Sensors

Cubik-T

v1.0

Programming manual

ingenium

www.besknx.com

Index

1 General description

The Cubik-T (Ref. SR59X1X0) is a temperature sensor flush-mounted in box with vented cap mechanism, which also incorporates PI control thermostat functions and a small logic module.

It is suitable for use as an additional temperature probe for any KNX temperature control system but also can be used as a main control thermostat in a room as it incorporates all necessary functions for it.

It allows you to control heating or air conditioning using a simple 2-point control with hysteresis or more advanced form using PI automatic regulation algorithms with output pulse width (PWM) modulation or continuous value. By using IP, a more precise control of the room temperature, improving comfort and energy saving, is achieved.

Used as the main controller allows to refer its own internal temperature measure or any external sensor measure. It also lets you add an additional on/off heat or cold output to control the main climate system support. You can program the heating and cooling modes switched for different seasons or simultaneously for places with more variations of temperature. Moreover, the logic unit allows to program simple logic operations using bus objects, for example to control a floor heating recirculation according to several rooms.

General characteristics:

- Temperature sensor 0 to 51°C.
- Temperature Controller: 2 points, or continuous PWM.
- PI control algorithm.
- Control system hot / main heat output and additional on/off.
- Flush mounting in box with vented cap mechanism.
- Simple logic unit.

2 Technical information

KNX supply	29Vdc from KNX BUS.
KNX consumption	10mA from KNX BUS.
Mounting	Flush mounting in box with vented cap mechanism.
Size	55 x 45 x 5mm
Connection	KNX BUS connection
Temperature sensor	0 - 51ºC
Sensor accuracy	+/- 1ºC Possible to calibrate probe by parameter
Regulation	2 controllers with PI regulation algorithm
Working modes	Cool / heat (summer / winter) or auto with additional output.
Environment temperature range	Operation: -10ºC / 55ºC Storage: -30ºC / 60ºC Transportation: -30ºC / 60ºC
Regulation	According to the directives of electromagnetic compatibility and low voltage: EN 50090-2-2 / UNE-EN 61000-6-3:2007 / UNE-EN 61000-6-1:2007 / UNE-EN 61010-1

3 Programming

3.1 Application program information

Manufacturer: Ingenium

Application: Stibus-K_Cubik-T

Maximum number of communication objects: 28

Maximum number of assignments: 28

Version: v1.0

3.2 Communication objects table

Object	Name | Function	Length	DPT	Flags				
0	Temperature | Measured temperature	2 bytes	9.001	C	R		T	
0	Temperature | External measured temperature	2 bytes	9.001	C	R	W		
1	Temperature | Set point temperature	2 bytes	9.001	C		W		
2	Temperature | Set point temperature status	2 bytes	9.001	C	R		T	
3	Mode | Controller switch on/off	1 bit	1.001	C		W		
4	Mode | Controller switch on/off status	1 bit	1.001	C	R		T	
5	Mode | Heating/Cooling switch over (=1 / 0)	1 bit	1.001	C		W		
6	Mode | Heating/Cooling status (=1 / 0)	1 bit	1.001	C	R		T	
5	Mode | Heating (1=set mode / 0=nothing)	1 bit	1.001	C		W		
6	Mode | Heating status	1 bit	1.001	C	R		T	
7	Mode |Cooling (1=set mode / 0=nothing)	1 bit	1.001	C		W		
8	Mode | Cooling output	1 bit	1.001	C	R		T	
9	Mode | Auto (1=set mode / 0=nothing)	1 bit	1.001	C		W		
10	Mode | Auto status	1 bit	1.001	C	R		T	
11	Controller | Heating output (2 steps)	1 bit	1.001	C			T	
11	Controller | Heating output (pwm)	1 bit	1.001	C			T	
12	Controller | Heating output (continuous)	1 byte	5.001	C			T	
13	Controller | Cooling output (2 steps)	1 bit	1.001	C			T	
13	Controller | Cooling output (pwm)	1 bit	1.001	C			T	

14	Controller | Cooling output (continuous)	1 byte	5.001	C			T	
15	Mode | HVAC Mode	1 byte	20.102	C		W		
16	Mode | HVAC Mode status	1 byte	20.102	C	R		T	
17	Logic 1 (OR/NOR) | Input 1	1 bit	1.002	C		W		
18	Logic 1 (OR/NOR) | Input 2	1 bit	1.002	C		W		
19	Logic 1 (OR/NOR) | Input 3	1 bit	1.002	C		W		
20	Logic 1 (OR/NOR) | Input 4	1 bit	1.002	C		W		
21	Logic 1 (OR/NOR) | Output	1 bit	1.002	C			T	
22	Logic 2 (AND/NAND) | Input 1	1 bit	1.002	C		W		
23	Logic 2 (AND/NAND) | Input 2	1 bit	1.002	C		W		
24	Logic 2 (AND/NAND) | Input 3	1 bit	1.002	C		W		
25	Logic 2 (AND/NAND) | Input 4	1 bit	1.002	C		W		
26	Logic 2 (AND/NAND) | Output	1 bit	1.002	C			T	
27	Controller | Additional heating output	1 bit	1.001	C			T	
28	Controller | Additional cooling output	1 bit	1.001	C			T	

