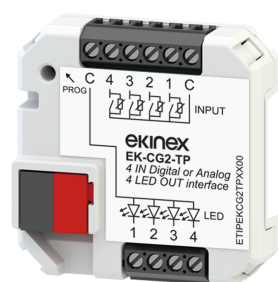




APPLICATION MANUAL EK-CG2-TP



**UNIVERSAL INTERFACE
4 DIN/NTC
FOR DIGITAL INPUT AND NTC PROBE**

Summary

2.1 Input functions	6
2.2 Temperature probes and thermostats	6
2.3 Uscite Led	6
6.1 Offline Operation	7
6.2 OnLine Operation	8
6.3 Software working cycle	8
6.4 Pushbutton inputs.....	8
6.4.1 Pushbutton input events	8
6.4.2 Lock function	8
6.4.3 State variables (communication objects)	8
6.4.4 Binding between Events and Communication objects	9
6.4.5 Repeated send	9
6.4.6 Input pairs	9
6.4.7 Single or independent input mode	10
6.4.8 Coupled input mode	10
6.4.9 Dimming function.....	11
6.4.10 Shutter / venetian blind function	12
6.5 Outputs for LED signaling.....	14
6.5.1 Individual parameters.....	14
6.5.2 Funzioni logiche	14
7.1 Control algorithms	17
7.2 Two-point control with hysteresis.....	18
7.3 PWM Proportional-Integral control	20
7.4 Setpoint management.....	21
7.5 Operating modes	22
7.6 Heating/cooling switch over	22
7.7 Temperature control alarm.....	23
8.1 Characteristics and timeout.....	23
8.2 Weighted temperature	23
8.3 Surface temperature limitation function	24
8.4 Anticondensation protection function	25
8.5 Window contacts.....	25
8.6 Presence sensors	25
9.0.1 Info su EK-CG2-TP	27
9.0.2 General setting	28
9.0.3 Input configuration.....	28
9.0.4 Independent or single: send values or sequences	30
9.0.5 Independent or single: dimming	30
9.0.6 Independent or single: shutter or venetian blind	31
9.0.7 Independent or single: scene	31
9.0.8 Coupled: switch	32

9.0.9 Coupled: dimming	32
9.0.10 Coupled: shutter or venetian blind	32
9.1 Input x: <i>Function A/B</i> configuration	33
9.1.1 Independent or single.....	33
9.1.2 Independent or single: Lock function enabled	33
9.1.3 Independent or single: send values or sequences	35
9.1.4 Independent or single: dimming	38
9.1.6 Independent or single: scene	39
9.1.7 Coupled.....	40
9.1.8 Coupled: Lock function enabled	40
9.1.9 Coupled: switch	40
9.1.10 Coupled: dimming	41
9.1.11 Coupled: shutter or venetian blind	42
9.2 Sensor temperature	43
9.2.1 Parameters and communication objects	43
9.2.2 Acquisition filter	44
9.2.3 Correction of the measured temperature	44
9.2.4 External sensors (from bus)	45
9.2.5 Parameter and communication object tables	45
9.3 Weighted temperature value	50
9.3.1 Parameter and communication object tables	50
9.4 Temperature control.....	51
9.4.1 Settings	51
9.5 Heating.....	54
9.5.1 Parameter and communication object tables	54
9.6 Cooling.....	59
9.6.1 Parameter and communication object tables	59
9.7 Main and auxiliary ventilation	64
9.7.1 Parameter and communication object tables	64
9.7.2 Delayed fan start (“hot-start”).....	67
9.7.3 Antistratification function	67
9.7.4 2-stage configuration with fan-coils as auxiliary stage	67
9.7.5 Remote fan speed modification.....	68
9.8 Scenes.....	70
9.8.1 Parameter and communication object tables	70
9.9 Relative humidity control	71
9.9.1 Dehumidification	72
9.9.2 Humidification	74
9.10 Energy saving	75
9.10.1 Window contacts.....	75
9.10.2 Presence sensors	75
9.10.3 Card holder	76
10.1 Parameter and communication object tables.....	78

12.1 Summary of KNX communication objects	80
13.1 Other information	92

Revisione	Modifiche	Data
1.0	Emission	04/09/2017
1.1	Minor corrections	12/03/2018
1.2	Notes on value 7F FF sent by sensors when the real value is not available	14/01/2019

1 Scope of document

This application manual describes the application details for the ekinex® contact interface version EK-CG2 –TP ekinex®

Item	File name (## = release)	Version	Device release	Update
Product datasheet	STEKCDG2TP_IT.pdf	EK-CG2-TP	A1.0	09/2017
Application manual	MAEKCG2TP_IT.pdf	EK-CG2-TP		
Application program	APEKCG2TP##.knxprod	EK-CG2-TP		

You can access the most up-to-date version of the full documentation for the device using following QR codes

EK-CG2-TP:



1. Product description

The EK-CG2-TP ekinex® device include four separate digital inputs each configurable as:

- Binary input.
- NTC temperature Probe

This device is equipped with an integrated communication module for KNX bus and is intended for box mounting; DIN rail or coupled to the FF and 71 series

The device also comes with programmable LED outputs for each command, which can be used for signaling functions or as nighttime orientation.

The device is powered by the KNX bus line with a 30 VDC SELV voltage and does not require auxiliary power..

2.1 Input functions

Each one of two active positions of the input, or physical pushbutton, of the device. Such actions, in relation to a single input, will be labelled with letters A and B..

When the input is pressed, the device sends on the KNX bus the telegram (or sequence) associated to the corresponding function according to how the device is programmed..

In the most common situation, for instance, one side of the Input might send an “ON” telegram for a lighting unit, while the other side would send the “OFF” telegram for the same unit. Another typical application would be for one side of the Input to increase the brightness of a dimmed light (and respectively decrease it for the opposite side), or to raise / lower a curtain or blind and so on.

The two functions associated with a Input can also be programmed to perform exactly the same operation, thereby effectively causing one Input to act as a single pushbutton..

2.2 Temperature probes and thermostats

By parameterizing through ETS the input as probe, the device allows to enable and configure up to four thermostats, regardless of the number of inputs that have been configured.

2.3 Uscite Led

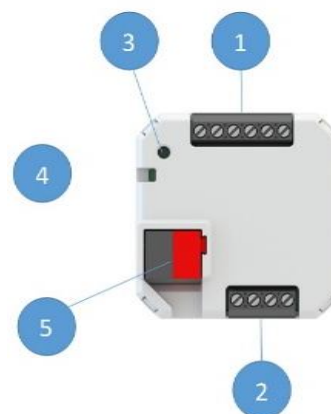
The interface has a number of outputs for the signal LED connection, the number of inputs, which can be freely programmed (also with functions independent of inputs), both as functional indications and for obtaining aesthetic effects or as night orientation lights.

For a more detailed description of the LED outputs and their configuration parameters refer to the application section of the manual.

3 Connection elements

The elements present and necessary for connecting the device are:

1. Digital / NTC inputs terminal block
2. Signal LED link terminal block
3. Aggregate button
4. LED programming mode indication
5. KNX bus line connectio



4 Configuration

The exact functionality of the device depends on the software settings.

In order to configure and commission the device you need ETS4 or later releases and the ekinex® application program.

The application program allows you to access, within the ETS4 / 5 environment, the configuration of all the work parameters of the device.

The program must be loaded into ETS (alternatively you can only load the entire ekinex® database of products in one operation), and then all device specimens of the type considered can be added to the project being drafted.

Configurable parameters for the device will be described in detail in the following paragraphs.

The configuration can be, and will generally be, completely defined in off-line mode; the transfer to the configured device will then occur at the programming stage, described in next paragraph.

Device Code	Input n°	Out Led n°	Applicative Program ETS (## = version)	Cominication Objects (Nr. max)	Group Addresses (Nr. max)
EK-CG2-TP	4	4	APEKCG2TP##.knxprod	365	365

5 Commissioning

After the device has been configured within the ETS project according to user requirements, the commissioning of the device requires the following activities:

electrically connect the device, as described in the product datasheet, to the bus line on the final network or through a purposely setup network for programming;

apply power to the bus;

switch the device operation to programming mode by pressing the programming pushbutton located on the rear side of the housing. In this mode of operation, the programming LED is turned on steady;

upload the configuration (including the physical address) to the device with the ETS program.

At the end of the upload, the operation of the device automatically returns to normal mode; in this mode the programming LED is turned off. Now the device is programmed and ready for use on the bus.

6 Function Description

After switching on the bus, which also acts as a power supply, the device becomes fully functional after a very short time needed for reinitialization. A delay is programmable for the device to become active on the bus in order to avoid a bus traffic overload during the first moments of start-up of the whole network.

In case of a bus power failure (voltage lower than 19 V for 1 s or more), the device becomes unreactive: before the power supply becomes insufficient, the status is internally stored. The timing functions are not active, neither are the programmed group addresses.

As soon as the bus voltage is restored, the device will resume operation in its previous state (which is saved on power fail), unless different initialization settings are programmed

6.1 Offline Operation

A fully unprogrammed device does not operate in standby mode. Since the operation relies entirely on the exchange of information through communication objects, there is no part of the device that can operate independently from a KNX bus.

6.2 OnLine Operation

In general the device works like a configurable digital sensor that is listening to own inputs or outputs of other devices. On input events the device performs output functionality over KNX bus like sending values or controlling external devices like KNX actuators.

6.3 Software working cycle

The main purpose of the software is following:

- Handle user pushbutton presses and generate bus telegrams according to the assigned functions;
- Implement pushbutton interlock and timing functions;
- reagire ai telegrammi sul bus di richiesta dello stato degli ingressi o delle variabili locali.
- Respond to bus messages requesting feedback on the status of the inputs.

The status of the device and specifically of its entities (input activation status) relies on KNX *communication objects*, which can be freely defined and bound in various ways to the physical elements of the device; these communication objects acts as *state variables* for the device.

There are also special events on which it is possible to trigger additional features. These events are the bus failure and recovery, and the download of a new configuration with ETS.

6.4 Pushbutton inputs

The press of a pushbutton can be bound to different effects on a state variable.

6.4.1 Pushbutton input events

A button press can be handled either as an “on-off” event (“on” means when the button is pushed, “off” when it is released), or as a “short press - long press” event (whereby a time period can be defined to discriminate the duration of the “long” from the “short” press).

In both cases, for each of the two available events a separate action can be assigned that operates on a selected variable (actually, more than one; see below for details).

6.4.2 Lock function

For each input (or channel if inputs are coupled, see below), a lock feature can be enabled which allows to block the operation of an input through a message on a communication object.

When in a locked state, the input is effectively disabled.

A value (for each transition) can be specified to be assigned to the communication object upon entering or exiting the locked state.

The locked state can also be automatically activated when the bus is connected.

6.4.3 State variables (communication objects)

The variable that is changed by the input events can be one of the types available for KNX communication objects, i.e. for instance a 1-bit value (on-off), a 2-bit value or an integer value of larger size.

In all cases, each of the two events can:

- change the value of the variable to one of two definable values within its range (which is trivial in the case of the 1-bit value);
- toggle between the two defined values
- do nothing (value is unaffected)

This state variable, once assigned a group address, is actually a **KNX communication object**; as such, it undergoes the usual rules for communication objects, among which – for instance – the effect of flags to determine how the change of value affects the transmission of the objects.

6.4.4 Binding between Events and Communication objects

The above description is a little simplified in order to ease comprehension; as a matter of fact, to each event can be assigned not just one, but several communication objects (up to 8), even of different types. Each of these communication objects can have its own behaviour and its own associated value set.

6.4.5 Repeated send

For most features, is it possible to set the device to send a telegram not just when a value changes as a consequence of an input transition, but also at regular intervals whenever that value setting is active.

This behaviour, also referred to as Cyclical Transmission, can be set separately for each of the two values that are associated to an input (or both, or none of them).

If an input is set to “*send values or sequences*” mode, repeated send is not available if more than 1 Communication Object is assigned to that input.

6.4.6 Input pairs

The 4 inputs described can be considered, and used, as independent; however, due to the physical structure of the device and the nature of the functions it most frequently performs, these inputs can be naturally grouped in pairs, which in the application program are referred to as *channels*. Each channel is made of a pair of inputs, and is physically associated to a Input.

Since the channels of the device are labelled 1 to 4, the inputs are labelled 1A / 1B for channel 1, 2A / 2B for channel 2 and so on. The same numbering is used whether the channel pairing is used or not.

In order to specify channel pairings, each Input can be configured in two ways: single mode and coupled mode. This setting appears among Input-level settings rather than input-level settings, because only inputs belonging to the same Input can be coupled. The only combinations allowed for coupling are in fact 1A with 1B, 2A with 2B, and so on.

- In *single or independent mode*, each input operates independently, has its own parameters and communication objects. This is the mode of operation described so far.
- In *coupled mode*, 2 inputs operate logically grouped under a channel in order to perform a common functionality; therefore, they operate on shared communication objects.

It is possible to configure some of the inputs in *single or independent* mode and the others in *coupled* mode, with the pairing constraints just described.

It must be mentioned that there is actually a third way to configure an input pair, which lies somehow halfway between the two modes above (although it is considered as a variation of the single mode): each second input, i.e. inputs 1B, 2B, 3B etc., can be configured to perform exactly the same function as its first input. In this fashion, both pushbuttons associated with a Input are effectively operated “in parallel”, so as to operate the whole Input as a single, larger control (either pushbutton or switch, according to programmed operation).

Following there is a description of all possible features of the channels. *Single or independent* and *coupled* modes have a similar functionality, but differ for the configuration and will be therefore be treated separately

6.4.7 Single or independent input mode

Each single input can be configured for one of following different features:

1. *Send values or sequences*

An event triggers the transmission on the bus of configurable values or sequence of values.

These values can be of a logical type or a numerical type with a different size.

A sequence of values can be made of up to 8 communication objects of different value types.

Time delays can set between values in the sequence.

2. *Dimmer control*

This mode is intended to be used with dimming actuators for the control of lighting devices.

The functionality is triggered on short press and long press events.

On short press events, the device sends on/off telegrams to the dimming actuator.

On long press events, the dimming percentage is varied up or down until the button is released.

3. *Shutter or Venetian blind control*

This mode is intended to be used together with actuators for the control of motorized blinds, shutters and similar devices. These actuators have functions for blind opening and closing; two movement types are selectable, i.e. continuous movement and stepwise movement. On input events, the device sends operation telegrams to the actuators.

The operation is configurable through following parameters:

- If *toggle* mode is enabled, on each activation of the same input the movement direction is inverted; if it is disabled, the movement direction is fixed and it can be set to “up” or “down”.
- If *blinds* mode is enabled, the device sends “full movement” telegrams on long press and “step” telegrams on short press; if it is disabled, the device sends “full movement” telegrams on long press and “stop” telegrams on short press.

4. *Scene function output*

This mode is intended to be used together with several KNX actuators that support using a scene function; this function allows storing and recalling a communication object value on an actuator.

In this mode, the role of the device is to send a “store / recall scene” telegram to the actuator on a long / short press event.

This mode has two possible configurations:

- Activate pre-set scene on short press, and store current setting as scene value on long press
- Activate two different scenes on long and short press.

6.4.8 Coupled input mode

Each pair of coupled inputs, corresponding to the two sides of a same Input, can be configured for one of following different features (only the differences from the single mode are highlighted):

1. *Switch control*

Both inputs in a pair are bound to the same communication object; unlike single mode, the object can only be of the 1-bit type (on-off), therefore building a conventional switching behaviour.

The user can configure which of the two inputs sets the “off” or resp. “on” value.

2. *Dimmer control*

The functionality is triggered on short press and long press events of the inputs in the pair.

The user can configure which of the two inputs sets the “up” or resp. “down” value.

On short press events, the input configured as “up” sends an “on” switching telegram to the dimming actuator; while the “down” input sends an “off” telegram.

On long press events, the dimming percentage is varied up or down until the button is released.

3. Shutter or Venetian blind control

The two inputs of a pair are assigned to opposite movement directions; these can be assigned to inputs as desired, i.e. A up / B down or the other way around.

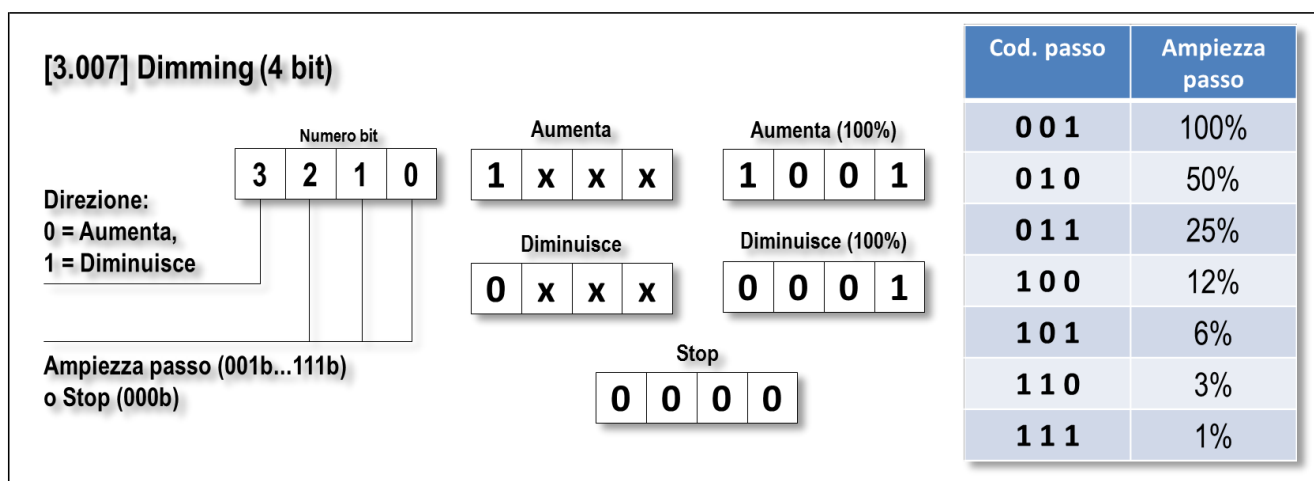
The *blinds* mode can also be set, and it works exactly as in single mode.

In coupled mode, there is no provision for a *scene* control feature

6.4.9 Dimming function

The dimming function is a device application profile included in KNX specifics. Those specifics define the basic requirements for interface mechanisms, in addition to which some aspects regarding the operating modes, peculiar for each device (for both command or actuation devices) are to be considered.

The dimmer control type is essentially based on a 4-bit communication object, whose data has the following format:



The transmission of telegrams containing data of such format tells the actuator to perform an increase or a decrease, by an amplitude equal to the specified step, or to stop an ongoing variation.

