	0/1	Indifferent	Indifferent	On/Off	0=Off; 1=On
	1	1 bit	O	RT	0/1	Parameteriz.	Parameteriz.	On/Off (Status)	0=Off; 1=On
	2	1 bit	I	W	0/1	Indifferent	Indifferent	Mode	0=Cooling; 1=Heating
	3	1 bit	O	RT	0/1	Cooling	Parameteriz.	Mode (Status)	0=Cool; 1=Heat
	4	1 bit	I	W	0/1	Indifferent	Indifferent	PWM Heating control	0=Close; 1=Open valve
	5	1 bit	I	W	0/1	Indifferent	Indifferent	PWM Cooling control	0=Close; 1=Open valve
	6	1 bit	O	RT	0/1	0	Parameteriz.	Valve (Status)	0=Closed; 1=Opened
	7	1 bit	I	W	0/1	Indifferent	Indifferent	Fan Step control	0=Lower; 1=Higher
	8	1 bit	I	W	0/1	Indifferent	Indifferent	Continuous Speed 0	0=Ignored; 1=Speed 0
	9	1 bit	I	W	0/1	Indifferent	Indifferent	Continuous Speed 1	0=Ignored; 1=Speed 1
	10	1 bit	I	W	0/1	Indifferent	Indifferent	Continuous Speed 2	0=Ignored; 1=Speed 2
	11	1 bit	I	W	0/1	Indifferent	Indifferent	Continuous Speed 3	0=Ignored; 1=Speed 3

SECTION	NUMBER	SIZE	IN/OUT	FLAGS	VALUES			NAME	DESCRIPTION
					RANGE	1st TIME	RESET		
GENERAL	12	1 bit	O	RT	0/1	Indifferent	Indifferent	Continuous Speed 0 (Status)	Speed 0
	13	1 bit	O	RT	0/1	Indifferent	Indifferent	Continuous Speed 1 (Status)	Speed 1
	14	1 bit	O	RT	0/1	Indifferent	Indifferent	Continuous Speed 2 (Status)	Speed 2
	15	1 bit	O	RT	0/1	Indifferent	Indifferent	Continuous Speed 3 (Status)	Speed 3
	16	1 byte	I	W	0-255 (0-100%)	Indifferent	Indifferent	Continuous Speed [1 byte]	0%=0;1-25%=1;26-50%=2;51-100%=3
	17	1 byte	O	RT	0-255 (0-100%)	0	Parameteriz.	Continuous Speed [1 byte] (Status)	0%=S0;25%=S1;50%=S2;100%=S3
	18	1 bit	I	W	0/1	Indifferent	Indifferent	Continuous Auto	0=Deactivate; 1=Activate
	19	1 bit	O	RT	0/1	0	Parameteriz.	Continuous Auto (Status)	0=Disabled; 1=Enabled
	20	2 bytes	I	W	0-95°C	Indifferent	Last	Ambient temperature	Setpoint temperature
	21	2 bytes	I	W	0-95°C	25°C	Last	Setpoint temperature	Setpoint temperature
	22	1 byte	I	W	0-255 (0-100%)	Indifferent	Indifferent	Continuous Fan control[1 byte]	0-100%
LOGICAL FUNCTIONS	23-38	1 bit	I	W	0/1	0	Last	[LF] (1 bit) Data Entry 1 ... [LF] (1 bit) Data Entry 16	Binary data entry (0/1) ... Binary data entry (0/1)
	39-46	1 byte	I	W	0-255	0	Last	[LF] (1 byte) Data Entry 1 ... [LF] (1 byte) Data Entry 8	1 byte data entry (0-255) ... 1 byte data entry (0-255)

SECTION	NUMBER	SIZE	IN/OUT	FLAGS	VALUES			NAME	DESCRIPTION
					RANGE	1st TIME	RESET		
LOGICAL FUNCTIONS	47-54	2 bytes	I	W	0-FFFF	0	Last	[LF] (2 bytes) Data Entry 1 ... [LF] (2 bytes) Data Entry 8	2 byte data entry (0-FFFF) ... 2 byte data entry (0-FFFF)
	55-59	1 bit	O	RT	0/1	0	Last	[LF] Function 1 RESULT (1 bit) ... [LF] Function 5 RESULT (1 bit)	Function 1 RESULT ... Function 5 RESULT
	60-64	1 byte	O	RT	0-255	0	Last	[LF] Function 1 RESULT (1 byte) ... [LF] Function 5 RESULT (1 byte)	Function 1 RESULT ... Function 5 RESULT
	65-69	2 bytes	O	RT	0-FFFF	0	Last	[LF] Function 1 RESULT (2 bytes) ... [LF] Function 5 RESULT (2 bytes)	Function 1 RESULT ... Function 5 RESULT
					0°C-120°C	25°C	Last	[LF] Function 1 RESULT (2 bytes) ... [LF] Function 5 RESULT (2 bytes)	Function 1 RESULT ... Function 5 RESULT



BECOME USER!

<http://zennioenglish.zendesk.com>

TECHNICAL SUPPORT