3.3 Communication objects description

Object	**Object 0: Temperature – Measured temperature**
Function	2 bytes-Communication object for reading or notification Room temperature.
Description	This object can be read or notify the temperature sensor is measuring. You can set the temperature sending the bus cyclically or change by transmission temperature measured parameter.
Object	**Object 0: Temperature – External measured temperature**
Function	2 bytes communication object for writing temperature measured by an external sensor.
Description	By this object is possible to use an external temperature sensor and temperature measurement using as a reference for the driver.
Object	**Object 1: Temperature – Set point temperature**
Function	2 bytes communication object for writing a new set point controller.
Description	This communication object allows you to set a new temperature set point to the temperature controller. After a write, the controller will adjust its output to achieve and maintain the new temperature value of the room as programming.

Object	Object 2: Temperature – Set point temperature status
Function	2 bytes communication object for reading or notification of the current set point temperature.
Description	This communication object allows to read or notify the current set point temperature controller.
Object	**Object 2: Temperature – Set point temperature status**
Function	1 bit communication object to turn on or off the temperature controller.
Description	Writing on this subject can be switched on (=1) or off (=0) the temperature controller. Turning the controller off to the heating and/or cooling is sent and all calculation functions work and what is left to control the temperature of the room stop.
Object	**Object 4: Mode – Controller switch on/off status (=1 / 0)**
Function	2 bytes communication object for reading or reporting the current status of the controller.
Description	This object can be read or notify if the controller is on or off.
Object	**Object 5: Mode – Heating/Cooling switch over (=1 / 0)**
Function	1 bit communication object for changing the controller work mode between heat and cold.
Description	Writing in this object can switch the operating mode between heat (= 1) and cold (= 0). In heating mode (winter), if the set temperature is higher than the temperature measured, the calculation of the controller will increase the work of the heating system. Cold mode (summer): if the set point temperature is less than the temperature measured, the controller increases the work of the refrigeration system
Object	**Object 6 : Mode - Heating/Cooling switch status**
Function	1 bit communication object for reading or reporting the current status of the controller.
Description	By this object is possible to read or notify if the controller is in heat or cool mode.
Object	**Object 5: Mode – Heating (1=set mode / 0=nothing)**
Function	1 bit communication object to set heat as operating mode of the controller.
Description	Writing a 1 in this object you can change the working mode to heating. Writing a value of 0 does not have any action. In heating mode (winter), if the set temperature is higher than the temperature measured, the calculation of the controller will increase the work of the heating system.
Object	**Object 6: Mode – Heating status**
Function	Communication object 1 bit for reading or notification of the current status of this mode.
Description	By this object is possible to read or notify if the controller is in heating mode (1 = enabled heating mode, heating mode 0 = disabled).
Object	**Object 7: Mode – Cooling (1=set mode/ 0=nothing)**
Function	1 bit communication object to establish working mode of the controller.
Description	Writing a 1 in this order is possible to change the working mode to cooling. Writing a value of 0 does not have any action. Cold mode (summer) if the set point temperature is less than the temperature measured, the controller increases the work of the refrigeration system.