The increase or decrease of an intensity value by the actuator is not instantaneous but gradual; therefore, an increase / decrease command with interval equal to the maximum allowed value has the effect of starting the intensity variation in the desired direction, which will continue until the maximum (or minimum) value has been reached. Such variation can be stopped, once the desired intensity value has been reached, by sending a "stop" command.

It is normally possible, and desirable, to have the possibility to instantly switch on or off the load (i.e. to instantaneously bring its value from 0% to 100%). In order to achieve that, an "On / Off" command based on another object is used; this is the same object used for the normal load switch, which is present also in absence of a dimming mechanism.

The command device – in this case, the Input unit – will define the operations to generate a sequence of commands with an opportune order and time interval, in order to achieve the desired command effect.

The defined operations and related commands are the following:

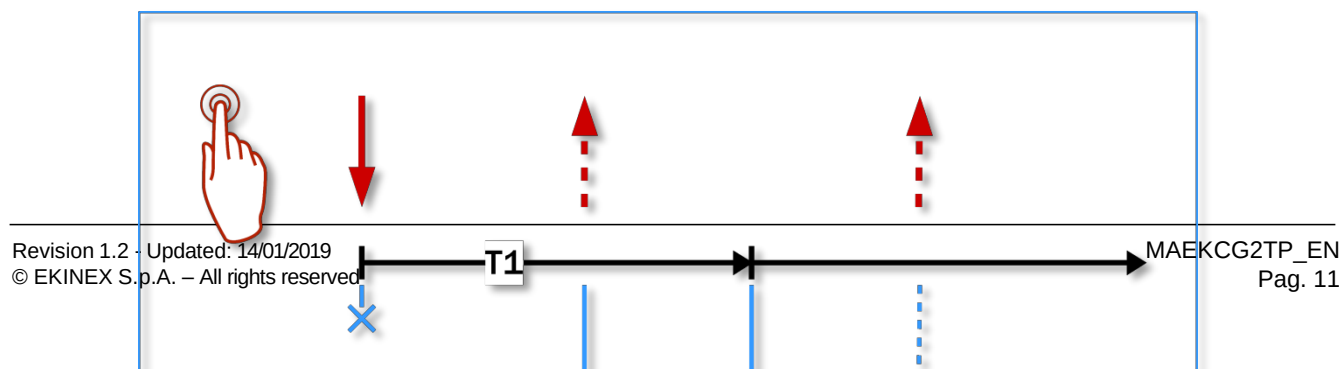


Figure 1 - Dimmer mode command sequence

- Short press: instantaneous switch on / off (toggle on / off on a switch object);
- Long press: increase / decrease value until 100% / 0%;
- Release: stop increase / decrease.

Please note that the same mechanism can be applied to the shutter or venetian blind control (in that case, "maximum / minimum" is substituted with "open / close"). For this purpose, the data type (DPT) 3.008 exists, whose structure and values are identical to those already described; in order to control a shutter with the same mode, it is possible to connect a communication object type 3.007 command side, to an object type 3.008 actuator side (if foreseen). In this case, obviously, the object type "On / Off" which allows instantaneous switch on / off is not used

6.4.10 Shutter / venetian blind function

The "Shutter / venetian blind" function is a bundle of application profiles included in KNX specifics. As for dimming function, such specifics define basic requirements related to interface mechanisms, in addition to which some aspects regarding the operating modes, peculiar for each device (for both command or actuation devices) are to be considered

In case of shutters, the actuator brings a mechanic component from one point to another in a gradual way, with possibility to stop at intermediate points; the command is carried out by 2 lines which, when activated (one line at a time) make the actuator move in the corresponding direction.

A venetian blind is essentially a shutter that, in addition to the up / down movement, is also equipped with slats that can be opened / closed same way as a shutter (gradual movement between extreme points). The peculiarity is that normally the slat's movement and the up / down movement are controlled by the same two lines; therefore, the activation of the electromechanic device must be carried out according to a specific sequence. For further detail please check the actuator's documentation; in this document all we need to point out is that, command side, the control sequences can be considered as independent from these aspects.

The basic control for a shutter or a venetian blind is essentially based on three 1-bit communication objects:

- [1.008] Move Up/Down
- [1.007] Stop – Step Up/Down
- [1.017] Dedicated Stop

The effect of the commands linked to these objects is the following:

- The command "Move", when received, starts the movement of the shutter in the indicated direction.
- The command "Stop – Step" has two functions: if the shutter is stopped, it moves by one step in the indicated direction (the duration is set in the actuator), if not, it stops the ongoing movement without doing anything else.
- The command "Stop" just stops the ongoing movement.

In addition, other types of control objects are normally available (“dimmer” type, absolute position, etc.) but they are not part of the basic control on which this manual is about; for further information please refer to the actuators’ manual or KNX specifics.

In the simplest version, on command side:

- In order to control a shutter at least the objects “Move” and “Stop” are required (and present).
- In order to control a venetian blind at least the objects “Move” and “Stop – Step” are required (and present).

On actuator side – whether it is a shutter or a venetian blind – the presence of objects “Move” and “Stop – Step” must be guaranteed, while the presence of the object “Stop” is optional (but usually present).

As for the operations to perform on the command device, in our specific case the Input unit, in order to generate a sequence of these commands with the proper order and time interval, there are multiple possibilities.

In case of ekinex input devices, two modes are available – indicated as “Shutter” and “Venetian blind” based on their typical destination – which are illustrated in the following figure.

Figure 2 - “Shutter” mode command sequence

In “Shutter” mode, when a Input is pressed – or a digital input is activated – the shutter starts moving in the corresponding direction (which can be alternatively in the two directions if the Input is in independent mode and has been configured as *toggle*).

If the Input is released quickly, the shutter will continue its run until full opening or closing; it is still possible to stop it by pressing again the Input with a long press.

If the Input is pressed with a long press, when it is released – which will be in correspondence with the desired position – the shutter will stop.

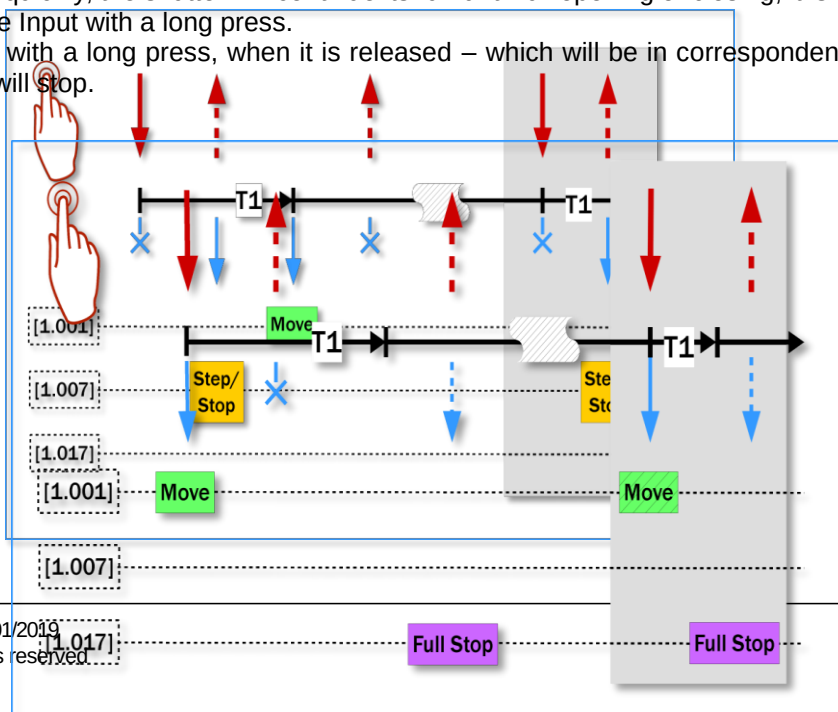


Figure 3 - “Venetian blind” mode command sequence

In “Venetian blind” mode, on release of a Input after a short press, the venetian blind performs a step; this operation, normally – i.e. even if the actuator is indeed configured for a venetian blind – is used for the slats regulation.

If the Input is pressed with a long press, when the threshold time is reached, a “Move” command is issued, which will bring the venetian blind to full open or close. In order to stop it at an intermediate position, the Input needs to be pressed again (short press).

6.5 Outputs for LED signaling

The LED indicators associated with each input can be individually addressed even if the corresponding inputs are paired.

6.5.1 Individual parameters.

The power of each LED can be set as follows:

- Fixed value (always on or off)
- Switches on when the corresponding input is activated. With this option, you can specify an additional delay after the button is released;
- Status determined by the bus through via communication object. In this case, you can specify that in the active state the LED is flashing (with different choices for on / off times); In addition, the on / off condition can be reversed with respect to the status of the reference communication object (LED lit when the value of the object is "off" and vice versa).

6.5.2 Funzioni logiche

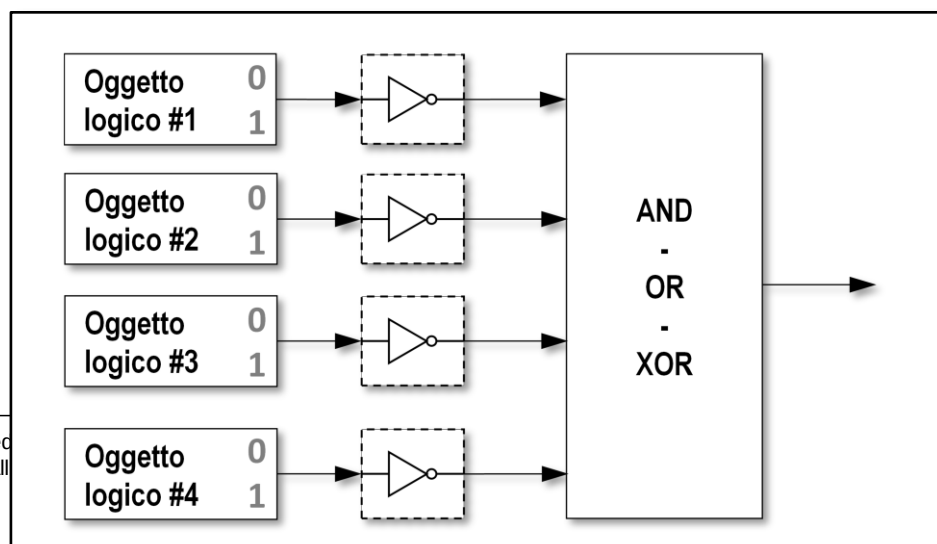
The KNX pushbutton allows to use some useful logic functions (AND, OR, NOT and exclusive OR) in order to implement complex functions in the building automation system.

You can configure:

- 4 channels of logical functions
- 4 inputs for each channel

Each object value, if desired, can be individually inverted by inserting a NOT logic operator.

The inputs created by the objects are then logically combined as shown in the following figure:



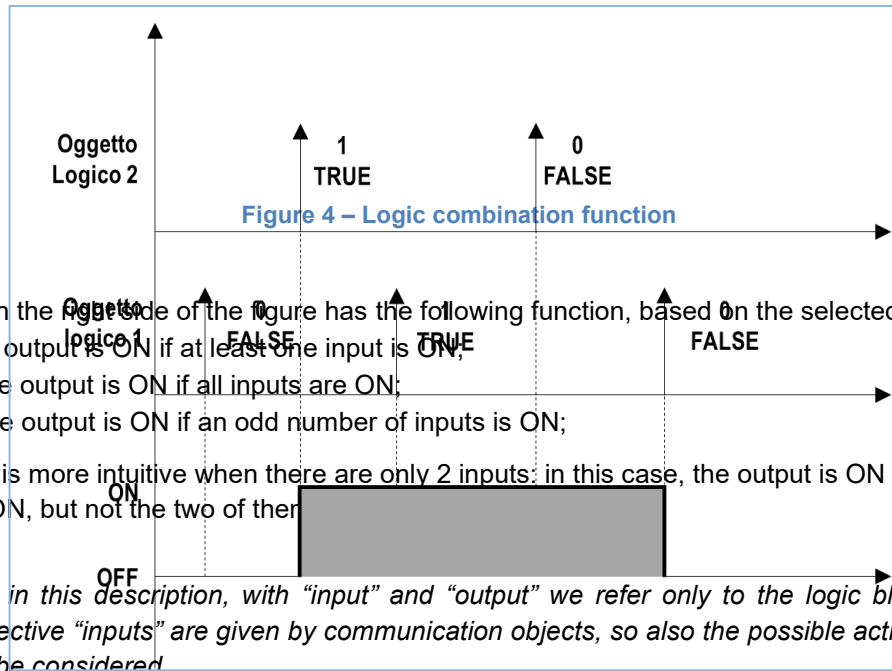


Figure 4 – Logic combination function

The logic block on the right side of the figure has the following function, based on the selected operation:

- OR – the output is ON if at least one input is ON;
- AND – the output is ON if all inputs are ON;
- XOR – the output is ON if an odd number of inputs is ON;

This last function is more intuitive when there are only 2 inputs: in this case, the output is ON when one input or the other one is ON, but not the two of them.

Please note that in this description, with “input” and “output” we refer only to the logic block; for the device operation, the effective “inputs” are given by communication objects, so also the possible activation of NOT logic operators has to be considered.

The following figures show the basic logic functions, assuming 2 inputs and only one logic communication object:

Figure 5 – Logic function OR

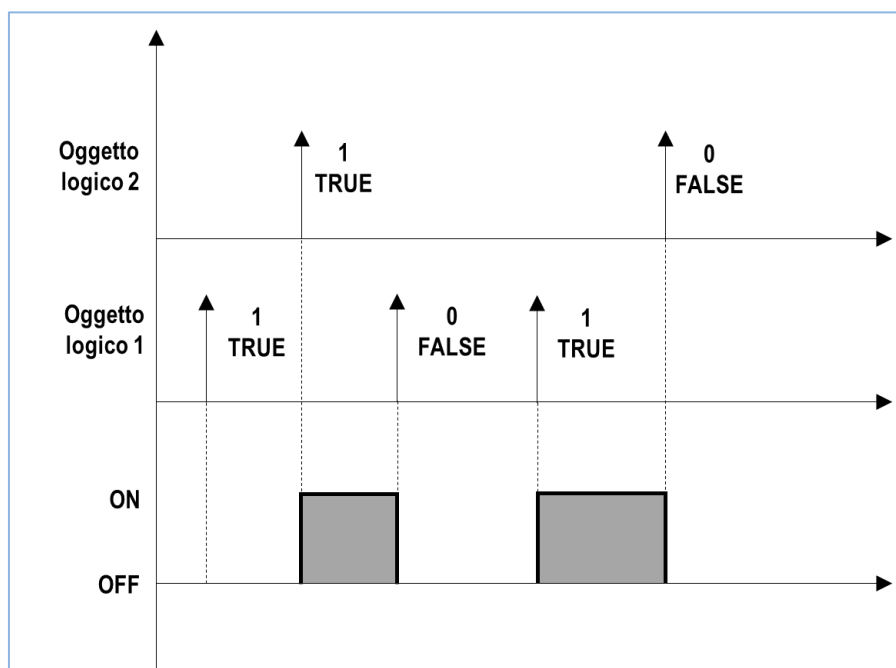


Figure 7 – Logic function AND

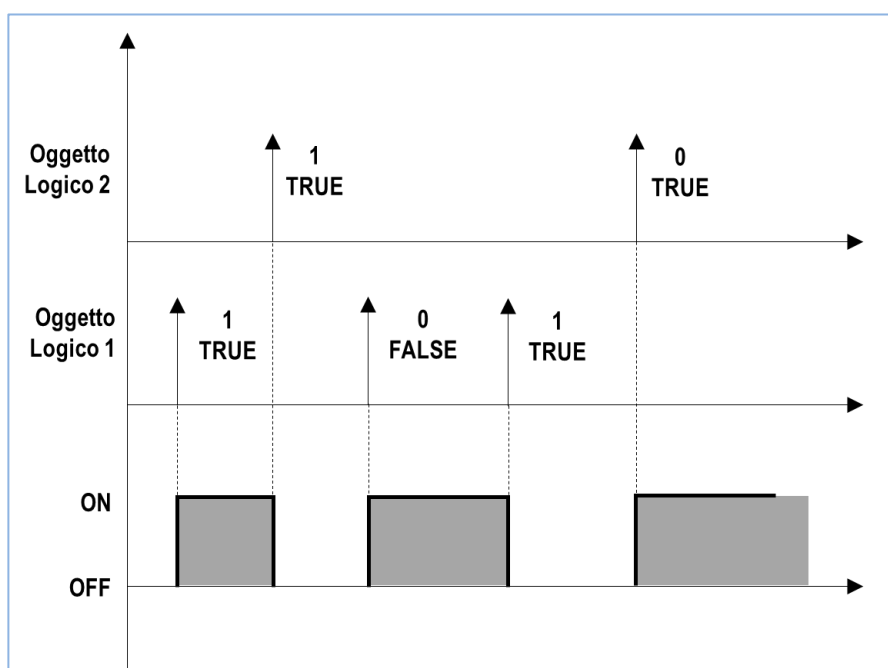


Figure 8 – Logic function XOR

For each channel, a parameter *Delay after bus voltage recovery* is available: this parameter represents the time interval between the bus voltage recovery and the first reading of the input communication objects for evaluating the logic functions.

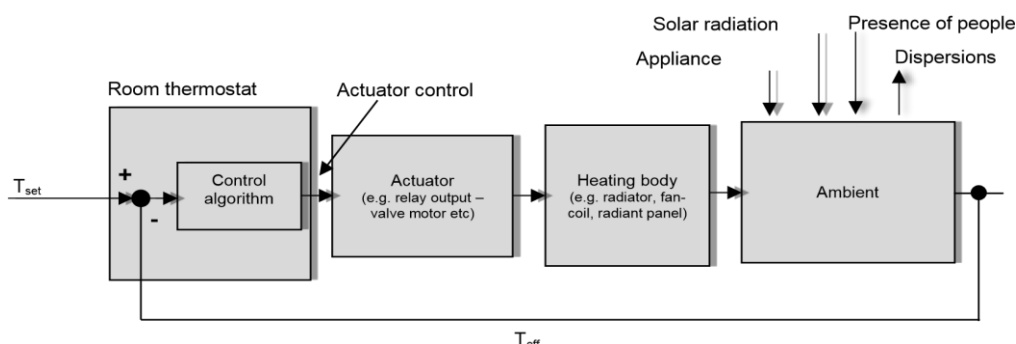
The communication function representing the logic function output is sent on the bus on event of change; alternatively, a cyclic sending can be set.

7 Room Controller

The room air temperature control is performed thanks to the intercept valve(s) on the heat exchange coil(s), with an ON/OFF or PWM control algorithm. In order to control those valves, electrothermal actuators and/or servomotors can be used

7.1 Control algorithms

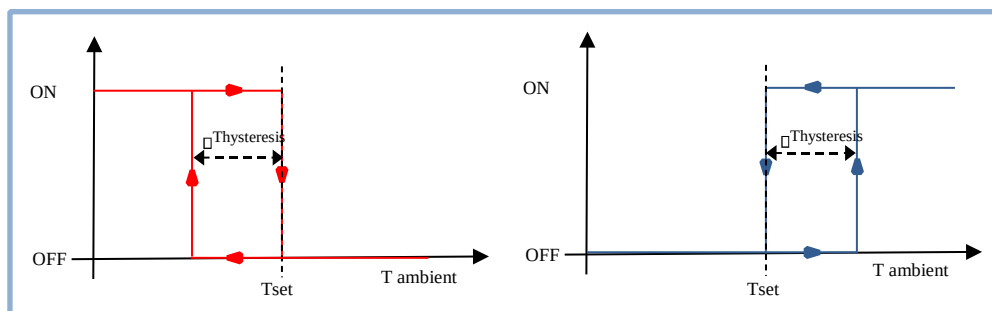
The picture below shows the components of a common generic control system for ambient temperature. The room thermostat measures the actual temperature of the air mass (T_{eff}) and constantly compares it to the setpoint value (T_{set}).



The control algorithm, basing on the difference between T_{set} and T_{eff} , processes a command value which can be of analogue or On / Off type; the command is represented by a CO that is transmitted via bus, periodically or event based, to a actuator device. The output of the actuator device is the driving variable of the control system, which can be e.g. a flow rate of water or air. The control system realized by the room thermostat is of feedback type, namely the algorithm takes into account the effects on the system in order to change the control action on the same entity.

7.2 Two-point control with hysteresis

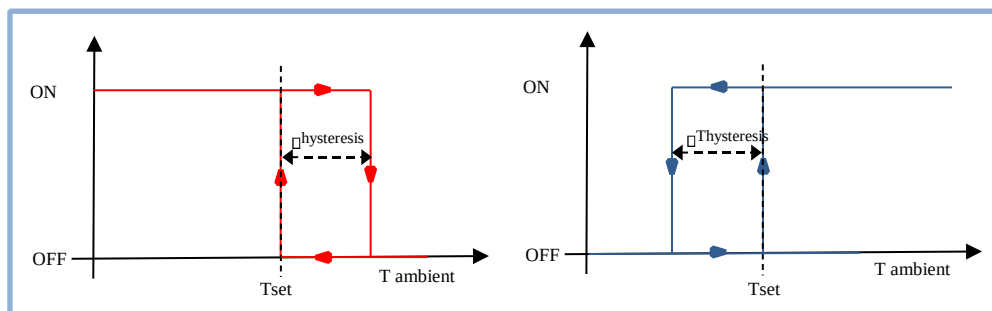
This control algorithm, which is also known as On / Off, is the most classic and popular. The control provides for the on / off switching of the system following a hysteresis loop, i.e. two threshold levels are considered for the switching instead of a single one.



Heating mode: when the measured temperature is lower than the value of the difference ($T_{\text{set}} - \Delta T_{\text{hysteresis}}$), whereby $\Delta T_{\text{hysteresis}}$ identifies the differential adjustment of the boilers, the device activates the heating system by sending a message or KNX telegram to the actuator that handles the heating system; when the measured temperature reaches the desired temperature (Setpoint), the device disables the heating system by sending another message. In this way, there are two decision thresholds for activation and deactivation of the heating, the first being the level ($T_{\text{set}} - \Delta T_{\text{hysteresis}}$) below which the device activates the system, whereas the second is the desired temperature above which the heating system is deactivated.

Cooling mode: When the measured temperature is higher than the value of the difference ($T_{\text{set}} + \Delta T_{\text{hysteresis}}$), whereby $\Delta T_{\text{hysteresis}}$ identifies the differential adjustment of the cooler, the device activates the air conditioning system by sending a message or KNX telegram to the actuator that handles it; when the measured temperature falls below the desired temperature T_{set} the device turns off the air conditioning system by sending another message. In this way, there are two decision thresholds for activation and deactivation of the cooling: the first being the level ($T_{\text{set}} + \Delta T_{\text{hysteresis}}$) above which the device activates the system, whereas the second is the desired temperature below which the air conditioning system is deactivated. In the ETS application program, two different parameters are available for the hysteresis value for heating and cooling: the values usually differ depending on the system type and its inertia.

In those applications where floor or ceiling radiant panels are present, it is possible to realize a different 2-point room temperature control. This type of control must be paired either to a proper regulation system for flow temperature that takes into account all internal conditions or an optimizer that exploits the thermal capacity of the building to adjust the energy contributions. In this type of control the hysteresis ($\Delta T_{\text{hysteresis}}$) of the room temperature high limit ($T_{\text{set}} + \Delta T_{\text{hysteresis}}$) represent the maximum level of deviation that the user is willing to accept during plant conduction.



Heating mode – When the measured temperature is lower than the desired temperature T_{set} , the device activates the heating system by sending a message or KNX telegram to the actuator that handles it; when the measured temperature reaches the value $(T_{set} + \Delta T_{hysteresis})$, whereby $\Delta T_{hysteresis}$ identifies the differential adjustment of the boilers the device disables the heating system by sending another message. In this way, there are two decision thresholds for activation and deactivation of the heating, the first being the desired temperature T_{set} below which the device activates the system, whereas the second is the value $(T_{set} + \Delta T_{hysteresis})$, above which the heating system is deactivated.

Cooling mode – When the measured temperature is higher than the desired temperature T_{set} , the device activates the air conditioning system by sending a message or KNX telegram to the actuator that handles it; when the measured temperature reaches the value $(T_{set} - \Delta T_{hysteresis})$, whereby $\Delta T_{hysteresis}$ identifies the differential adjustment of the air conditioning system, the device disables the air conditioning system by sending another message. In this way, there are two decision thresholds for activation and deactivation of the air conditioning system: the first being the desired temperature T_{set} above which the device activates the system, whereas the second is the value $(T_{set} - \Delta T_{hysteresis})$ below which the air conditioning system is deactivated.

In the ETS application program, two different parameters are available for the hysteresis value for heating and cooling: the values usually differ depending on the system type and its inertia.

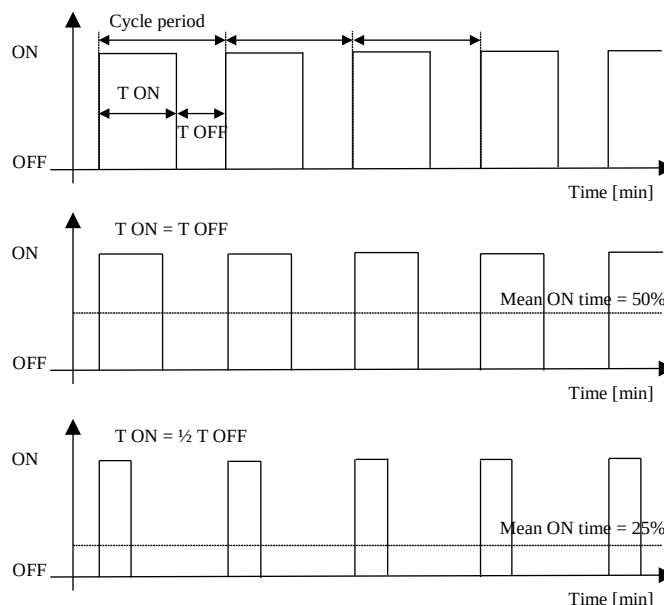
In the ETS application program, the default 2-point hysteresis control algorithm foresees inferior hysteresis for heating and superior for cooling. If Heating and/or cooling type = floor radiant panels or ceiling radiant panels, it is possible to select the hysteresis position according to the described second mode, i.e. with superior hysteresis for heating and inferior for cooling.

The desired temperature (T_{set}) is generally different for each one of the 4 operating modes and for heating/cooling modes. The different values are defined for the first time during ETS configuration and can be modified later on. In order to optimize energy saving (for each extra degree of room temperature, outbound dispersions and energy consumption go up 6%), it is possible to take advantage of the multifunctionality of the domotic system, for example with:

- Hour programming with automatic commutation of the operating mode by means of KNX supervisor;
- Automatic commutation of the operating mode according to window opening for air refreshment;
- Circuit deactivation when desired temperature is reached;
- Flow temperature reduction in case of partial load.

7.3 PWM Proportional-Integral control

The proportional-integral PWM (Pulse Width Modulator) controller uses an analogue control variable to modulate the duration of the time intervals in which a binary output is in the On or Off state. The controller operates in a periodic manner over a cycle, and in each period it maintains the output to the On value for a time proportional to the value of the control variable. As shown in the figure, by varying the ratio between the ON time and the OFF time, the average time of activation of the output varies, and consequently the average intake of heating or cooling power supplied to the environment



This type of controller is well suited for use with On / Off type actuators, such as relays and actuators for zone valves, which are less expensive (both for electrical and mechanical components) than proportional actuators. A distinctive advantage of this type of controller, compared with the raw On / Off controller already described, is that it eliminates the inertia characteristics of the system: it allows significant energy savings, because you avoid unnecessary interventions on the system introduced by the 2-point control with hysteresis and it only provides the power required to compensate for losses in the building.

Every time the user or the supervisor changes the desired temperature setpoint, the cycle time is interrupted, the control output is reprocessed and the PWM restarts with a new cycle: this allows the system to reach its steady state more quickly.

Terminal type	Proportional Band [K]	Integral Time [min]	Cycle Period [min]
Radiators	5	150	15-20
Electrical heaters	4	100	15-20
Fan-coil	4	90	15-20
Floor radiant panels	5	240	15-20

Guidelines for choosing the proper parameters of a PMW Proportional-Integral controller:

- Cycle time: for low-inertial systems such as heating and air conditioning systems, short cycle times must be chosen (10-15 minutes) to avoid oscillations of the room temperature.
- Narrow proportional band: wide and continuous oscillations of the room temperature, short setpoint settling time.
- Wide proportional band: small or no oscillations of the room temperature, long setpoint settling time.
- Short integral time: short setpoint settling time, continuous oscillations of the room temperature.
- Long integral time: long setpoint settling time, no oscillations of the room temperature.



7.4 Setpoint management

The device is not equipped with a local interface to control the integrated room thermostat, therefore the temperature setpoint modifications need to be performed by another KNX device (supervisor) and sent to this device through communication objects.

Three setpoint management modes are foreseen:

- ☐ Single setpoint; ☐
- Relative setpoints; ☐
- Absolute setpoints.

Single setpoint mode

In this mode, a unique communication object is exposed (*Input Setpoint*) to modify the desired temperature. This object can be updated cyclically or on event of change by the supervisory device. If power goes down, the last value is retained into the pushbutton's non-volatile memory. In case the object is not updated, the temperature controller acts anyway on default setpoints (both heating and cooling) set in the application program during commissioning.

If window contacts for energy saving are used, when detecting an open window the input setpoint freezes and the pre-set building protection setpoint is activated (the relative communication object is exposed and is different in heating or cooling mode).

Relative setpoints mode

In this mode, 4 communication objects are exposed, one for each operating mode:

- Comfort setpoint
- Stand-by offset
- Economy offset
- Building protection setpoint

Stand-by and economy setpoints are represented as attenuations to the comfort setpoint in order to facilitate the supervisor management: by uniquely modifying the comfort setpoint, references for attenuated modes are automatically transferred. The values modified from bus are retained in the pushbutton's non-volatile memory.

With this mode, the supervisory device can develop an hour-based time scheduling by sending to the device the current operating mode (comm. obj. *HVAC mode in* [20.102] DPT_HVAC Mode). The default value for *HVAC mode in* corresponds to the comfort setpoint value.

Same as single setpoint management, if the temperature controller is set as both heating and cooling mode with switch over from bus, it is necessary that the supervisory device also updates the input seasonal mode object (*Heating/cooling status in*, [1.100] DPT_Heat_Cool) in order to coherently switch over the controller's action.

Absolute Setpoint mode

In this mode, 3 communication objects are exposed, for each conduction mode:

- Comfort setpoint;
- Standby setpoint;
- Economy setpoint;
- Building protection setpoint.

All setpoint are absolute values: by modifying those values from bus through communication objects you need to keep the coherence among the values of the attenuated operating modes.

With this mode, the supervisory device can develop an hour-based time scheduling by sending to the device the current operating mode (comm. obj. *HVAC mode in* [20.102] DPT_HVAC Mode). The default value for *HVAC mode in* corresponds to the comfort setpoint value.

Same as single setpoint management, if the temperature controller is set as both heating and cooling mode with switch over from bus, it is necessary that the supervisory device also updates the input seasonal mode object (*Heating/cooling status in*, [1.100] DPT_Heat_Cool) in order to coherently switch over the controller's action.

7.5 Operating modes

In Single Setpoint mode, 2 levels for each conduction mode are available:

- Temperature setpoint
- Building protection setpoint

Time scheduling for attenuation can be configured from the supervisor, by directly modifying the temperature setpoint.

In Relative and Absolute Setpoint mode, 4 different operating modes are available, which are mutually exclusive to one another:

- comfort;
- stand-by;
- economy;
- building protection.

Through ETS application program, it is possible to assign 2 different setpoint values to each operating mode, for comfort and building protection level, and two different attenuation levels for stand-by and economy, corresponding to both heating and cooling.

Each setpoint is exposed through communication objects. Setpoints and attenuations can be modified remotely through the exposed communication objects. The setpoint intervention for building protection must be planned in ETS application program, as these parameters concern the safety and protection of the plant's components (especially during heating).

7.6 Heating/cooling switch over

The switch over between both heating and cooling mode can take place in 3 ways:

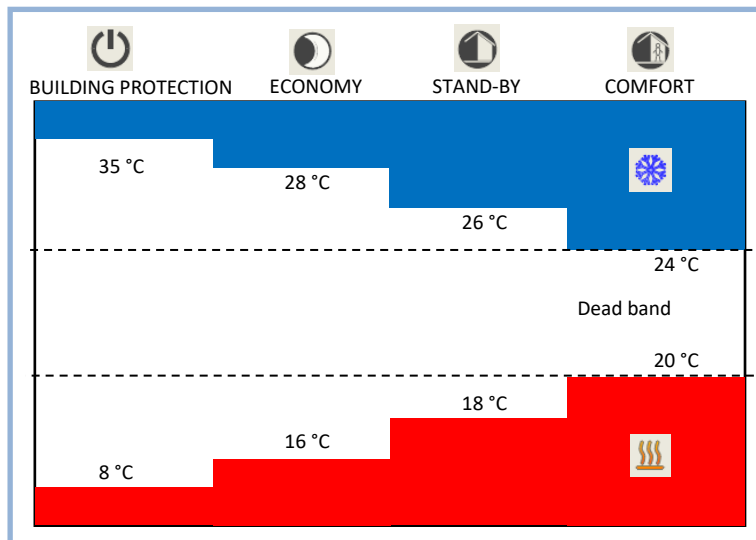
1. from KNX bus, through a communication object;
2. automatically, based on the room temperature.

Switchover from bus

In mode 1, the switch over command is issued through KNX bus and therefore it is performed by a different KNX device, e.g. the ekinex® Touch&See unit. The integrated temperature controller acts as a "slave": the switch over is carried out by input communication object [DPT 1.100 heat/cool].

Automatical switch over, based on the room temperature

Mode 2 is suitable for applications with heating / cooling systems with a 4-pipe. In addition, the information can be transmitted on the bus through an output communication object [DPT 1.100 heat/cool]; the difference with mode 1 is that the switch over is performed automatically by the machine, basing on the values of current temperature and setpoint. The automatic switch over is achieved by introducing a dead band as shown in the following figure.



The figure shows that, as long as the actual measured temperature is below the heating mode setpoint, the heating mode is selected; similarly, if the value is greater than the cooling setpoint, then cooling mode is selected. If the value is within the dead band, the operation mode remains unchanged; the heating/cooling switchover point must correspond to the actual setpoint of the current HVAC mode, and in the same way the cooling/heating switchover must correspond to the actual heating setpoint.

An automatic switchover can be performed by interlocking the valves based on the time needed by the actuator to make a complete run. The switchover sequence includes the full closing of all intercept valves of, for example, the hot fluid, before starting to open the valves of the cool fluid. This sequence can be realized with both electrothermal actuators and zone valve drives.

7.7 Temperature control alarm

The integrated temperature controller can stop the internal control algorithm for one of the following reasons:

- For an external event, which can be configured and linked to the *Thermal generator lock* communication object;
- For an internal temperature sensor's fault (measured room temperature too low while NTC resistance value is too high or vice versa);
- For a timeout (data not updated by the bus) when a weighted mean between the internal sensor's value and an auxiliary external sensor's value is used.

When one of these events occur, the internal controller stops the control algorithm and the command output is taken to complete closing position (OFF or 0%): this state is indicated through the communication object *Room temperature control alarm*.

Input from bus

8.1 Characteristics and timeout

When using the device with integrated temperature controller, variables acquired from the bus are available, which are different for each channel. All bus inputs allow to extend the device's functionality.

8.2 Weighted temperature

The device allows the acquisition of the room temperature in 2 ways:

- 1) from an external temperature sensor connected to a device input configured as analogic (Inputs □ Input 1 or 2 = [AI] room temperature sensor);
- 2) via bus from another KNX device, e.g. from an ekinex pushbutton (External sensors (from bus) □ Room temperature = enabled);

To optimize or correct the room temperature regulation in special cases (in large rooms, in presence of strong asymmetry of the temperature distribution, when the installation of the device is in a position not suitable, etc.), the device can then use a weighted average between two temperature values. The weights are assigned by the parameter *Relative weight* that assigns a ratio of the two values.

Note: the value for “Weighted Temperature” communication object is set to 7F FF in case of the real value cannot be read from bus.

8.3 Surface temperature limitation function

The floor heating system (warm water version) provides plastic pipes embedded in the concrete layer or placed directly under the final coating of the floor (light or “dry” system) filled by heated water. The water releases heat to the final coating that heats the room by radiation. The standard EN 1264 Floor heating (Part 3: Systems and components - Dimensioning) prescribes a maximum allowed temperature (T_{Smax}) for the surface of the floor that is physiologically correct defined as:

- $T_{Smax} \leq 29^{\circ}\text{C}$ for zones of normal occupancy;
- $T_{Smax} \leq 35^{\circ}\text{C}$ for peripheral zones of the rooms.