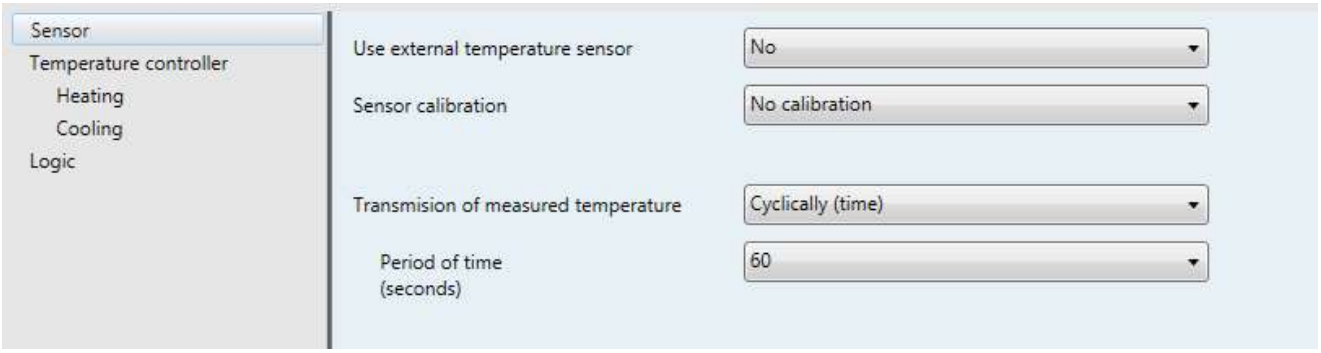
Object	Object 8: Mode – Cooling status
Function	Communication object 1 bit for reading or notification of the current status of this mode.
Description	By this object is possible to read or notify if the controller is in cooling mode (1 = enabled cooling mode, 0= cooling mode disabled).
Object	**Object 9: Mode – Auto (1=set mode / 0=nothing)**
Function	1 bit communication object to set auto as controller mode.
Description	Writing a 1 in this object you can change the work mode to auto. Writing a value of 0 does not have any action. In auto mode, the controller works both ways simultaneously. It automatically decides whether it is necessary to heat or cool. The temperature of the room is set by Object 1 – set point temperature and is the same for both modes heat and cold.
Object	**Object 10: Mode - Auto status**
Function	Communication object 1 bit for reading or notification of the current status of this mode.
Description	This object can be read or notify if the controller is in auto mode (1 = auto mode enabled, 0 = auto off mode).
Object	**Object 11: Controller – Heating output (2 steps)**
Function	1 bit communication object to control the heating system.
Description	By this object the heating system is controlled by the 2-step control (on / off).
Object	**Object 11: Controller – Heating output (pwm)**
Function	1 bit communication object to control the heating system.
Description	By this object the heating system is controlled by a pulse width modulation control (PWM).
Object	**Object 12: Controller – Heating output (continuous)**
Function	1 byte communication object to control the heating system.
Description	By this object the heating system is controlled by a continuous PID control.
Object	**Object 13: Controller – Cooling output (2 steps)**
Function	1 bit communication object to control the cooling system.
Description	By this object the cooling system is controlled by the 2-step control (on / off).
Object	**Object 13: Controller – Cooling output (pwm)**
Function	1 bit communication object to control the cooling system.
Description	By this object the cooling system is controlled by a pulse width modulation control (PWM).
Object	**Object 14: Controller – Cooling output (continuous)**
Function	1 byte communication object to control the heating system
Description	By this object the heating system is controlled by a continuous PID control .

Object	Object 15: Mode – HVAC mode
Function	1 byte communication object to establish the special HVAC mode.
Description	By this object is possible to set the desired special mode controller according to the following values (writing a value other than these will not have any action): 1 = Comfort 2 = Stand-by 3 = Economic 4 = Protection When selecting a special mode, the controller establishes a reference saved in accordance with the parameter (heat / cold) selected depending on where you are in that moment. After selecting a special mode, if the controller function switches between heating/cooling, set point changes to its current mode.
Object	**Object 16: Mode – HVAC mode status**
Function	1 byte communication object for reading or notifying the current status of this mode.
Description	By this object can be read or notified if the controller is in Comfort, Stand-by, Economy or Protection mode. If the password is changed manually using the Object 1 - Temperature set point special mode is notified.
Object	**Object 17: Logic 1 (OR/NOR)– Input 1**
Function	1 bit communication object for writing the logic input.
Description	By this object is possible to write a new value in the input 1 of OR or NOR logic function 1.
Object	**Object 18: Logic 1 (OR/NOR)– Input 2**
Function	1 bit communication object for writing the logic input.
Description	By this object is possible to write a new value in the input 2 of OR or NOR logic function 1.
Object	**Object 19: Logic 1 (OR/NOR)– Input 3**
Function	1 bit communication object for writing the logic input.
Description	By this object is possible to write a new value in the input 3 of OR or NOR logic function 1
Object	**Object 20: Logic 1 (OR/NOR)– Input 4**
Function	1 bit communication object for writing the logic input.
Description	By this object is possible to write a new value in the input 4 of OR or NOR logic function 1
Object	**Object 21: Logic 1 (OR/NOR)– Output**
Function	1 bit communication object for notifying the logic output.
Description	By this object is possible to notify the result of the OR or NOR operation of logic function 1. The result is sent whenever a change occurs in one input.
Object	**Object 22: Logic 2 (AND/NAND)– Input 1**
Function	1 bit communication object for writing the logic input.
Description	By this object is possible to write a new value in the input 1 of AND or NAND logic function 2.