National standards may also limit these temperatures at lower values. Peripheral zones are strips generally located along the external walls with a maximum width of 1 m.

The floor heating system (electrically powered version) involves the laying under the floor coating of an electric cable powered by the mains voltage (230 V) or low voltage (for example 12 or 45 V), possibly already prepared in the form of rolls with constant distance between sections of cable. The powered cable releases heat to the overlying coating that heats the room by radiation. The regulation is based on measurement of the temperature of the air mass, but generally requires the monitoring and limiting of the surface temperature by using a NTCtype sensor which is in contact with the floor surface.

The surface temperature limitation may be realized for several purposes:

- physiological compatibility (correct temperature at the height of the legs);
- when the system is used as auxiliary stage for heating. In this case, the heat losses to the exterior of the building are handled by the main heating stage, while the auxiliary stage only works to keep the floor temperature at a comfortable level (for example in bathrooms of residential buildings, sports centers, spas and thermal baths, etc.);
- protection against damages of the final coating due to an accidental overheating. Note that the warm water radiant panels are usually already equipped with a safety thermostat (with intervention on the hydraulic mixing group), while in the case of electrical power this device is not usable and it is common practice to realize a temperature limitation with a surface temperature sensor connected to the device.

The surface temperature limitation function closes the intercept valves on the distribution manifold when the temperature measured on the panel rises above set threshold (default value 29°C). The regular operation of the room thermostat resumes when the measured surface temperature drops below the hysteresis threshold ($29^{\circ}\text{C} - 0,3\text{ K}$). For related alarms please refer to the Appendix.

8.4 Anticondensation protection function

The objective of this function is to prevent the condensation on the thermal exchange surfaces of the installation or building when cooling is working. This function is mainly used in systems with thermal exchange consisting in surface terminals such as for the floor and ceiling cooling radiant systems. In this case the hydraulic circuits contain refrigerated water; usually the latent loads (due to the increase of air humidity in the room) are handled by air-conditioning units and the temperature and humidity conditions are far from those that could cause condensation. If this is not done in a satisfactory manner, or in case of stop of the airconditioning units, it is necessary to provide additional safety measures to prevent or restrict the accidental formation of condensation on cold surfaces.

The alarm contact must be connected to an input channel of another KNX device, for example a pushbutton interface or a binary input. In this case the signal coming from the probe is sent to the channel of the device via bus, through a communication object.

In case of anticondensation alarm, if the temperature controller is in cooling mode and is demanding for fluid, the intercept valve is closed. It automatically comes back to normal mode as soon as the sensor returns to normal operation. For related alarms please refer to the Appendix.

8.5 Window contacts

In order to realize energy-saving functions, window contacts (to detect the opening of windows or doors) can be used. The device can acquire the status of a contact by means of a digital input or receive the status of two contacts connected to different KNX devices (binary inputs, pushbutton interfaces). When a window opens, the device automatically switches to *Building Protection* operating mode; when it closes, the device automatically returns to the previous operating mode. When acquiring two signals, they can be combined in logical OR.

The window contact management is an optional feature, oriented to energy saving, which is available only when the device is configured as integrated temperature controller. When an open window is detected, the operating mode is forced into building protection and remains forced until all windows are closed. The application program features a time parameter for opening delay to discriminate between an occasional, short opening and a long opening, which justifies the energy saving mode recall.

The window contact management has absolute priority over the operating mode forced by time scheduling, over the mode forced by presence sensors (if enabled) and over the HVAC mode forced by supervisor through the communication object *HVAC Forced mode in DPT 20.102*.

8.6 Presence sensors

Presence sensors management includes a set of optional features, oriented to energy saving, which become available when the device is configured as integrated controller.

Generally speaking, if a human presence is detected and limited to the occupancy period, the comfort operating mode can be extended; vice versa, if no presence is detected, the comfort operating mode can be limited, because no longer necessary.

The occupancy status detection is performed by presence sensors which can be connected to KNX devices equipped with binary inputs; the device exposes two 1-bit communication objects for each one of the 4 channels; these objects are then synchronized to the situations detected by the sensors.

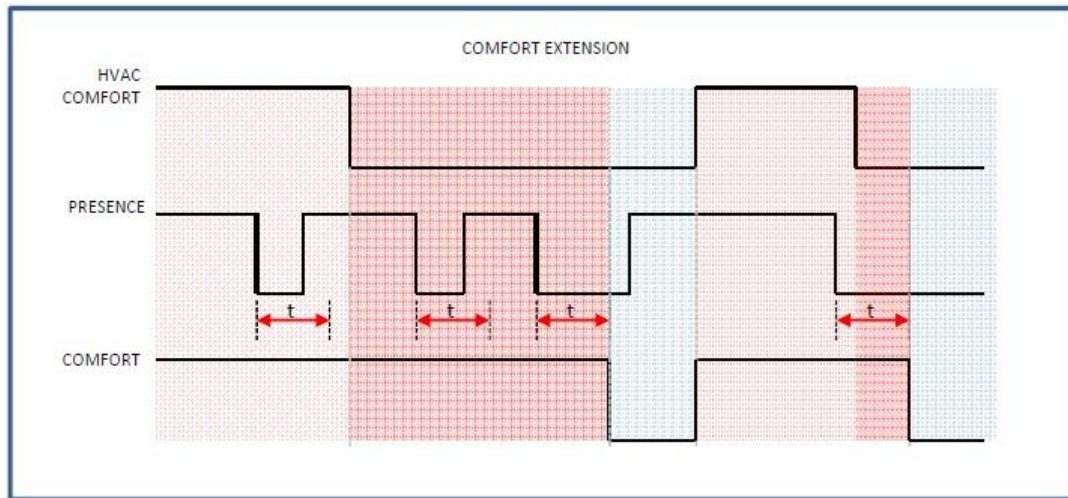
In order to determine which physical state corresponds to the presenc state, two different options can be selected:

- Not inverted (normally closed): an open contact corresponds to non-occupancy state, a close contact corresponds to detected presence;
- Inverted (normally open): an open contact corresponds to detected presence, a close contact corresponds to non-occupancy state;

There are three presence state management modes: comfort extension, comfort limitation and a combination of these two modes.

Comfort extension. This function is only active if the actual operating mode is set on comfort; if, during this time, a presence is detected, the operating mode remains comfort even if the operating mode forced by the time

scheduling function shifts to economy or standby. If a presence is not detected for a timeframe less than a preset time, the operating mode does not change; vice versa, if a presence is not detected for a time period greater than the same preset time, the operating mode becomes the one forced by the time scheduling function.

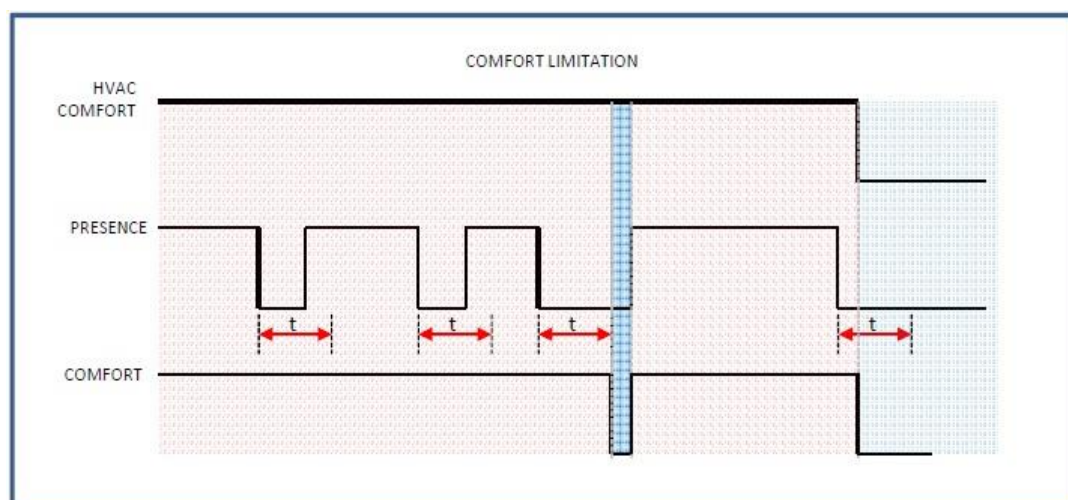


The figure above shows that, even if a presence is detected while the operating mode forced by the time scheduling function is not comfort, no change of operating mode is performed until the next programmed comfort event.

In case a forced HVAC mode is used by a supervisor through the communication object *HVAC forced mode in* DPT 20.102, the forced operating mode has a higher priority compared to the mode foreseen by the presence management, so it will prevail.

In case the energy saving management is carried out through window contacts, the latter has a higher priority compared to both the forced mode and the mode foreseen by the presence management; whatever operating mode is forced by the time scheduling function, by presence management or by forced mode, the system switches to building protection mode when detecting an open window.

Comfort limitation. This function is only active if the actual operating mode is set on comfort; if, during this time, a presence is not detected for a time period greater than a preset time, the operating mode shifts to economy or standby. The attenuation modes can be selected in the application program and are independent from the modes foreseen by the time scheduling function.



Same as comfort extension, in case a forced HVAC mode is used by a supervisor through the communication object *HVAC forced mode* in DPT 20.102, the forced operating mode has a higher priority compared to the mode foreseen by the presence management, so it will prevail.

In case the energy saving management is carried out through window contacts, the latter has a higher priority compared to both the forced mode and the mode foreseen by the presence management; whatever operating mode is forced by the time scheduling function, by presence management or by forced mode, the system switches to building protection mode when detecting an open window.

Comfort extension and comfort limitation. This mode is a combination of comfort extension and comfort limitation modes.

Application program for ETS

In the following chapters, there is the list of folder, parameters and communication objects of the application program.

Every channel, and every input or input pair under a channel, offers the same set of communication objects and parameters, but they may all be independently configured.

Hereafter, all channel-specific settings are listed grouped by channel; a generic channel number is referenced as "x" (where x = 1...2), while a generic input is referenced as "xx" (xx = 1A, 1B, 2A, 2B).

i

I valori dei parametri evidenziati in neretto sono quelli di default.

1.1.1 Interfaccia universale 4 DIN / NTC > About EK-CG2-TP

About EK-CG2-TP

General

Channels configuration

Leds

— Channel 1

Input 1

Input 2

+ Channel 2

— Logic functions

Logic function 1

Logic function 2

Logic function 3

Logic function 4

© Copyright SBS S.p.A. 2017
Application software for ETS4 & ETS5
Version 0.05
EK-CG2-TP -

SBS S.p.A.
Via Circonvallazione s/n
I-28010 Miasino (NO)
www.ekinex.com
info@ekinex.com

9.0.1 Info su EK-CG2-TP

The folder **About EK-CG2-TP** is for information purposes only and does not contain parameters to be set. The information given is:

© Copyright EKINEX S.p.A. 2017
Application software per ETS4/5
Version 1.00
Universal Interface 4 DIN / NTC

EKINEX S.p.A.
Via Circonvallazione s/n
I-28010 Miasino (NO) Italy www.ekinex.com info@ekinex.com

9.0.2 General setting

The parameters in this section define the overall behaviour of the device, including the setting that defines which and how many channels are available

Parameter name	Conditions	Values
Delay after bus voltage recovery	-	hh:mm:ss.fff (00:00:04.000)
	Delay before bus telegrams can be sent after a recovery of the bus voltage. The delay time affects the transmission generated by an event as well as the cyclical transmission. For the cyclical transmission: after the delay time finished, the cycle restarts and the first telegram will be sent after the cycle time.	
Logic functions		enabled / disabled
	Enables the folders to configure AND, OR e XOR logic functions and their relative input and output communication objects.	

9.0.3 Input configuration

Nome parametro	Condizioni	Valori
Input x	-	disabled independent or single coupled temperature probe
	Sets the operation mode for inputs associated with Input x. The identification of which inputs and physical inputs corresponding to a given number (eg 1A - 2B - 4A etc.), depending on the layout chosen for the plates, depends on the parameter "General / Input Configuration	
Function A	Input x = independent or single	enabled / disabled
	Enables or disables the capability to generate events for the first pushbutton of the input.	
Type	Input x = independent or single Function A = enabled	send values or sequences dimming shutter or venetian blind scene
	Determines the kind of function performed by the FIRST Input input. Further parameters for the selected function will appear in the individual Input configuration sections (see below).	
Temperature probe A	Input x = temperature probe Function A = enabled	enabled / disabled
	Determines the kind of function performed by the FIRST Input input. Further parameters for the selected function will appear in the individual Input configuration sections (see below).	
Function B	Input x = independent or single	disabled enable in parallel with function A, as a function. single copy parameters from function A
	Enables or disables the ability to generate events for the second Input button. If it is not disabled, the second button can be assigned its own independent function, it can be used as a "duplicate" of the first input (in parallel ...) or perform the same type of function as the first button but based on its own independent communication objects .	

Nome parametro	Condizioni	Valori
Type	Input x = independent or single Function B = enabled	send values or sequences dimming shutter or venetian blind scene
	<i>Determines the kind of function performed by the FIRST Input input. Further parameters for the selected function will appear in the individual Input configuration sections (see below).</i>	
Type	Ingresso x = accoppiato	switch dimming blinds or Venetian blinds
	<i>Determines the kind of function performed by the FIRST and SECOND Input input. Further parameters for the selected function will appear in the individual Input configuration sections (see below).</i>	
Temperature probe B	Input x = temperature probe Function B = enabled	enabled / disabled
	<i>Determines the kind of function performed by the FIRST Input input. Further parameters for the selected function will appear in the individual Input configuration sections (see below).</i>	

9.0.4 Independent or single: send values or sequences

Object name	Conditions	Size	Flags	DPT	No. Comm. Obj.
Input x – Switching status [type], object n	Input x = independent or single Function x = enabled Type = send values or sequences	according to the configuration (1-bit)	CRWTU	according to the configuration ([1.001] switch)	5, 22 (1A, 1B) 46, 60 (2A, 2B)
The size and type of the individual objects can be configured as described in following sections.					

9.0.5 Independent or single: dimming

Object name	Conditions	Size	Flags	DPT	No. Comm. Obj.
Input x – Switching command	Input x = independent or single Function x = enabled Type = dimming	1 bit	CRWTU	[1.001] switch	13, 30 (1A, 1B) 51, 68 (2A, 2B)
Send a command to a dimming actuator to switch the light on or off. The command is triggered by a short press on the input. The value sent can be a fixed value or it can be toggled at each input activation.					
Input x – Dimming up / down / stop command	Input x = independent or single Function x = enabled Type = dimming	4 bit	CR-T-	[3.*] 3-bit control	14, 31 (1A, 1B) 52, 69 (2A, 2B)
Send a command to a dimming actuator to change dimming intensity (brighter or darker). Three values are used which mean start increase, start decrease or stop the change. Increase Decrease 1 0 0 0 0 0 0 1 Stop dimming 0 0 0 0 Increase/decrease values are sent when a long press action occurs and stop value on press release. The value sent can be a fixed value or it can be toggled at each input activation.					

9.0.6 Independent or single: shutter or venetian blind

Object name	Conditions	Size	Flags	DPT	No. Comm. Obj.
Input x – Dedicated stop command	Input x = independent or single Function x = enabled Type = shutter or venetian blind	1 bit	CRWTU	[1.017] trigger	13,30 (1A, 1B) 51,68 (2A, 2B)
<i>Immediately stop any movement of the blind. The object is sent on a short press if the blind mode is disabled and at the end of a long press if the venetian blind mode is enabled.</i>					
Input x – Stop – step up/down command	Input x = independent or single Function x = enabled Type = shutter or venetian blind Blind mode = enabled	1 bit	CR-T-	[1.007] step	16,33 (1A, 1B) 54,71 (2A, 2B)
<i>Move the blind to fully open or fully closed position. The object is sent at the end of a long press.</i>					
Input x – Move up / down command	Input x = independent or single Function x = enabled Type = shutter or venetian blind	1 bit	CRWTU	[1.008] up/down	17,34 (1A, 1B) 55,72 (2A, 2B)
<i>Increase or decrease the opening of the blind stepwise. The object is sent on a short press.</i>					

9.0.7 Independent or single: scene

Object name	Conditions	Size	Flags	DPT	No. Comm. Obj.								
Input x – Scene number	Input x = independent or single Function x = enabled Type = scene	1 Byte	CR-T-	[17.*] Scene number [18.*] Scene control	18,35 (1A, 1B) 61,73 (2A, 2B)								
Store or recall a scene. The lowest 6 bits in the byte form the code of the scene, while the highest bit is the operation code (store or recall). 1 Byte Bit number <table border="1" style="margin: auto;"><tr><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr></table> scene number (1-64) not used 0 = recall , 1 = save						7	6	5	4	3	2	1	0
7	6	5	4	3	2	1	0						

9.0.8 Coupled: switch

Object name	Conditions	Size	Flags	DPT	No. Comm. Obj.
Input x – Switching command	Input x = coupled Function x = enabled Type = switch	1-bit	CRWTU	[1.001] switch	13 51 89 127

9.0.9 Coupled: dimming

Object name	Conditions	Size	Flags	DPT	No. Comm. Obj.
Input x – Switching command	Input x = coupled Function x = enabled Type = dimming	1 bit	CRWTU	[1.001] switch	13 51 89 127
<i>See notes for independent input.</i>					
Input x – Dimming up / down / stop command	Input x = independent or single Function x = enabled Type = dimming	4 bit	CR-T-	[3.*] 3-bit control	14 52 90 128
<i>See notes for independent input.</i>					

9.0.10 Coupled: shutter or venetian blind

Object name	Conditions	Size	Flags	DPT	No. Comm. Obj.
Input x – Dedicated stop command	Input x = coupled Function x = enabled Type = shutter or venetian blind Blind mode = disabled	1 bit	CRWTU	[1.017] trigger	13 51 89 127
<i>See notes for independent input.</i>					
Input x – Stop – step up/down command	Input x = coupled Function x = enabled Type = shutter or venetian blind Blind mode = enabled	1 bit	CR-T-	[1.007] step	16 54 92 130
<i>See notes for independent input.</i>					
Input x – Move up / down command	Input x = coupled Function x = enabled Type = shutter or venetian blind	1 bit	CRWTU	[1.008] up/down	17 55 93 131
<i>See notes for independent input.</i>					

9.1 Input x: Function A/B configuration

9.1.1 Independent or single

For the *independent* or single channel setting, all parameters listed below are referred to either Function A or Function B (whichever are enabled).

In the following sections, it is implicitly understood that for the listed parameters to appear, the corresponding functions xA and/or xB must be enabled.

The entries assigned to “Object n” are repeated so many times as the number of configured objects according to the *Number of Communication Objects* parameter.

For all Type values

Parameter name	Conditions	Values
Lock function	-	enabled / disabled
	<i>Enables or disables the capability of locking the input through a remote command (telegram).</i>	
Lock function – Invert lock device signal	Input x = independent or single Type = send values or sequences	not inverted / inverted
	<i>Allows interpreting a “lock activate” telegram as unlock and vice-versa.</i>	
Lock function – Lock after bus recovery	Input x = independent or single Type = send values or sequences	no / yes
	<i>If active, after returning from a bus failure or power-off the device will retain the lock status it had before. Otherwise (in the default case), the device will restart in the non-locked condition.</i>	

9.1.2 Independent or single: Lock function enabled

Object name	Conditions	Size	Flags	DPT	No. Comm. Obj.
Input xx – Lock function	Input x = Independent or single Lock function = enabled	1 bit	C-W--	[1.003] enable	4,21 (1A, 1B) 42,59 (2A, 2B)

When the lock function is enabled, for each input or Input the user can define an action to execute when a lock or unlock command is received.

Details are shown in the following sections; a resume of all options is shown in the table below

Channel mode	Input type	Behaviour at locking	Behaviour at unlocking		
independent	send values or sequences	none as close or short press as open or long press			
	switching	none off on toggle	none off on as previous		
coupled					
independent	dimming				
coupled					
independent	scene	none send first scene send second scene			
independent	shutter or venetian blind	none up down			
coupled					

\

9.1.3 Independent or single: send values or sequences

Parameter name	Conditions	Values
Number of communication objects	Input x = independent or single Type = send values or sequences	1...8 (1)
	<i>Number of communication objects configured in association with the button event.</i>	
Lock function – Behaviour at locking	Input x = independent or single Type = send values or sequences	none as close or short press as open or long press
	<i>Allows performing the operation associated to the specified event when a locking command is received. You can choose between operations linked to two possible closing (or short press, depending on the configuration) or opening (or long press) events.</i>	
Lock function – Behaviour at unlocking	Input x = independent or single Type = send values or sequences	none as close or short press as open or long press
	<i>Allows performing the operation associated to the specified event when an unlocking command is received. You can choose between operations linked to two possible closing (or short press, depending on the configuration) or opening (or long press) events.</i>	
Event	Input x = independent or single Type = send values or sequences	close / open contact short / long press
	<i>Type of event that should be used as trigger for an action.</i>	
Long press time	Input x = independent or single Type = send values or sequences Event = short / long press	hh:mm:ss.fff (00:00:03.000)
	<i>Minimum push time for a press in order to be recognized as a long press.</i>	
Object n – Send delay	Input x = independent or single Type = send values or sequences	hh:mm:ss.fff (00:00:00.00)
	<i>Delay before the object is transmitted on the bus. By defining a delay after the event occurs and before the object value is sent, it is possible to associate a time defined sequence of values to an input event.</i>	
Object n – Send cyclically	Input x = independent or single Type = send values or sequences Number of communication objects = 1	none off / value 1 on / value 2 both off and on / both values
	<i>Defines which of the values, if any, must be cyclically retransmitted whenever activated. The cyclical transmission is only available if the number of communication objects to link is 1.</i>	
Object n – Cyclic sending interval	Input x = independent or single Type = send values or sequences Number of communication objects = 1 Send cyclically ≠ none	hh:mm:ss (00:02:00)
	<i>Interval between cyclical transmissions.</i>	
Object n – send dimension	Input x = independent or single Type = send values or sequences	1 bit value 2 bits value 1 byte unsigned value 1 byte percentage 1 byte signed value 2 bytes unsigned value 2 bytes signed value 2 bytes floating value
	<i>Defines size and type of the values to be sent when an event occurs.</i>	
Object n – Close or Short press	Input x = independent or single Type = send values or sequences send dimension = 1 bit	none on off toggle

Parameter name	Conditions	Values
	Input x = independent or single Type = send values or sequences send dimension = 2 bit	none disable enable off / up enable on / down enable off / up ↔ disable enable on / down ↔ disable enable off / up ↔ enable on / down
	Input x = independent or single Type = send values or sequences send dimension = any byte value	none send value 1 send value 2 send value 1 ↔ send value 2
<i>Value change behaviour caused by either a Close or a Short Press event (according to event configuration)</i>		
Object n – Open or Long press	Input x = independent or single Type = send values or sequences send dimension = 1 bit	none on off toggle
	Input x = independent or single Type = send values or sequences send dimension = 2 bit	none disable enable off / up enable on / down enable off / up ↔ disable enable on / down ↔ disable enable off / up ↔ enable on / down
	Input x = independent or single Type = send values or sequences send dimension = any byte value	none send value 1 send value 2 send value 1 ↔ send value 2
<i>Value change behaviour caused by either an Open or a Long Press event (according to event configuration)</i>		
Object n – Value 1	Input x = independent or single Type = send values or sequences send dimension = any byte value	0...255 (1 byte unsigned value) 0...100 (1 byte percentage) -128...127 (1 byte signed value) 0...65535 (2 bytes unsigned value) -32768... 32767 (2 bytes signed value) -671088.64...670760.96 (2 bytes floating value)
<i>First value available for association in send events</i>		
Object n – Value 2	Input x = independent or single Type = send values or sequences send dimension = any byte value	<i>same as value 1</i>
<i>Second value available for association in send events</i>		

Object name	Conditions	Size	Flags	DPT	No. Comm. Obj.
Input xx – Switching status [type] Object n	Input x = Independent or single Type = send values or sequences	see the table below	CR-TU	see the table below	3..35 (1A, 1B) 36...69 (2A, 2B)

The data sizes and DPT are as follows:

Dimension	DPT
1 bit	[1.001] switch
2 bits	[2.*] 1-bit controlled
1 byte unsigned	[4.*] Character [5.*] 8-bit unsigned value [20.*] 1-byte
1 byte percentage	[4.*] Character [5.*] 8-bit unsigned value [20.*] 1-byte
1 byte with sign	[6.*] 8-bit signed value

2 bytes unsigned	[7.*] 2-byte unsigned value
2 bytes with sign	[8.*] 2-byte signed value
2 bytes floating point	[9.*] 2-byte float value

9.1.4 Independent or single: dimming

Parameter name	Conditions	Values
Long press time	Input x = independent or single Type = dimming	hh:mm:ss.fff (00:00:03.000)
	<i>Minimum push time for a press in order to be recognized as a long press.</i>	
Toggle mode	Input x = independent or single Type = dimming	enabled / disabled
	<i>When enabled, causes the short press to toggle the on-off status of the destination CO; otherwise, a fixed status can be assigned to the short press.</i>	
Long action	Input x = independent or single Type = dimming Toggle mode = enabled	darker brighter darker ↔ brighter
	<i>Defines the function to be assigned to the long press. If the toggle mode is enabled, the Short press action is already defined as toggle.</i>	
Short / Long action	Input x = independent or single Type = dimming Toggle mode = disabled	off / darker on / brighter off / darker ↔ brighter on / darker ↔ brighter
	<i>Defines the function to be assigned to the long and short press.</i>	
Send cyclically	Input x = independent or single Type = dimming	none off / value 1 on / value 2 both off and on / both values
	<i>Defines which of the values, if any, must be cyclically retransmitted whenever activated.</i>	
Cyclic sending interval	Input x = independent or single Type = dimming Send cyclically ≠ none	hh:mm:ss (00:02:00)
	<i>Interval between cyclical transmissions.</i>	
Lock function – Behaviour at locking	Input x = independent or single Type = dimming	none off on toggle
	<i>Value to be assigned to the object when a locking command is received.</i>	
Lock function – Behaviour at unlocking	Input x = independent or single Type = dimming	none off on as previous
	<i>Value to be assigned to the object when an unlocking command is received.</i>	

9.1.6 Independent or single: scene

Parameter name	Conditions	Values
First scene number	Input x = independent or single Type = scene	1...64 (1)
	<i>Main scene number to be assigned to button press. It is named "first" for the case that an alternative scene number is used.</i>	
Learning mode	Input x = independent or single Type = scene	enabled / disabled
	<i>When enabled, a long key press can be used to program the selected scene by storing the current parameters.</i>	
Long press time	Input x = independent or single Type = scene Learning mode = enabled	hh:mm:ss.fff (00:00:03.000)
	<i>Minimum push time for a press in order to be recognized as a long press.</i>	
Scene activation	Input x = independent or single Type = scene Learning mode = disabled	send first scene only toggle between two scenes
	<i>Allows the key to be used to alternate between two different scenes.</i>	
Second scene number	Input x = independent or single Type = scene Learning mode = disabled Scene activation = toggle between two scenes	1...64 (2)
	<i>Alternate scene number to be assigned to button press.</i>	
Lock function – Behaviour at locking	Input x = independent or single Type = scene	none send first scene send second scene
	<i>Operation to perform when a locking command is received.</i>	
Lock function – Behaviour at unlocking	Input x = independent or single Type = scene	none send first scene send second scene
	<i>Operation to perform when an unlocking command is received.</i>	

9.1.7 Coupled

For a *coupled* channel, all the parameters are referred to the single menu entry for Function xA and xB.

In the following sections, it is implicitly understood that for the listed parameters to appear, the corresponding functions xA and xB must be enabled.

For all Type values:

Parameter name	Conditions	Values
Lock function	Input x = coupled	enabled / disabled
<i>Enables or disables the capability of locking the input through a remote command (telegram).</i>		

9.1.8 Coupled: Lock function enabled

Object name	Conditions	Size	Flags	DPT	No. Comm. Obj.
Input xx – Lock function	Input x = coupled Lock function = enabled	1 bit	C-W--	[1.003] enable	4 42

9.1.9 Coupled: switch

Parameter name	Conditions	Values
xA and xB use	Input x = coupled Type = switch	A on, B off A off, B on
Allows to invert side A and side B functionalities		
Send cyclically	Input x = coupled Type = switch	none off / value 1 on / value 2 both off and on / both values
<i>Defines which of the values, if any, must be cyclically retransmitted whenever activated.</i>		
Cyclic sending interval	Input x = coupled Type = switch Send cyclically ≠ none	hh:mm:ss (00:02:00)
<i>Interval between cyclical transmissions.</i>		
Lock function – Behaviour at locking	Input x = coupled Type = switch	none on off toggle
<i>Value to be assigned to the object when a locking command is received.</i>		
Lock function – Behaviour at unlocking	Input x = coupled Type = switch	none on off as previous
<i>Value to be assigned to the object when an unlocking command is received.</i>		

9.1.10 Coupled: dimming

Parameter name	Conditions	Values
Long press time	Input x = coupled Type = dimming	hh:mm:ss.fff (00:00:03.000)
	<i>Minimum push time for a press in order to be recognized as a long press.</i>	
xA and xB use	Input x = coupled Type = dimming	A increases, B decreases A decreases, B increases
Send cyclically	Input x = coupled Type = dimming	none off / value 1 on / value 2 both off and on / both values
	<i>Defines which of the values, if any, must be cyclically retransmitted whenever activated.</i>	
Cyclic sending interval	Input x = coupled Type = dimming Send cyclically ≠ no	hh:mm:ss (00:02:00)
	<i>Interval between cyclical transmissions.</i>	
Lock function – Behaviour at locking	Input x = coupled Type = dimming	none on off toggle
	<i>Value to be assigned to the object when a locking command is received.</i>	
Lock function – Behaviour at unlocking	Input x = coupled Type = dimming	none on off as previous
	<i>Value to be assigned to the object when an unlocking command is received.</i>	

9.1.11 Coupled: shutter or venetian blind

Parameter name	Conditions	Values
Long press time	Input x = coupled Type = shutter or venetian blind	hh:mm:ss.fff (00:00:03.000)
	<i>Minimum push time for a press in order to be recognized as a long press.</i>	
xA and xB use	Input x = coupled Type = shutter or venetian blind	A up, B down A down, B up
Blind mode	Input x = coupled Type = shutter or venetian blind	enabled / disabled
	<i>If blinds mode is enabled, the device sends "full movement" telegrams on long press and "step" telegrams on short press; if it is disabled, the device sends "full movement" telegrams on long press and "stop" telegrams on short press.</i>	
Lock function – Behaviour at locking	Input x = coupled Type = shutter or venetian blind	none up down
	<i>Operation to perform when a locking command is received.</i>	
Lock function – Behaviour at unlocking	Input x = coupled Type = shutter or venetian blind	none up down
	<i>Operation to perform when an unlocking command is received.</i>	

9.2 Sensor temperature

The Temperature Sensor tab is available for Channel 1 – Input 1 only.
It is enabled by selecting “Input 1 = enabled NTC” and contains the following parameters:

- Enable the sensor
- Filter type on internal data processing
- Measured temperature correction
- Minimum variation for sending value [K]
- Cyclic delivery interval
- Threshold 1
- Threshold 2

9.2.1 Parameters and communication objects

Parameter name	Conditions	Values
Temperature sensor		enabled disabled
	<i>The temperature sensor is enabled as default.</i>	
Filter type	Temperature sensor = enabled	low medium high
	<i>Low = average value every 4 measurements</i> <i>Medium = average value every 16 measurements</i> <i>High = average value every 64 measurements</i>	
Temperature offset	Temperature sensor = enabled	0°C [range -5°C ... +5°C]
Minimum change of value to send [K]	Temperature sensor = enabled	0,5 [range 0 ...5]
	<i>If the parameter is set to 0 (zero), no value is sent after a change.</i>	
Cyclic sending interval	Temperature sensor = enabled	no sending [other values in the range 30 s ... 120 min]
Threshold 1	Temperature sensor = enabled	not active below above
Value [°C]	Temperature sensor = enabled, Threshold 1 = below or above	7 [range 0 ... 50]
Threshold 2	Temperature sensor = enabled	not active below above
Value [°C]	Temperature sensor = enabled, Threshold 2 = below or above	45 [range 0 ... 50]
Hysteresis	Temperature sensor = enabled, Threshold 1 and/or Threshold 2 = below or above	0,4 K [other values between 0,2 K and 3 K]
Cyclic sending interval	Temperature sensor = enabled, Threshold 1 and/or Threshold 2 = below or above	no sending [other values in the range 30 s ... 120 min]

--	--

Object name	Conditions	Dim.	Flags	DPT	Comm. Obj. No.
Temperature value	Temperature sensor = enabled	2 Bytes	CR-T--	[9.001] temperature (°C)	
Temperature threshold1 - Switch	Temperature sensor = enabled, Threshold 1 = below or above	1 Bit	CR-T--	[1.001] switch	
Temperature threshold 2- Switch	Temperature sensor = enabled, Threshold 2 = below or above	1 Bit	CR-T--	[1.001] switch	

9.2.2 Acquisition filter

The acquisition filter calculates an average with a series of measured values before sending on the bus. The parameter can have the following values:

- low = average value every 4 measurements;
- medium = average value every 16 measurements;
- high = average value every 64 measurements.

9.2.3 Correction of the measured temperature

The sampling of the temperature value occurs every 10 seconds, while the display is updated every minute. During the configuration with ETS the opportunity is given to correct the measured temperature value within the offset range of - 5 °C ... + 5 °C (step: 0.1 K).

9.2.4 External sensors (from bus)

As “external sensors” are intended KNX-devices (or conventional sensors interfaced to the bus through KNX devices) which send states or values to the room temperature controller via the bus. Enabling an external sensor, without connecting the corresponding communication object, generates a permanent alarm on the display and suspends the thermoregulation function.

The folder **External sensors (from bus)** includes the following parameters:

- Room brightness
- Room temperature
- Relative humidity
- Antistratification temperature
- Outdoor temperature
- Floor surface temperature
- Flow temperature
- Anticondensation
- Window contact X (X = 1, 2)
- Presence sensor X (X = 1, 2)
- Card holder contact
- Sensor timeout

9.2.5 Parameter and communication object tables

Parameter name	Conditions	Values
Room temperature		disabled / enabled
	<i>It enables a bus temperature sensor. The measured value can be used to calculate a weighted average value in combination with the temperature sensor integrated into the device or a temperature sensor connected to a device input.</i>	
Cyclic reading interval	Room temperature = enabled	no reading [other values in the range 30 s ... 120 min]
	<i>If the parameter is set to “no reading”, the corresponding communication object must be updated by the remote device sending data. With any different value, data are updated with a reading request by the room thermostat.</i>	
Relative humidity		disabled / enabled
Humidity CO dimension	Relative humidity = enabled	1 byte (DPT 5.001) 2 byte (DPT 9.007)
Cyclic reading interval	Relative humidity = enabled	no reading [other values in the range 30 s ... 120 min]
Antistratification temperature		disabled / enabled
	<i>It enables a temperature bus sensor to carry out the antistratification function.</i>	
Cyclic reading interval	Antistratification temperature = enabled	no reading [other values in the range 30 s ... 120 min]

Outdoor temperature		disabled / enabled
	<i>It enables an outdoor temperature bus sensor to display the measured value on the display. This is alternative to an outdoor temperature sensor connected to a device input: the parameter appears only if the external temperature sensor is disabled in the Inputs folder.</i>	
Light sensor		disabled / enabled
Cyclic reading interval	Light sensor = enabled	no reading [other values in the range 30 s ... 120 min]
Coil temperature		disabled / enabled
	<i>It enables a bus sensor for measuring the coil temperature of the conveying fluid for heat exchange. The acquisition of the value allows realizing the hot-start function of a fan.</i>	
Cyclic reading interval	Coil temperature = enabled	no reading [other values in the range 30 s ... 120 min]
Floor surface temperature		disabled / enabled
	<i>It enables a bus sensor for measuring the surface temperature of a floor heating system. The acquisition of the value allows to realize the function of surface temperature limitation.</i>	
Cyclic reading interval	Floor surface temperature = enabled	no reading [other values in the range 30 s ... 120 min]
Flow temperature		disabled / enabled
	<i>It enables a bus sensor for measuring the flow temperature of the conveying fluid. The acquisition of the value allows calculating the dew-point temperature to realize the active anticondensation protection function in surface cooling plants (floor or ceiling).</i>	
Cyclic reading interval	Flow temperature = enabled [other	no reading values in the range 30 s ... 120 min]
Analog sensors timeout		00:05:00hh:mm:ss [range 00:00:00 ... 18:12:15]
	<i>The field has format hh:mm:ss (hours : minutes : seconds): the default value 00:05:00 corresponds to a timeout of 5 minutes. The value 00:00:00 means that the timeout of the analogic sensors is disabled.</i>	
Anticondensation		disabled / enabled
	<i>It enables a bus sensor for detecting the condensation.</i>	
Signal	Anticondensation = enabled	not inverted / inverted
Cyclic reading interval	Anticondensation = enabled	no reading [other values in the range 30 s ... 120 min]
Window contact 1		disabled / enabled
	<i>It enables a bus sensor for detecting the state of opening / closing of a window or a door.</i>	