Object	Object 23: Logic 2 (AND/NAND)– Input 2
Function	1 bit communication object for writing the logic input.
Description	By this object is possible to write a new value in the input 2 of AND or NAND logic function 2.
Object	**Object 24: Logic 2 (AND/NAND)– Input 3**
Function	1 bit communication object for writing the logic input.
Description	By this object is possible to write a new value in the input 3 of AND or NAND logic function 2.
Object	**Object 25: Logic 2 (AND/NAND)– Input 4**
Function	1 bit communication object for writing the logic input.
Description	By this object is possible to write a new value in the input 4 of AND or NAND logic function 2.
Object	**Object 25: Logic 2 (AND/NAND)– Output**
Function	1 bit communication object for writing the logic input.
Description	By this object is possible to notify the result of the AND or NAND operation of logic function 2. The result is sent whenever a change occurs in one input.
Object	**Object 27: Controller – Additional heating output**
Function	1 bit communication object to control the heating system.
Description	By this object an additional output (on / off) of the heating system that works to support the principal when the deviation between the set point and the measurement is too large is controlled.
Object	**Object 28: Controller – Additional cooling output**
Function	1 bit communication object to control the cooling system.
Description	By this object an additional output (on / off) of the cooling system that works to support the principal when the deviation between the set point and the measurement is too large is controlled.