Signal	Window contact 1= enabled	not inverted / inverted
Cyclic reading interval	Window contact 1= enabled	no reading [other values in the range 30 s ... 120 min]
Window contact 2		disabled / enabled
	<i>It enables a bus sensor for detecting the state of opening / closing of a window or a door.</i>	
Signal	Window contact 2= enabled	not inverted / inverted
Cyclic reading interval	Window contact 2= enabled	no reading [other values in the range 30 s ... 120 min]
Presence sensor 1		disabled / enabled
	<i>It enables a bus sensor for detecting the presence / absence of people within a room.</i>	
Signal	Presence sensor 1= enabled	not inverted / inverted
Cyclic reading interval	Presence sensor 1= enabled	no reading [other values in the range 30 s ... 120 min]
Presence sensor 2		disabled / enabled
	<i>It enables a bus sensor for detecting the presence / absence of people within a room.</i>	
Signal	Presence sensor 2= enabled	not inverted / inverted
Cyclic reading interval	Presence sensor 2= enabled	no reading [other values in the range 30 s ... 120 min]
Card holder contact		disabled / enabled
	<i>It enables a bus sensor for detecting the presence / absence of people in a hotel room provided with a card holder.</i>	
Signal	Card holder contact = enabled	not inverted / inverted
Cyclic reading interval	Card holder contact = enabled	no reading [other values in the range 30 s ... 120 min]
Digital sensors timeout		00:05:00hh:mm:ss [range 00:00:00 ... 18:12:15]
	<i>The field has format hh:mm:ss (hours : minutes : seconds): the default value 00:05:00 corresponds to a timeout of 5 minutes. The value 00:00:00 means that the timeout of the digital sensors is disabled.</i>	

Object name	Conditions	Dim.	Flags	DPT	Comm. Obj. No.
Brightness value (from bus)	enabled	2 Byte	C-W---	[9.004] Lux (lux)	4
Room temperature (from bus)	enabled	2 Byte	C-W---	[9.001] temperature (°C)	5
Humidity (2 bytes, from bus)	Relative humidity sensor = enabled, Humidity comm. obj. size = 2 byte	2 Byte	C-W---	[9.007] humidity (%)	6
Humidity (1 byte, from bus)	Relative humidity sensor = enabled, Humidity comm. obj. size = 1 byte	1 Byte	C-W---	[5.001] percentage (0..100%)	7
Antistratification temperature (from bus)	enabled	2 Byte	C-W---	[9.001] temperature (°C)	8
Outdoor temperature (from bus)	enabled	2 Byte C-	W--- 9	[9.001] temperature °C	
Coil temperature (from bus)	enabled	2 Byte	C-W---	[9.001] temperature (°C)	10
Floor temperature (from bus)	enabled	2 Byte	C-W---	[9.001] temperature (°C)	11
Flow temperature (from bus)	enabled	2 Byte	C-W---	[9.001] temperature (°C)	12
Anticondensation (from bus)	enabled	1 Bit	C-W---	[1.001] switch	18
Windows contact sensor 1 (from bus)	enabled	1 Bit	C-W---	[1.019] window/door	13
Windows contact sensor 2 (from bus)	enabled	1 Bit	C-W---	[1.019] window/door	14
Presence sensor 1 (from bus)	enabled	1 Bit	C-W---	[1.018] occupancy	15
Presence sensor 2 (from bus)	enabled	1 Bit	C-W---	[1.018] occupancy	16
Contact of card holder (from bus)	enabled	1 Bit	C-W---	[1.001] switch	17

About sensor timeout

The internal control system of the thermostat cyclically monitors the updating status of the values of the external sensors (from bus) and the inputs when the timeout setting expires. In case no updated value has been received, the regulation function is suspended, an alarm is displayed on the display through the symbol and the corresponding alarm code (see also the list of alarms in the paragraph Diagnostics)

9.3 Weighted temperature value

The **Weighted temperature value** folder appears only if two sensors for measuring the room temperature are enabled and includes the following parameters:

- Relative weight
- Minimum change of value to send [K]
- Cyclic sending interval

9.3.1 Parameter and communication object tables

Parameter name	Conditions	Values
Relative weight		100% main sensor 90% / 10% 80% / 20% 70% / 30% 60% / 40% 50% / 50% 40% / 60% 30% / 70% 20% / 80% 10% / 90% 100% sensor from bus
Minimum change of value to send [K]		0,5 [other values in the range 0 ... 5 K]
	If the parameter is set to 0 (zero), no value is sent at the change.	
Cyclic sending interval		no sending [other values in the range 30 s ... 120 min]

Object name	Conditions	Dim.	Flags	DPT	Comm. Obj. No.
Weighted temperature	Cyclic sending interval ≠ no sending	2 Byte	CR-T--	[9.001] temperature °C	19

About weighted temperature

The device allows the acquisition of the room temperature in two ways:

- 1) from the temperature sensor integrated in the device;
- 2) via bus from another KNX device, e.g. from an ekinex pushbutton (External sensors (from bus) ☐ Room temperature = enabled);

To optimize or correct the room temperature regulation in special cases (in large rooms, in presence of strong asymmetry of the temperature distribution, when the installation of the device is in a position not suitable, etc.), the device can then use a weighted average between two temperature values. The weights are assigned by the parameter *Relative weight* that assigns a ratio of the two values

9.4 Temperature control

The **Temperature control** folder includes the following secondary folders:

- Settings
- Heating
- Cooling
- Ventilation
- Scenes

The **Cooling** and **Ventilation** secondary folders appear only if in the **Settings** folder the parameter Thermostat function is set to the value *both heating and cooling* or *cooling*.

The **Scenes** secondary folder appears only if in the **Settings** secondary folder the parameter Scenes is set to the value *enabled*.

9.4.1 Settings

The **Settings** folder includes the following parameters:

- Thermostat function
- Command Communication Object
- Heating – cooling switchover
- Setpoint Cyclic sending interval
- Max manual temperature change
- Saving timeout (manual change)
- End of manual operation
- Max setpoint temperature change
- Scenes
- Valve protection function
- Frequency
- Time interval

9.4.1.1 Parameter and communication object tables

Parameter name	Conditions	Values
Thermostat function		heating cooling both heating and cooling
Command Communication Object	Thermostat function = both heating and cooling	separated / unique
Heating–cooling switch over	Thermostat function = both heating and cooling	manual from bus automatic
Heating-cooling cyclic sending interval	Thermostat function = both heating and cooling	no sending [other values in the range 30 s ... 120 min]
Parameter name	Conditions	Values

Setpoint cyclic sending interval		no sending [other values in the range 30 s ... 120 min]
	<i>The setpoint value that can be sent cyclically is the actual one, depending on the operating mode set manually by the user or automatically by another KNX supervising device with the possibility of time scheduling. The actual setpoint value takes also into account the actual state of the contacts window and presence detection (if the corresponding functions are enabled).</i>	
Max manual temperature change		not allowed, $\pm 1^{\circ}\text{C}$, $\pm 2^{\circ}\text{C}$, $\pm 3^{\circ}\text{C}$, $\pm 4^{\circ}\text{C}$, $\pm 5^{\circ}\text{C}$, $\pm 6^{\circ}\text{C}$, $\pm 7^{\circ}\text{C}$, $\pm 8^{\circ}\text{C}$, $\pm 9^{\circ}\text{C}$, $\pm 10^{\circ}\text{C}$
	<i>It defines the maximum range allowed for the manual change of the temperature value.</i>	
End of manual operation	General ☐ Device operation as = [other values in the	till first telegram from bus stand-alone range 30 min ... 48 h]
Exit manual mode on setpoint from bus		no / si
	<i>It defines the exit from manual/forced mode in case of setpoint modification on bus (communication objects index from 29 to 36)</i>	
Max setpoint temperature change		not allowed, $\pm 1^{\circ}\text{C}$, $\pm 2^{\circ}\text{C}$, $\pm 3^{\circ}\text{C}$, $\pm 4^{\circ}\text{C}$, $\pm 5^{\circ}\text{C}$, $\pm 6^{\circ}\text{C}$, $\pm 7^{\circ}\text{C}$, $\pm 8^{\circ}\text{C}$, $\pm 9^{\circ}\text{C}$, $\pm 10^{\circ}\text{C}$
	<i>It defines the maximum time allowed for changing the values of temperature setpoint in the several operating modes.</i>	
Duration of comfort extension	General ☐ Device operation as = range 15 min ... 4 h]	not allowed , slave [other values in the
	<i>When device operation as = slave, if the operating mode forced by the time scheduling is not comfort or building protection, this parameter defines the maximum duration of temporary comfort before returning to the programmed operating mode.</i>	
Scenes		disabled / enabled
Transmission delay after mode change		
Valve protection function		disabled / enabled
	<i>It enables the function that activates the drive for the valve control during periods of inactivity of the system.</i>	
Frequency	Valve protection function = enabled	once a day once a week once a month
Time interval	Valve protection function = enabled	10 s [other values in the range 5 s ... 20 min]

Object name	Conditions	Dim.	Flags	DPT	Comm. Obj. No.
Actual setpoint		2 Byte	CR-T--	[9.001] temperature ($^{\circ}\text{C}$)	27
Manual setpoint		2 Byte	C-W---	[9.001] temperature ($^{\circ}\text{C}$)	28

Object name	Conditions	Dim.	Flags	DPT	Comm. Obj. No.
Heating/cooling status out	Always visible	1 Bit	CR-T--	[1.100] heating/cooling	20
The communication object is updated on the bus on event of change internally elaborated by the controller. The object is always exposed and contains the information about the current conduction mode of the internal temperature controller. [1.100] DPT Heat/Cool 1 Bit 					
Heating/cooling status in	Thermostat function = both heating and cooling; Heating – cooling switchover = from bus	1 Bit	C-W---	[1.100] heating/cooling	21
The communication object is received by the bus. On switching event, internal controllers of primary and auxiliary stage (if enabled) switch their operating mode. The actual operating mode is displayed by the corresponding icon.					
HVAC mode in		1 Byte	C-W---	[20.102] HVAC mode	22
The device receives the operating mode (HVAC mode) from a bus device with function of supervisor. The operating mode received through this communication object can be later modified by the user (in this case the room thermostat switches to manual mode).					
HVAC forced mode in		1 Byte	C-W---	[20.102] HVAC mode	23
The communication object allows to receive the operating mode same way as "HVAC mode in" communication object, except the operating mode received through this object (apart from AUTO command) can no longer be modified by user. User can modify the operating mode only after "HVAC forced mode in" has sent AUTO command.					
HVAC mode out		1 Byte	CR-T--	[20.102] HVAC mode	24
HVAC manual mode		1 Byte	C-WTU-	[20.102] HVAC mode	25
Chrono active status		1 Bit	CR-T--	[1.011] state	26
Manual/forced setpoint active status		1 Bit	CRWTU-	[1.011] state	50
Building protection HVAC mode active		1 Bit	CR-T--	[1.011] state	74

About heating/cooling terminals

The application functions of the room temperature controller configurable with ETS are particularly suitable for the control through general-purpose or dedicated KNX actuators of the following heating/cooling terminals:

- radiators;
- electric heaters;
- fancoils;
- radiant panels;
- dehumidification units;
- radiant panels + radiators (as auxiliary system);
- radiant panels + fancoils (as auxiliary system);
- radiant panels + dehumidification units.

9.5 Heating

The **Heating** folder includes the following parameters:

- Comfort temp. setpoint [°C]
- Standby temperature setpoint [°C]
- Economy temp. setpoint [°C]
- Building protection temp. setpoint [°C]
- Heating type
- Control type
- Hysteresis
- Cyclic sending interval
- Min. change of value to send [%]
- PWM cycle time
- Proportional band [0,1 K]
- Integral time [min]
- Floor temperature limitation
- Temperature limit [°C]
- Hysteresis [K]
- Auxiliary heating
- Communication object
- Disabled from bus
- Offset from setpoint
- Hysteresis
- Cyclic sending interval
- Ventilation for auxiliary heating

9.5.1 Parameter and communication object tables

Conditions: *Settings* □ Thermostat function = heating or both heating and cooling.

Parameter name	Conditions	Values
Comfort temp. setpoint [°C]		21 [range 10 ... 50]
Standby temp. setpoint [°C]		18 [range 10 ... 50]
	<i>For a correct operation of the device the standby temperature setpoint has to be < comfort temperature setpoint.</i>	
Economy temp. setpoint [°C]		16 [range 10 ... 50]
	<i>For a correct operation of the device the economy temperature setpoint has to be < standby temperature setpoint.</i>	
Building protection temp. setpoint [°C]		7 [range 2 ... 10]
Heating type		radiators electric fan-coils floor radiant panels ceiling radiant panels

	It defines the terminal used for the thermal exchange in the room. The choice affects the parameters of the PWM control algorithm (Proportional band and Integral time) and the control options.	
Control type		2 point hysteresis PWM (pulse width modulation) continuous
Hysteresis	Control type = 2 point hysteresis	0,3 K [other values in the range 0,2 K ... 3 K]
Hysteresis position	Heating type = floor radiant panels, ceiling radiant panels, Control type = 2 point hysteresis	below / above
	The above hysteresis is suitable in case of special applications requiring mixing group control.	
Cyclic sending interval	Control type = 2 point hysteresis, continuous	no sending [other values in the range 30 s ... 120 min]
Min. change of value to send [%]	Control type = continuous	10 [range 0 ... 100]
PWM cycle time	Control type = PWM	15 min [range 5 ... 240 min]
Proportional band [0,1 K]	Control type = continuous or PWM	* [range 0 ... 255]
	The value is in tenths of Kelvin (K) degree. *) The field contains a preset value that depend on the selected heating type (the value can be modified): □ radiators: 50 (5 K) • electric: 40 (4 K) • fan-coils: 40 (4 K) • floor radiant panels: 50 (5 K) • ceiling radiant panels: 50 (5 K) The value of the parameter Proportional band represents the max difference between the setpoint temperature and the measured temperature that causes the max control output.	
Integral time [min]	Control type = continuous or PWM	* [other values in the range 0 ... 255 min]
	*) The field contains a preset value that depend on the selected heating type (the value can be modified): • radiators: 150 min • electric: 100 min • fan-coils: 90 min • floor radiant panels: 240 min • ceiling radiant panels: 180 min	
Min control value [%]	Control type = continuous or PWM	15 [range 0 ... 30]

Parameter name	Conditions	Values
Max control value [%]	Control type = continuous or PWM	85 [range 70 ... 100]
Floor temperature limitation	Heating type = floor radiant panels, External sensors □ Floor surface temperature sensor = enabled	disabled / enabled
	<i>This parameter enables the floor temperature limitation of a floor radiant panel. It is mandatory to measure the floor surface temperature by enabling the corresponding temperature sensor in "External sensors (from bus)" folder.</i> Important! This function does not replace the overtemperature protection usually installed in hydronic floor systems, realized with the proper safety thermostat.	
Temperature limit [°C]	Floor temperature limitation = enabled	29 [range 20 ... 40]
	According to EN 1264 a maximum allowed temperature is prescribed for the surface of a floor heating system: • $T(\text{sup}) \text{ max} \leq 29^{\circ}\text{C}$ per le zone di normale occupazione; • $T(\text{sup}) \text{ max} \leq 35^{\circ}\text{C}$ per le zone periferiche degli ambienti. National standard may limit those temperatures to lower values. Per zone periferiche si intendono fasce situate generalmente lungo i muri dell'ambiente rivolti verso l'esterno dell'edificio con larghezza massima di 1 m.	
Hysteresis [K]	Floor temperature limitation = enabled	0,3 K [other values in the range 0,2 K ... 3 K]
	Before quitting from the alarm status, the device waits until the surface temperature decreases under the threshold set offset pari al valore di isteresi.	
Auxiliary heating		disabled / enabled
Communication object	Auxiliary heating = enabled	separated unique
Disabled from bus	Auxiliary heating = enabled	no / yes
	It enables the activation and deactivation of the function through a telegram sent on the bus by a supervising device.	
Offset from setpoint	Auxiliary heating = enabled	0,6 K [other values in the range 0 ... 3 K]
Hysteresis [K]	Auxiliary heating = enabled	0,3 K [other values in the range 0,2 K ... 3 K]
Cyclic sending interval	Auxiliary heating = enabled	no sending [other values in the range 30 s ... 120 min]
Ventilation for auxiliary heating	Heating type = floor radiant panels or ceiling radiant panels	disabled / enabled
	<i>This option allows matching a system with high inertia as the floor radiant panels (hydronic version) to a system with low inertia as the fan-coils.</i>	

Object name	Conditions	Dim.	Flags	DPT	Comm. Obj. No.
Comfort setpoint (heating)		2 Byte	CRWTU-	[9.001] temperature (°C)	29
Standby setpoint (heating)		2 Byte	CRWTU-	[9.001] temperature (°C)	31
Economy setpoint (heating)		2 Byte	CRWTU-	[9.001] temperature (°C)	33
Building protection setpoint (heating)		2 Byte	CRWTU-	[9.001] temperature (°C)	35
Heating out command	Control type = 2 points hysteresis or PWM, Command communication object = separated	1 Bit	CR-T--	[1.001] switch	38
Heating out command	Control type = continuous, Command communication object = separated	1 Byte	CR-T--	[5.001] percentage (0..100%)	38
Heating and cooling out command	Tipo Control type = 2 points hysteresis or PWM, Command communication object = unique	1 Bit	CR-T--	[1.001] switch	38
Heating and cooling out command	Control type = continuous, Command communication object = unique	1 Byte	CR-T--	[5.001] percentage (0..100%)	38
Auxiliary heating out command	Auxiliary heating = enabled, Command communication object = separated	1 Bit	CR-T--	[1.001] switch	40
Auxiliary heating and cooling output command	Auxiliary heating = enabled, Command communication object = unique	1 Bit	CR-T--	[1.001] switch	40
Auxiliary heating disable	Auxiliary heating = enabled, Disabled from bus = yes		1 Bit	C-W--- [1.003] enable 42	
Temperature setpoint change lock		1 Bit	C-W---	[1.003] enable	37

About floor temperature limitation function

The floor heating system (warm water version) provides plastic pipes embedded in the concrete layer or placed directly under the final coating of the floor (light or "dry" system) filled by heated water. The water releases heat to the final coating that heats the room by radiation. The standard EN 1264 Floor heating (Part 3: Systems and components - Dimensioning) prescribes a maximum allowed temperature (T_{Smax}) for the surface of the floor that is physiologically correct defined as:

- $T_{Smax} \leq 29^{\circ}\text{C}$ for zones of normal occupancy;

- $T_{Smax} \leq 35^{\circ}\text{C}$ for peripheral zones of the rooms.

National standards may also limit these temperatures at lower values. Peripheral zones are strips generally located along the external walls with a maximum width of 1 m.

The floor heating system (electrically powered version) involves the laying under the floor coating of an electric cable powered by the mains voltage (230 V) or low voltage (for example 12 or 45 V), possibly already prepared in the form of rolls with constant distance between sections of cable. The powered cable releases heat to the overlying coating that heats the room by radiation. The regulation is based on measurement of the temperature of the air mass, but generally requires the monitoring and limiting of the surface temperature by using a NTC-type sensor which is in contact with the floor surface.

The surface temperature limitation may be implemented for several purposes:

- physiological compatibility (correct temperature at the height of the legs);
- when the system is used as auxiliary stage for heating. In this case, the heat losses to the exterior of the building are handled by the main heating stage, while the auxiliary stage only works to keep the floor temperature at a comfortable level (for example in bathrooms of residential buildings, sports centers, spas and thermal baths, etc.);
- protection against damages of the final coating due to an accidental overheating. Note that the warm water radiant panels are usually already equipped with a safety thermostat (with intervention on the hydraulic mixing group), while in the case of electrical power this device is not usable and it is common practice to realize a temperature limitation with a surface temperature sensor connected to the device.

9.6 Cooling

The **Cooling** folder includes the following parameters:

- Comfort temp. setpoint [°C]
- Standby temperature setpoint [°C]
- Economy temp. setpoint [°C]
- Building protection temp. setpoint [°C]
- Cooling type
- Control type
- Hysteresis
- Cyclic sending interval
- Min. change of value to send [%]
- PWM cycle time
- Proportional band [0,1 K]
- Integral time [min]
- Anticondensation with probe
- Anticondensation protection type 2
- Auxiliary cooling
- Disabled from bus
- Offset from setpoint
- Hysteresis ON/OFF [K]

9.6.1 Parameter and communication object tables

Conditions: *General* □ Thermostat function = cooling or both heating and cooling.

Parameter name	Conditions	Values
Comfort temp. setpoint [°C]		23 [range 10 ... 50]
Standby temp. setpoint [°C]		26 [range 10 ... 50]
	<i>For a correct operation of the device the standby temperature setpoint has to be > comfort temperature setpoint.</i>	
Economy temp. setpoint [°C]		28 [range 10 ... 50]
	<i>For a correct operation of the device the standby temperature setpoint has to be > economy temperature setpoint.</i>	
Building protection temp. setpoint [°C]		36 [range 30 ... 50]
Cooling type		fan-coils floor radiant panels ceiling radiant panels
	<i>If in Settings the parameter Thermostat function = both heating and cooling and Command communication object = unique, the parameter Cooling type is bound to the choice done for Heating.</i>	
Control type	Command communication object = separated	2 point hysteresis PWM (pulse width modulation) continuous
	<i>If in Settings the parameter Thermostat function = both heating and cooling and Command communication object = unique, the parameter Control type is bound to the choice done for Heating.</i>	

Parameter name	Conditions	Values
Hysteresis	Control type = 2 point hysteresis	0,3 K [other values in the range 0,2 K ... 3 K]
Hysteresis position	Cooling type = floor radiant panels, ceiling radiant panels, Control type = 2 point hysteresis	below / above
	<i>The above hysteresis is suitable in case of special applications requiring mixing group control.</i>	
Cyclic sending interval	Control type = 2 point hysteresis or continuous	no sending [other values in the range 30 s ... 120 min]
Min. change of value to send [%]	Control type = continuous	10 [range 0 ... 100]
PWM cycle time	Control type = PWM (puls width modulation)	15 min [range 5 ... 240 min]
Proportional band [0,1 K]	Control type = continuous or PWM	* [range 0 ... 255]
	<i>The value is in tenths of Kelvin (K) degree.</i> <i>*) The field contains a preset value that depend on the selected cooling type (the value can be modified):</i> • fan-coils: 40 (4 K) • floor radiant panels: 50 (5 K) • ceiling radiant panels: 50 (5 K) <i>The value of the parameter Proportional band represents the max difference between the setpoint temperature and the measured temperature that causes the max control output.</i>	
Integral time [min]	Control type = continuous or PWM	* [range 0 ... 255 min]
	<i>*) The field contains a preset value that depend on the selected cooling type (the value can be modified):</i> □ fan-coils: 90 min • floor radiant panels: 240 min • ceiling radiant panels: 180 min	
Min control value [%]	Control type = continuous or PWM	15 [range 0 ... 30]
Max control value [%]	Control type = continuous or PWM	85 [range 70 ... 100]
Anticondensation with probe	Cooling type = floor radiant panels or ceiling radiant panels, External sensors (from bus) □ Anticondensation = enabled	disabled / enabled

Parameter name	Conditions	Values
Active anticondensation	Cooling type = floor radiant panels or ceiling radiant panels, External sensors (from bus) ⇒ Flow temperature sensor = enabled	disabled enabled (project temperature)
	Cooling type = floor radiant panels or ceiling radiant panels, External sensors (from bus) ⇒ Flow temperature sensor = enabled	disabled enabled (comparison between flow temperature and dew-point)
<i>If flow temperature is lower than calculated dew point, the operating mode is cooling and the room thermostat is in flow request, then the thermostat will close the valve and display an alarm condition.</i>		
Flow temperature (project)	Cooling type = floor radiant panels or ceiling radiant panels, External sensors (from bus) ⇒ Flow temperature sensor = disabled	14°C [other values in the range 14,5°C ... 20°C]
<i>Only displayed if the flow temperature from external sensor (from bus) is not available.</i>		
Hysteresis [K]	Anticondensation active = enaled	0,2 K / 0,3 K / 0,4 K / 0,5 / 0,6 K 0,8 K / 1 K / 1,5 K / 2 K / 2,5 K / 3 K
<i>Before quitting the alarm condition you must wait for the calculated dew temperature to be greater than the flow temperature value by an offset equal to the hysteresis value.</i>		
Auxiliary cooling		disabled / enabled
Disabled from bus	Auxiliary cooling = enabled	no / yes
<i>This parameter enables the activation and deactivation of the function through a telegram from a bus device with supervising function.</i>		
Offset from setpoint	Auxiliary cooling = enabled	0,2 K / 0,3 K / 0,4 K / 0,5 / 0,6 K 0,8 K / 1 K / 1,5 K / 2 K / 2,5 K / 3 K
Hysteresis ON/OFF [K]	Auxiliary cooling = enabled	0,2 K / 0,3 K / 0,4 K / 0,5 / 0,6 K 0,8 K / 1 K / 1,5 K / 2 K / 2,5 K / 3 K
Cyclic sending interval	Auxiliary cooling = enabled	hh:mm:ss (00:00:00)
<i>00:00:00 means that the cyclic sending is not enabled.</i>		
Ventilation for auxiliary cooling	Cooling type = floor radiant panels or ceiling radiant panels	disabled / enabled
<i>This option allows combining a high-inertial system as the floor radiant panels to a low-inertial one as the fan-coils.</i>		

Object name	Conditions	Dim.	Flags	DPT	Comm. Obj. No.
Comfort setpoint (cooling)		2 Byte	CRWTU-	[9.001] temperature (°C)	30
Standby setpoint (cooling)		2 Byte	CRWTU-	[9.001] temperature (°C)	32
Economy setpoint (cooling)		2 Byte	CRWTU-	[9.001] temperature (°C)	34
Building protection setpoint (cooling)		2 Byte	CRWTU-	[9.001] temperature (°C)	36

Cooling out command	Control type = 2 point hysteresis or PWM	1 Bit	CR-T--	[1.001] switch	39
Object name	Conditions	Dim.	Flags	DPT	Comm. Obj. No.
Cooling out command	Control type = continuous	1 Byte	CR-T--	[5.001] percentage (0..100%)	39
Auxiliary cooling output command	Auxiliary cooling = enabled	1 Bit	CR-T--	[1.001] switch	41
Auxiliary cooling enable	Auxiliary cooling = enabled, Disabled from bus = yes	1 Bit	C-W---	[1.003] enable	43
Anticondensation alarm	At least one anticondensation protection enabled	1 Bit	CR-T--	[1.005] alarm	66

About anticondensation protection function

The objective of this function is to prevent the condensation on the thermal exchange surfaces of the installation or building when cooling is working. This function is mainly used in systems with thermal exchange consisting in surface terminals such as for the floor and ceiling cooling radiant systems. In this case the hydraulic circuits contain refrigerated water; usually the latent loads (due to the increase of air humidity in the room) are handled by air-conditioning units and the temperature and humidity conditions are far from those that could cause condensation. If this is not done in a satisfactory manner, or in case of stop of the air-conditioning units, it is necessary to provide additional safety measures to prevent or restrict the accidental formation of condensation on cold surfaces.

From a general point of view, the anticondensation protection function can be realized:

- by installing a proper room anticondensation probe; when this is active, the hydraulic circuit closes down. It is a passive protection, because the intervention takes place when condensation has already started;
- by calculating the dew-point temperature and confronting it with the conveying fluid flow temperature. If the critical condition for condensation is approaching, you can intervene by closing down the hydraulic circuit or adjusting the mixing conditions of the conveying fluid. This is an active protection because the goal is to prevent the condensation.

Nr.	Type	Denomination	Description
1a	Passive	Anticondensation protection by probe (via bus)	The thermostat receives the information about condensation via bus from a different KNX device through communication object 27: Anticondensation (from bus) [DPT 1.001 switch].
2a	Active	Anticondensation protection with comparison between flow temperature (constant projected value, set as parameter on ETS and dewpoint temperature (calculated by the thermostat)	Software protection that intervenes by closing down the room cooling circuit when the flow temperature defined in the hydronic project (as set in the corresponding ETS parameter) is lower than dew-point temperature calculated by the room thermostat using temperature and relative humidity values. The communication object involved is 45: Cooling out command [DPT 1.001 switch].

2b	Active	Anticondensation protection with comparison between flow temperature (constant projected value, set as parameter on ETS and dewpoint temperature (calculated by the thermostat)	Software protection that intervenes by closing down the room cooling circuit when the actual measured flow temperature and received via bus from a different KNX device is lower than dew-point temperature calculated by the room thermostat using temperature and relative humidity values. The communication objects involved are 21 at input: Flow temperature (from bus) [DPT 9.001 temperature °C] and 45: Cooling out command [DPT 1.001 switch].
3	Active	Anticondensation protection with dew-point temperature sending over the bus and adjustment of the flow temperature	Software protection that foresees the sending on the bus of the dew-point temperature calculated by the room thermostat using temperature and relative humidity values to a KNX device capable of controlling the mixing condition of the conveying fluid for the cooling circuit. The regulation is performed by the KNX device receiving the dew-point temperature sent by the thermostat. The communication object involved is 59: Dew-point temperature [DPT 9.001 temperature °C].

Anticondensation protection modes

If an anticondensation sensor is used, it is necessary use a device provided with a potential-free signalling contact. It is possible to connect the signalling contact to an input channel of another KNX device, e.g. a pushbutton interface or a binary input (External sensors (from bus) □ Anticondensation sensor = enabled). In this case the signal of the sensor is transmitted to the room temperature controller through the status of a communication object (case 1b of the table).

If the comparison between dew-point temperature calculated by the thermostat and flow temperature of the conveying fluid is used, there are 3 options:

- if the flow temperature value is not available (case 2a of the table), you can insert the value used in the project (parameter Flow temperature (projected));
- if the flow temperature value is available (case 2b of the table), you enable the Anticondensation Active parameter for comparison;
- if an bus actuator capable of intervention on the conveying fluid's mixing is available, the thermostat sends on the bus the calculated value of the dew-point temperature; the actuator compares this value with the flow temperature and, if necessary, modifies the mixing conditions in order to prevent the risk for condensation formation.

The proper anticondensation protection mode needs to be evaluated during the thermal plant design and depends on many factors such as type of building, continuity of service and desired comfort level, available KNX devices, and so on.

9.7 Main and auxiliary ventilation

The **Ventilation** folder includes the following parameters:

- Ventilation function
- Control type
- Threshold first speed [0,1 K]
- Threshold second speed [0,1 K]
- Threshold third speed [0,1 K]
- Speed control hysteresis [K]
- Proportional band [0,1 K]
- Minimum change of value to send [%]
- Hot start
- Min. temp.to start ventilation [°C]
- Disable ventilation from bus
- Signal from bus
- Fan start delay
- Fan stop delay

The conditions for the appearance of the **Ventilation** folder are:

Heating ☐ Type of heating = fan-coils or Type of cooling = fan-coils or a combination of the two conditions:

Heating ☐ Type of heating = floor radiant panels or ceiling radiant panels and Heating ☐ Ventilation ☐ Auxiliary heating = enabled

Cooling ☐ Type of cooling = floor radiant panels or ceiling radiant panels and Cooling ☐ Ventilation for auxiliary = enabled

This way two types of installations can be controlled: i) fan-coil terminals or ii) radiant panels as main stage and fan-coil terminals as auxiliary stage.

9.7.1 Parameter and communication object tables

Parameter name	Conditions	Values
Control type		1 speed 2 speeds 3 speeds continuous regulation
Threshold first speed [0,1 K]	Control type ≥ 1 speed	0 [range 0 ... 255]
	The value is represented in tenths of Kelvin degrees. If the parameter Thermostat function = both heating and cooling, the threshold value is valid for both seasonal modes.	
Threshold second speed [0,1 K]	Control type ≥ 2 speeds	10 [range 0 ... 255]
	The value is represented in tenths of Kelvin degrees. If the parameter Thermostat function = both heating and cooling, the threshold value is valid for both seasonal modes. For a correct operation of the ventilation, Threshold second speed > Threshold first speed.	

Parameter name	Conditions	Values
----------------	------------	--------

Threshold third speed [0,1 K]	Control type = 3 speeds	20 [range 0 ... 255]
	<i>The value is represented in tenths of Kelvin degrees. If the parameter Thermostat function = both heating and cooling, the threshold value is valid for both seasonal modes. For a correct operation of the ventilation, Threshold third speed > Threshold second speed.</i>	
Speed control hysteresis [K]	Control type = 1, 2 or 3 speeds	0,3 K [other values in the range 0,2 K ... 3 K]
Proportional band [0,1 K]	Control type = continuous regulation	30 [range 0 ... 255]
	<i>The value is represented in tenths of Kelvin degrees. If the parameter Thermostat function = both heating and cooling, the threshold value is valid for both seasonal modes.</i>	
Min. change of value to send [%]	Control type = continuous regulation	10 [range 2 ... 40]
	<i>Please refer to the Control Algorithms chapter for further information about the meaning of this parameter.</i>	
Manual operation		not depending on the temperature depending on the temperature
	<i>If the parameter = not depending on the temperature, the fan speed set by the user is not changed even when the temperature setpoint is reached; if the parameter = depending on the temperature, the fan stops when the temperature setpoint is reached.</i>	
Hot start	Thermostat function = both heating and cooling, External sensors (from bus) ☐ coil temperature = enabled	no / yes
	<i>For carrying out the function must be enabled a sensor for measuring the temperature of the heat exchanger of the fan coil. To this purpose an external sensor (from bus) can be used.</i>	
Min. temp.to start ventilation [°C]	Hot start = yes	35 [range 28 ... 40]
	<i>If enabled, the function is active only in heating mode.</i>	
Antistratification function	External sensors (from bus) ☐ Antistratification temperature = enabled	disabled / enabled
	<i>For carrying out the function at least a sensor for measuring a second temperature value must be enabled at a different height than that of the room temperature controller. To this purpose an external sensor (from bus) can be used.</i>	
Antistratification temp. differential	Antistratification function = enabled	2 [K/m] [other values in the range 0,25 ... 4,00 K/m]
	<i>The DIN 1946 recommends a max temperature gradient of 2 K/m for rooms with standard height (between 2,70 and 3 m).</i>	
Hysteresis	Antistratification function = enabled	0,5 K [other values in the range 0,2 ... 3 K]
Disable ventilation from bus		no / yes

Parameter name	Conditions	Values
Signal from bus	Disable ventilation from bus = yes	not inverted inverted
Fan start delay		0 s [other values in the range 10 s ... 12 min]
	<i>It appears also if the hot-start function is active (through measuring of the conveying fluid temperature at the battery for the thermal exchange). The function is active in both seasonal modes (heating and cooling).</i>	
Fan stop delay		0 s [other values in the range 10 s ... 12 min]
	<i>The function allows prolonging the operation of the ventilator, dissipating in the room the residual heat or cool present in battery for the thermal exchange. The function is active in both seasonal modes (heating and cooling).</i>	
Cyclic sending interval		no sending [other values in the range 30 s ... 120 min]

Object name	Conditions	Dim.	Flags	DPT	Comm. Obj. No.
Fan continuous speed	Control type = continuous regulation	1 Byte	CR-T--	[5.001] percentage (0..100%)	44
Fan speed 1	Control type = 1, 2 o 3 speeds	1 Bit	CR-T--	[1.001] switch	45
Fan speed 2	Control type = 2 or 3 speeds	1 Bit	CR-T--	[1.001] switch	46
Fan speed 3	Control type = 3 speeds	1 Bit	CR-T--	[1.001] switch	47
Fan control disable	Disable ventilation from bus = yes	1 Bit	C-W---	[1.002] boolean	48
Fan manual speed		1 Byte	CRWTU-	[5.010] counter pulses (0...255)	51
Fan speed status		1 Byte	CR-T--	[5.010] counter pulses (0...255)	52

Fan manual active status		1 Bit	CRWTU-	[1.011] state	53
Fan manual speed percentage		1 Byte	CR-T--	[5.001] percentage	75
Fan manual speed off status		1 Bit	CR-T--	[1.011] state	76

9.7.2 Delayed fan start ("hot-start")

This function is used in case the fan forces in the room air passing through a heat exchange coil (as in the case of the terminals to the fan-coil). In the heating mode of operation, to avoid possible discomfort caused by the dispatch of cold air in the room, the room temperature controller does not start the fan until the fluid has not reached a sufficiently high temperature. This situation normally occurs at the first start or after long periods of inactivity. The function can be carried out by:

1. a temperature control (through a temperature sensor on the coil exchange battery);
2. a delayed start (function approximated);

In the first case the temperature of the heat conveying fluid is acquired at the exchange battery. The function then has an effective temperature control, but for the execution is necessary that the heat exchange coil is equipped with a sensor of minimum water temperature that acquires the temperature of the heat conveying fluid.