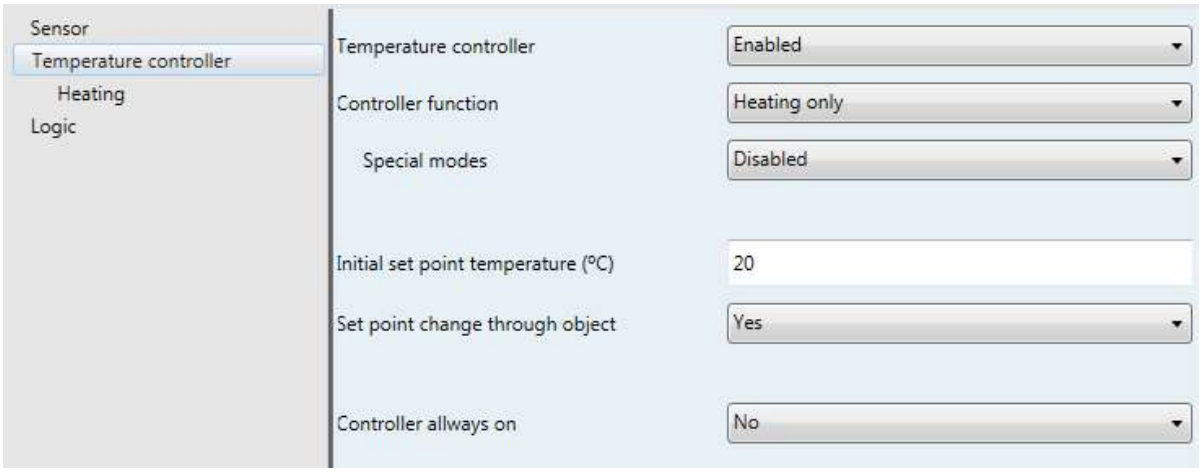
3.4 Parameters

3.4.1 Sensor

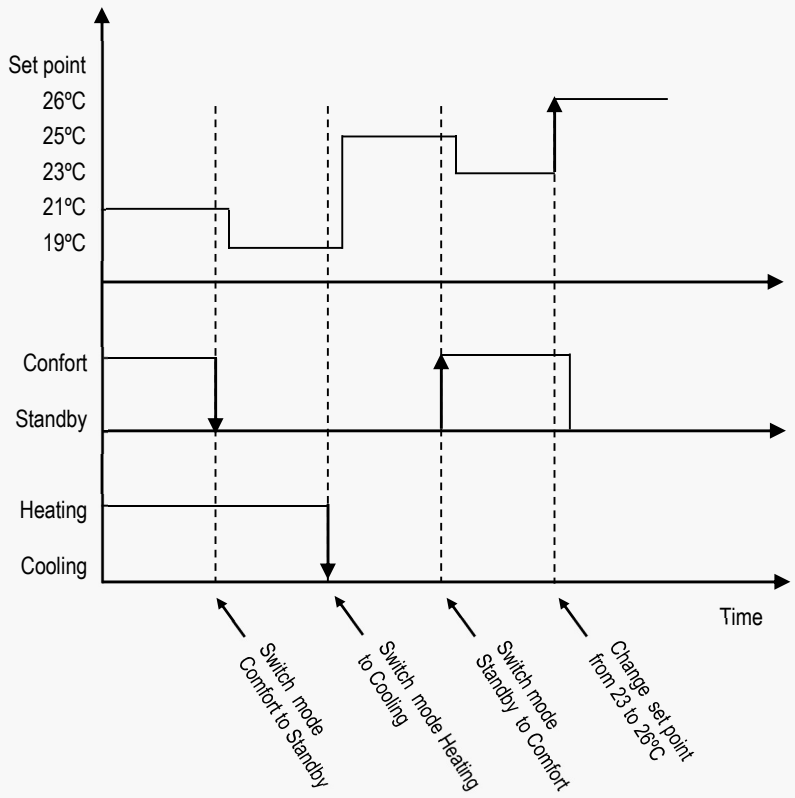


Parameter	Use external temperature sensor
Values	Yes / No
Description	This parameter allows to choose if the controller reference is the internal temperature probe or the measured form another sensor connected to the KNX system. - No: The controller will use its internal measure. You can calibrate the measurement and set up recurring or change notifications. The value can be read or notify via Object 0 - Measured temperature. - Yes: The temperature controller used as a reference value written to the Object 0 - Measured temperature and despises its internal measure.
Parameter	**Sensor calibration**
Values	From -8ºC to +5ºC
Description	This parameter allows you to correct the measurement of the temperature sensor if necessary. The set value is added or subtracted to the current measurement sensor, which can be checked by Object 0: Measured temperature.
Parameter	**Transmission of measured temperature**
Values	Only read, cyclically (time), on change (ºC) or both (ºC y time).
Description	This parameter is used to configure the frequency with which to send notifications of temperature by Object 0: measured temperature.
Parameter	**Period of time (seconds)**
Values	From 5 to 80
Description	If the transmission measured temperature parameter is set as cyclically or both, this additional parameter allows to define the period of sending notifications in seconds.
Parameter	**Minimum change**
Values	From 0,2 to 3,2 ºC
Description	If the transmission measured temperature parameter is set as on change (ºC) or both, this additional parameter allows to define the minimum change necessary to send a notification. When sending a minimum change notification, the periodical sending countdown restarts.

3.4.2 Temperature controller



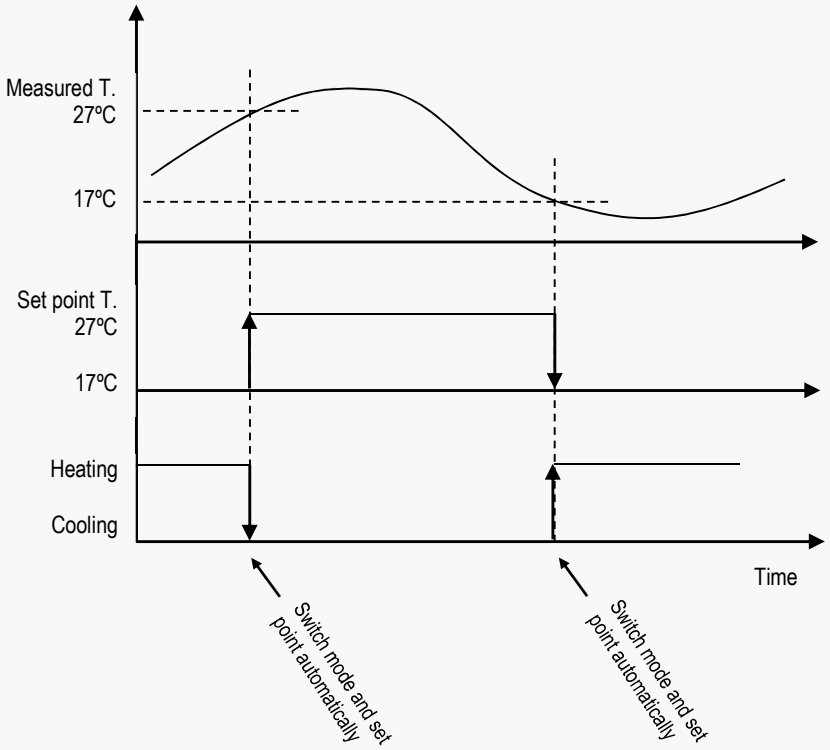
Parameter	Temperature controller
Values	Enabled / Disenabled
Description	This parameter lets you choose whether the device uses the temperature controller or not. If not in use, the equipment can simply be used as a temperature sensor.
Parameter	**Controller function**
Values	Heating only, Cooling only, Heating/Cooling switch over, Heating/Cold simultaneous.
Description	By this parameter, controller function is chosen from the following: - Heating only: The controller only manages the heating (winter mode). - Cooling only: The controller only manages the air conditioning system (summer mode). - Heating/Cooling Switch over: The controller manages both systems separately. The mode change is effected by corresponding communication objects (can be a single object or separate objects). - Heating/Cooling simultaneous: The controller manages both systems at once, and can automatically switch between them if necessary.
Parameter	**Special modes**
Values	Enabled / Disenabled
Description	This parameter is enabled and disabled and special modes Comfort, Stand-by, Economy and Protection. These modes can be set using the Object 15 – HVAC mode sending the corresponding value. When selecting a special mode, the controller adjusts its temperature set point according to the memorized parameter mode (hot or cold) in which it is at that time. After selecting a special mode, if the controller mode is switched between heating/cooling, set point changes to its mode according to the parameters.



Parameter	Change objects
Values	One object / Separated object
Description	This parameter allows you to choose which objects use to change the function of the controller. - Object: a single communication object is used to switch between hot and cold Object 5 Heat/Cool switch (= 1/ 0). Writing a 1 in this object the controller switches to heating mode while if you write 0 mode it switches to cold. - Separate objects: - Heating (mode 1 = Set / 0 = no) and Object 7 - Cooling (mode 1 = Set / 0 = no) two communication objects are used different Object 5. If you write a 1 in these objects is changed to the corresponding mode while writing a 0 has no action. (Note: Changing the heating / cooling mode, if the controller set point is set by selecting a special way, it will change to the special mode. If you have set using the Object 1 - Set point temperature will be maintained.)

Parameter	Automatic heating/cooling change
Values	No deviation (1 point) / With deviation.
Description	If the controller function is set to heating / cooling simultaneously, this parameter allows you to choose the way the controller decides when to switch from one mode to another. - No deviation: the controller will work with both systems (heating and cooling) to maintain a single temperature set point which can only be changed by the Object 1 - Temperature. If the auto mode is activated by Object 9 – Auto mode the controller will change from cold to heat when the reference is greater than the measure and viceversa when it is less. - With deviation: The function of heat or cold is changed depending on the difference between the measured temperature and the set point according to the value set in the deviation parameter to change heat / cold (see next item). In both cases it is possible to change manually the heating/cooling mode via the corresponding communication objects.

Parameter	Heating/Cooling change drift
Values	Value of opposite special mode, fixed value of 1 ° C to 15 ° C.
Description	- Value of opposite special mode: Switching between heating/cooling is done when the measurement is greater or less than the set point of the current special mode. For example: The default set points for special economic mode is 17 or 27 °C for heat and cold respectively. If the controller is in economy mode and heat and the temperature of the room rises above 27 °C (set point especially opposite: cold), then change to cool mode.



- Fixed value (1 to 15 ° C): Switching between heating/cooling is done when the difference between the current set point and the measurement is greater than the selected value.

In both cases it is possible to manually change mode heating/cooling by the corresponding communication objects.

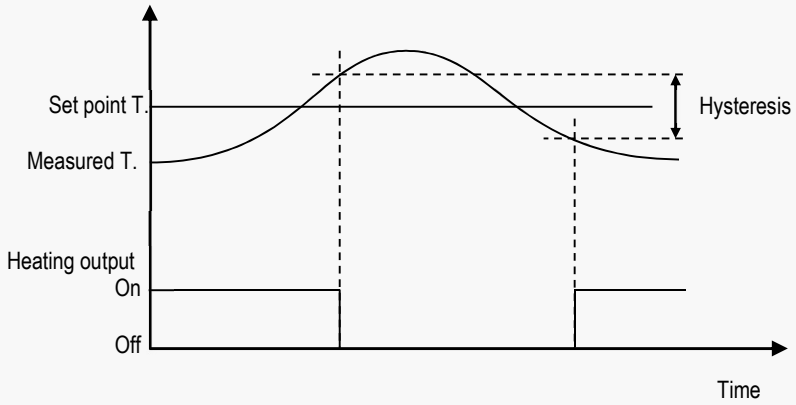
(Note: An incorrect setting of these parameters and / or set points special modes can cause instability of the controller.)

Parameter	**Initial set point temperature (ºC)**
Values	-1000 to 1000 ºC.
Description	It is the set point temperature of the controller in which it is initialized after downloading the application program from the ETS.
Parameter	**Set point change through object**
Values	Yes / No
Description	It is allows to enable or disenable the Object 1 – Set point temperature.
Parameter	**Controller always on**
Values	Yes / No
Description	- Yes: The controller is initialized on after downloading the application from the ETS and you cannot turn it off. - No: Object 3 is enabled - Controller on/off (=1/0) for switching the controller via a bus telegram.

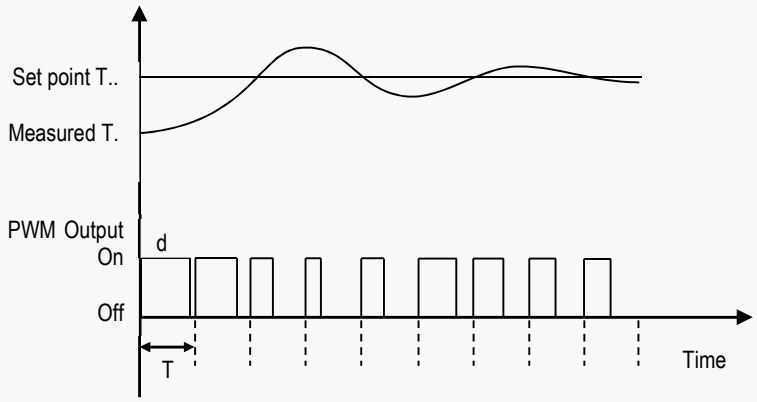
3.4.2.1 Heating / cooling

The following parameters let you set the type of controller for each operating mode, heating and cooling, being similar for both.

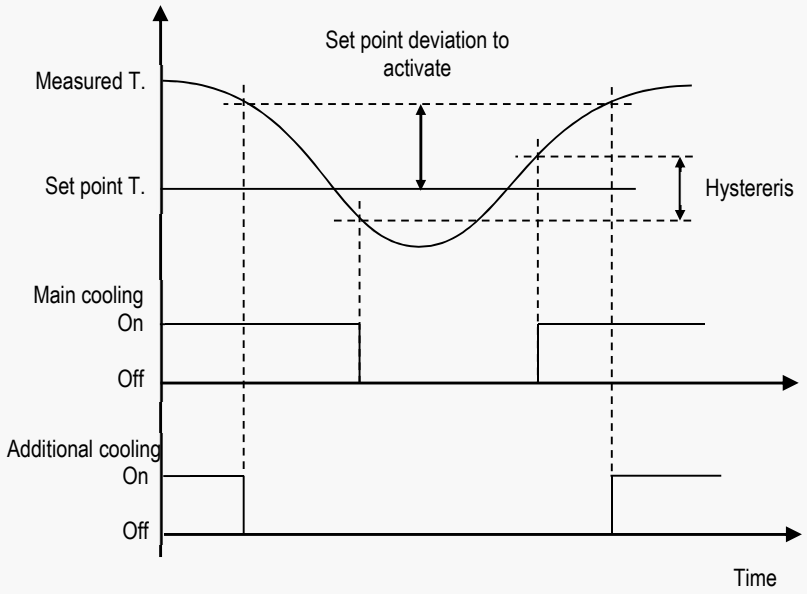
Parameter	Controller type
Values	2 steps, PWM (1 bit), Continuous (1 byte)
Description	- 2 steps is a type of on/off control with hysteresis (see next item). In heating mode, the control output (object 11) is activated when the set point is greater than the measure and is disabled otherwise. In cooling mode the control output of cooling (object 13) is activated when the reference is less than the measure and disabled otherwise. - PWM (1 bit): PI control (items 11 and 13) allows a more accurate room temperature control. The controller output is PWM modulation (Pulse Width Modulation), an on/off signal which switches in a fixed period of time "T" and whose pulse width "d" is calculated by the control (see parameter PWM cycle time). - Continuous (1 byte): PI control (items 12 and 14) enables more accurate room temperature control. In this case the controller output is proportional signal (0- 100%).
Parameter	**Heating/Cooling system**
Values	Floor heating, Water radiators, Electric radiators, Convection or Custom.
Description	By this parameter, it is possible to set the PI constants of the controller which controls the heating or cooling depending on the type of system installed. - Floor heating (5K / 240min). - Water radiators (5K / 150min). - Electric radiators (4K / 140min). - Convection (4K / 90min). - Custom: Allows you to set manually the proportional and integral constants of the PI controller. (Note: An incorrect setting of these parameters can lead to instability of the controller).
Parameter	**Hysteresis**
Values	0,2 to 5 ºC.
Description	When using a 2-steps control, if the temperature is stabilized close to the set point, the output can be switched too frequently. This parameter is obtained leave a dead band above and below the set point to reduce this effect (see next image).



Parameter	PWM cycle time
Values	4 to 28 min
Description	When using a PWM control, this parameter allows you to set the value of the period "T" of the output waveform (see picture below).



Parameter	Output value send on change of
Values	0.5 to 15 %
Description	When using a continuous control, this parameter determines the minimum change in the output signal (from 0 to 100%) in order to send the value through the bus.
Parameter	**Additional heating/cooling output**
Values	Enabled / Disenabled
Description	This parameter allows you to configure the controller to use an additional output (on / off) of the heating or cooling system which works to support the principal when the deviation between the set point and the measure is too big (as the next parameter).
Parameter	**Deviation from set point to activate**
Values	1 to 20 ºC.
Description	The value set in this parameter defines when the additional heating/cooling output is activated or deactivated. When the difference between the set point and the measurement is higher than the deviation, additional output is activated (see next image).



3.4.2.2 Special modes

Parameter	Heating/Cooling set points
Values	0 to 51 ºC
Description	By these parameters is possible to define the set points for each special mode, depending on the controller function is defined heating or cooling. These are the default values for the four special modes: - Comfort: Heating = 21 °C / Cooling: 23 °C - Stand–by: Heating= 19 ° C / Cooling 25 °C - Economic: Heating = 17ºC / Cooling 27ºC - Protection: Heating = 9 ° C / Cooling = 35 ºC

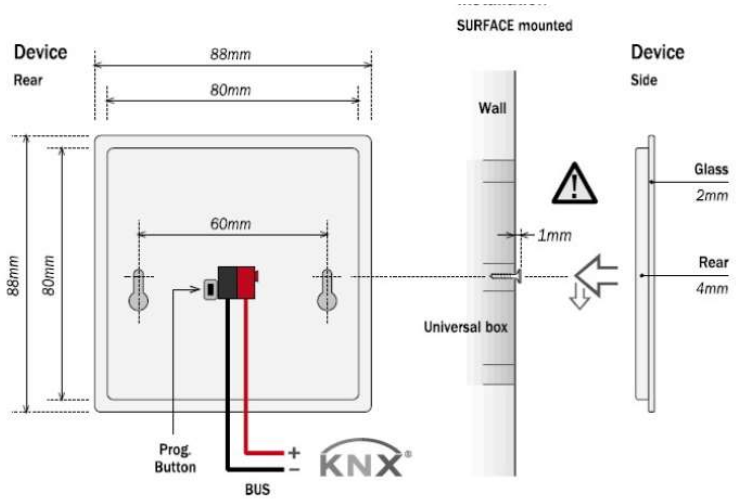
3.4.3 Logic

Enabling logic functions it is passed to have a set of parameters and communication objects that allow easy simple logic operations on the controller.

Parameter	Logic functions
Values	Enabled / Disenabled
Description	When enabling logic functions it is passed to have 2 operating doors OR, NOR, AND and NAND with which you can perform advanced programming using bus telegrams.

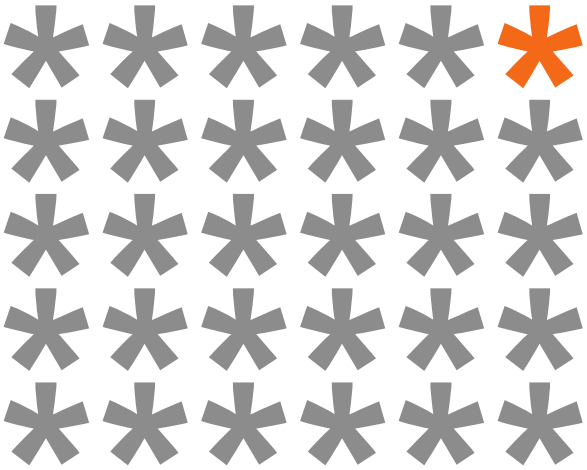
Parameter	Function
Values	OR, NOR, AND or NAND
Description	It defines the operation type which will be performed by the logic function.
Parameter	**Number of inputs**
Values	2, 3 or 4
Description	It defines the number of inputs which are used to calculate the selected logic function. According to this parameter are available a higher or lower number of communication objects
Parameter	**Initial values**
Values	1 / 0
Description	It defines the initial value of each input logic function after downloading the application program from the ETS.

4 Installation



Feed low voltage lines (BUS) in separate ducting to that of power (230V) to ensure there is enough insulation and avoid interferences.

Do not connect the main voltages (230 V) or any other external voltages to any point of the BUS or inputs.



KNX products by ingenium

ingenium Bes

www.besknx.com

Ingenium, Ingeniería y Domótica S.L.

Parque Tecnológico de Asturias, Parcela 50

33428 Llanera, Asturias, Spain

T (+34) 985 757 195

tec@besknx.com

www.besknx.com

www.ingeniumsl.com

Manual version: v1.0