The effectiveness of the function depends on a field measurement of the time actually required to have sufficiently warm air from the terminal.

9.7.3 Antistratification function

This function is used in the case of heating systems with thermal exchange of convective type for rooms with height and volume much higher than usual (atriums, fitness facility, commercial buildings, etc.). Because of the natural convection - with warm air rising to the highest altitudes of the room - the phenomenon of air stratification occurs, with energy waste and discomfort for the occupants at the same time. The function opposes to the air stratification, forcing the warm air downwards.

The antistratification function requires:

- rooms of great height;
- availability of ventilation devices able to force the air movement downwards (opposed to the natural convective movement of warm air);
- measuring of the temperature at two heights through the installation of a second temperature sensor at an adequate height in order to measure the actual air stratification (the main room temperature controller is supposed to be installed at 1.5 m).

For rooms with ordinary height (2,70÷3,00 m) the DIN 1946 standard recommends not to exceed 2 K/m in order to have an adequate comfort; this gradient may be bigger in higher rooms.

9.7.4 2-stage configuration with fan-coils as auxiliary stage

The fan-coil units may be used both as a main stage and secondary stage. As main stage they can be combined only to radiators as auxiliary stage. If, however, the main stage is done with (floor or ceiling) radiant panels, the fan-coils can be used as auxiliary stage. In the latter case they work in automatic mode with a configurable offset with respect to the temperature setpoint for the main stage, and then carry out their compensation function while the main stage is brought in temperature with bigger inertia.

The **Ventilation** folder, that is unique, configures a main or a auxiliary stage depending on the settings choosed in the **Heating** and **Cooling** folders. Similarly, the display interface will act on manual / automatic and manual forcing of the only fan-coil.

A particular case occurs when a fan-coil unit works in a season as auxiliary stage and in the other one as main stage. It is for example the case of:

- a radiant panels system that works only for heating and has a fan-coil as auxiliary stage; the same fan-coil works as main stage for cooling;
- a radiator system that has a fan-coil as auxiliary stage for heating; the same fan coil unit functions as main stage for cooling.

In these cases with the configuration adopted, the following steps are necessary:

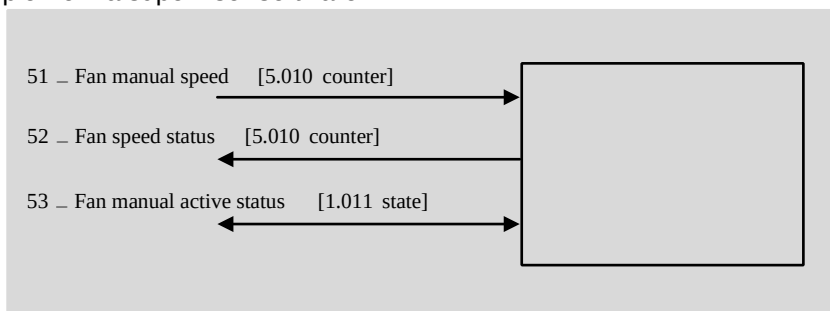
1. Settings □ Thermostat function = both heating and cooling. This configuration enables both folders (heating and cooling)
2. Heating □ Heating type = floor radiant panels or ceiling radiant panels
3. Heating □ Command communication object = separated (if unique is choosen, the parameter Cooling □ Cooling type does not appear)
4. Heating □ Auxiliary heating = enabled
5. Auxiliary heating □ Communication object = separated
6. Heating □ Ventilation for auxiliary heating = enabled
7. Cooling □ Cooling type = fancoils

Important! If the fan-coil system has a 2-pipe hydraulic configuration, the objects Auxiliary heating output command (1 bit) and Cooling out command (1 byte) have to be set in logical OR in the actuator for controlling the fan-coil which in this case is unique.

i An alternative solution that avoids the setting of a logic OR can be realized by configuring a main stage for heating and cooling with radiant panels through separate valves and an auxiliary stage for heating and cooling fan coil through combined valves. The offset of the auxialiary stage for cooling is set to the value 0 (zero); this corresponds to a configuration for main stage. The object Cooling out command (1 byte) is not connected so that the radiant panel system works only for heating.

9.7.5 Remote fan speed modification

The communication objects shown in figure allow to monitor actual fan speed forced automatically (A) by the temperature controller or set locally by the user when interacting with the LCD display and the touch buttons of the room thermostat. The communication objects (from now on: C.O.) also allow to perform the same modifications remotely, for example from a supervisor software.



The C.O. 52 – *Fan manual status* allows to evaluate the actual fan speed; the C.O. 53 – *Fan manual active status* contains the information about automatic (=0, not active) or manual (=1, active) operating mode. By modifying the C.O.

51 – *Fan manual speed*, the fan automatically switches to the setpoint speed; to return to automatic mode (A), the supervisor must exit from manual mode by modifying the C.O. 53 – *Fan manual active status* (=0, not active).

Accepted values for C.O.s 51 and 52 depend on the number of speeds set in ETS.

If *Control Type* parameter in Ventilation folder is = 1, 2 or 3 speeds, C.O.s with DPT [5.010 counter] accept the following values:

- = 0: OFF
- = 1: speed 1
- = 2: speed 2 (if *Control Type* > 1 speed)
- = 3: speed 3 (if *Control Type* > 2 speed)

If *Control Type* parameter in Ventilation folder is = continuous regulation, the values of the C.O.s with DPT [5.010 counter] match the following percentage of the maximum speed:

- = 0: OFF
- = 1: 20%
- = 2: 40%
- = 3: 60%
- = 4: 80%
- = 5: 100%

9.8 Scenes

The folder allows the scenes configuration (up to 8), assigning to each one an identification number and the operating mode to be activated when recalled (e.g. with an ekinex pushbutton or another KNX device with this function). If *Learning mode = enabled*, receiving a telegram of scene storage determines the association of the scene to the operating mode currently set on the Touch&See unit.

Important! Be careful setting the *Download overwrite* parameter. The download of the application program, particularly after the first commissioning of the system, may cause the loss of the already stored scenes.

The **Scenes** folder includes the following parameters:

- Download overwrite
- Scene X
- Scene number
- HVAC mode
- Activation delay
- Learning mode

Condition: *Temperature control* □ Settings □ Scenes = enabled.

9.8.1 Parameter and communication object tables

Parameter name	Conditions	Values
Download overwrite		disabled / enabled
	If Overwrite download = disabled: at the download of the application on the device, the operating modes previously stored are not overwritten. If Overwrite download = enabled: at the download of the application on the device, the operating modes previously stored are reprogrammed with the values selected from the HVAC mode parameter.	
Scene X		disabled / enabled
	This parameter enables the scene X (X = 1, 2, ... 8).	
Scene number	Scene X = enabled	1 [range 1 ... 64]
HVAC mode	Scene X = enabled	auto / comfort / standby / economy / building protection
	This parameter defines the operating mode of the scene X.	
Activation delay	Scene X = enabled	hh:mm:ss (00:00:00)
	Receiving a telegram that recalls a scene, once elapsed the time interval set in the Activation delay parameter, the programmed operating mode is activated.	
Learning mode	Scene X = enabled	disabled / enabled

Nome oggetto	Condizioni	Dim.	Flags	DPT	N° Ogg. Com.
HVAC scene number		1 Byte	C-W---	[17.001] scene number [18.001] scene control	54
It stores or recalls a scenario. The six least significant bits (0 to 5) in the byte of the code represent the number of the scene, while the most significant bit (7) is the operation code (1 = stores, 0 = recalls). 1 Byte <pre> graph TD subgraph "1 Byte" direction LR B7[7] --- B6[6] --- B5[5] --- B4[4] --- B3[3] --- B2[2] --- B1[1] --- B0[0] end B7 --- L1["0 = recall, 1 = save"] B6 --- L2["not used"] B5 --- L3["scene number (1-64)"] B4 --- L3 B3 --- L3 B2 --- L3 B1 --- L3 B0 --- L3 </pre>					

If the room thermostat is configured as slave (*General* folder ☐ *Device operation as = slave*) all scenes sent to the device have no effect: the device is therefore configured only to answer to the chrono program it is enslaved to.

i

If the room thermostat is configured as stand-alone, it is not possible to activate a scene that sets the device to automatic mode (*HVAC mode = auto*): all other HVAC modes are available to be activated by a scene.

If the room thermostat is configured as stand-alone/chrono, it is possible to activate scene that recall all HVAC modes, including Automatic mode (*ModoHVAC = auto*): the latter will enslave the room thermostat to the configured chrono program.

9.9 Relative humidity control

The **Relative humidity control** folder includes the following secondary folders:

- Dehumidification
- Humidification

The secondary folders **Dehumidification** and **Humidification** appear if a humidity sensor is enabled. The acquisition of relative humidity is made by bus from a KNX R.H. sensor.

The sensor acquires the air humidity value inside the room, which can be used for the following purposes:

- Local display and sending on the bus (for information purpose) through DPT [9.007] percentage (%);
- Use of detected value for derived psychrometric values calculations and sending on the bus through corresponding DPTs;
- Calculation of correlated indexes (perceived temperature) for local display or sending on the bus;
 - o Use for room ventilation through ventilation start, external intakes opening, window opening through motorized actuators. Control is performed upon thresholds;
- Use for control of thermoigrometric comfort conditions of radiant panel cooling systems equipped with integration of latent heat (starting of dedicated terminals without modification of cooling water flow temperature);
- Use for safety control in radiant panel cooling systems not equipped with integration of latent heat through calculation of critical thermoigrometric conditions (dew point) and corresponding modification of cooling water flow temperature.

NOTE: if the sensor is not able to read the RU value from bus, the value 7F FF is sent, until the real value is available.

9.9.1 Dehumidification

The secondary folder **Humidification** includes the following parameters:

- Operating modes where dehumidification is active
- Relative humidity setpoint for dehumidification control [%]
- Dehumidification control hysteresis [%]
- Dehumidification subordinated to temperature control
- Function of integration of sensible heat
- Disable from bus

9.9.1.1 Parameter and communication object tables

Parameter name	Conditions	Values
Dehumidification function		disabled cooling heating both cooling and heating
<i>Parameter that selects the dehumidification function.</i>		
Humidity setpoint [%]	Dehumidification function ≠ disabled	55 [range 30 ... 60]
Hysteresis	Dehumidification function ≠ disabled	0,8 % [other values in the range 0,5 ... 4%]
Subordinated to temperature control	Temperature control ☐ Settings ☐ Thermostat function = both heating and cooling, Temperature control ☐ cooling ☐ cooling type = floor radiant panels or ceiling radiant panels, Relative humidity ☐ dehumidification ☐ dehumidification function = cooling	no / yes
Dehumidification start delay	Subordinated to temperature control = no	00:05:00 hh:mm:ss [range 00:00:00 ... 18:12:15]
<i>Value 00:00:00 means that the start delay is disabled.</i>		
Integration		no / yes
Temperature difference for integration	Integration = yes	1,5°C [other values in the range 0,5 ... 3°C]
Hysteresis for integration	Integration = yes	0,5 K [other values in the range 0,2 K ... 3 K]
Cyclic sending interval		no sending [other values in the range 30 s ... 120 min]

Disable dehumid. control from bus		no / yes
Signal from bus	Disable dehumid. control from bus = yes	not inverted / inverted

Object name	Conditions	Dim.	Flags	DPT	Comm. Obj. No.
Relative humidity setpoint for dehumidification		2 Byte	CRWTU-	[9.007] humidity (%)	58
Dehumidification command		1 Bit	CR-T--	[1.001] switch	60
Dehumidification water battery command	Temperature control ☐ Settings ☐ Thermostat function = both heating and cooling, Temperature control ☐ cooling ☐ cooling type = floor radiant panels or ceiling radiant panels, Relative humidity ☐ dehumidification ☐ dehumidification function = cooling	1 Bit	CR-T--	[1.001] switch	61
Dehumidification integration control	Temperature control ☐ Settings ☐ Thermostat	1 Bit	CR-T--	[1.001] switch	62
Object name	Conditions	Dim.	Flags	DPT	Comm. Obj. No.
	function = both heating and cooling, Temperature control ☐ cooling ☐ cooling type = floor radiant panels or ceiling radiant panels, Relative humidity ☐ dehumidification ☐ dehumidification function = cooling				
	This object switches ON if (simultaneously) the relative humidity is greater than the relative humidity setpoint and the room temperature is greater than the setpoint of the parameter Temperature difference for integration.				
Dehumidification control disable		1 Bit	C-W---	[1.002] boolean	63

9.9.2 Humidification

The secondary folder **Humidification** includes the following parameters:

- Operating modes where humidification is active
- Relative humidity setpoint for humidification control [%]
- Dehumidification control hysteresis [%]
- Disable from bus

9.9.2.1 Parameter and communication object tables

Parameter name	Conditions	Values
Humidification function		disabled cooling heating both cooling and heating
<i>Parameter that selects the humidification function.</i>		
Humidity setpoint	Humidification ≠ disabled	35 [range 25 ... 45 %]
Humidity hysteresis [%]		0,8 % [other values in the range 0,5 ... 4%]
Cyclic sending interval	[other values in the range 30 s ... 120 min]	no sending
Disable humidification control from bus		no / yes
Signal from bus	Disable humidification control from bus = yes	not inverted / inverted

Object name	Conditions	Dim.	Flags	DPT	Comm. Obj. No.
Relative humidity setpoint for humidification		2 Byte	CRWTU-	[9.007] humidity (%)	59
Humidification command		1 Bit	CR-T--	[1.001] switch	64
Humidification control disable	Disable humidification control from bus = yes	1 Bit	C-W---	[1.002] boolean	65

9.10 Energy saving

In order to realize energy-saving functions, window contacts (to detect the opening of windows or doors), presence and movement sensors and card holders can be used.

The **Energy saving** folder includes the following secondary folders:

- Window contacts
- Presence sensors
- Card holder

9.10 .1 Window contacts

The **Window contacts** secondary folder appears if at least a sensor dedicated to this function is enabled i.e. if the following condition is verified: External sensors (from bus) □ Windows contact sensor 1 or 2 (from bus) = enabled

The **Window contacts** folder includes the following parameters:

- Window contacts function
- Wait time to building protection mode

9.10.1.1 Parameter and communication object tables

Parameter name	Conditions	Values
Window contacts function		disabled / enabled
	This parameter enables the window contact function.	
Wait time to building protection mode	Window contacts function = enabled	00:01:00 hh:mm:ss [range 00:00:00 ... 18:12:15]
	Time interval before the automatic switching of the device to the Building protection operating mode	

Nome oggetto	Conditions	Dim.	Flags	DPT	Comm. Obj. No.
Windows contact sensor 1 (from bus)	Window contacts function = enabled, Window contact 1 = enabled	1 Bit	C-W--	[1.019] window/door	13
Windows contact sensor 2 (from bus)	Window contacts function = enabled, Window contact 2 = enabled	1 Bit	C-W--	[1.019] window/door	14

9.10.2 Presence sensors

The **Presence sensors** folder includes the following parameters:

- Presence sensor function
- Presence sensors use
- Thermostat modes
- Absence time to switch HVAC mode

For this function only use external sensors (from bus) can be used, such as the ekinex EK-SM2-TP movement sensor or the ekinex EK-DX2-TP (X = B, C, D, E) presence sensor. The following condition has to be true:

External sensors (from bus) □ Presence sensor 1 (from bus) or Presence sensor 2 (from bus) = enabled

11.8.2.1 Parameter and communication object tables

Parameter name	Conditions	Values
Presence sensors function		disabled / enabled
<i>Parameter that enables the presence sensor function.</i>		
Presence sensors use	Presence sensor function = enabled	comfort extension comfort limitation comfort extension and comfort limitation
Thermostat modes	Presence sensor function = enabled, Presence sensors use = comfort extension and comfort limitation or = comfort limitation	comfort-standby comfort-economy
Absence time to switch HVAC mode	Presence sensor function = enabled	00:01:00 hh:mm:ss [range 00:00:00 ... 18:12:15]
<i>Time interval before the automatic switching of the operating mode set in the Thermostat modes parameter.</i>		

Object name	Conditions	Dim.	Flags	DPT	Comm. Obj. No.
Presence sensor 1 (from bus)	Presence sensor function=enabled	1 Bit	C-W---	[1.018] occupancy	15
Presence sensor 2 (from bus)	Presence sensor function = enabled	1 Bit	C-W---	[1.018] occupancy	16

9.10.3 Card holder

The **Card holder** secondary folder appears only if the corresponding sensor is enabled i.e. if the following condition is true: External sensors (from bus) □ Card holder contact = enabled

The **Card holder** folder includes the following parameters:

- Card holder function
- On card insertion switch HVAC mode to
- Activation delay on card insertion
- On card removal switch HVAC mode to
- Activation delay on card removal

9.10.3.1 Parameter and communication object tables

Parameter name	Conditions	Values
Card holder function		disabled / enabled
<i>Parameter that enables the card holder function.</i>		
On card insertion switch HVAC mode to	Card holder function = enabled	none comfort standby economy
<i>This parameter defines to which operating mode the device should automatically switch inserting the card into the holder.</i>		
Activation delay on card insertion	Card holder function = enabled	00:00:00 hh:mm:ss [range 00:00:00 ... 18:12:15]
<i>Time interval before the automatic switching of the operating mode, inserting the card into the holder.</i>		
On card removal switch HVAC mode to	Card holder function = enabled	none standby economy building protection
<i>This parameter defines to which operating mode the device should automatically switch removing the card from the holder.</i>		
Activation delay on card removal	Card holder function = enabled	00:00:00 hh:mm:ss [range 00:00:00 ... 18:12:15]
<i>Time interval before the automatic switching of the operating mode, removing the card from the holder.</i>		

Object name	Conditions	Dim.	Flags	DPT	Comm. Obj. No.
Card holder contact sensor (from bus)	Card holder function = enabled	1 Bit	CR-T--	[1.018] occupancy	17

- About card holder function

The information of card insertion (removal) in (from) a card holder allows you to directly control the temperature by means of the room thermostat, while sending the object value on the bus allows you to control other room functions with KNX (lighting, electrical loads, feedback status for the hotel reception, etc.) depending on the configuration done with ETS. The value of the setpoint temperature and the switching have to be defined with the hotel responsible in accordance with the target of energy saving and level of service to be offered to the guests.

- Conventional (not KNX) card holder

With a conventional card holder the status (card present or absent) of a signal contact is detected through an input of the device configured as *[DI] card holder contact sensor*. This way you can detect only the insertion and extraction of the card, but it cannot be detected e.g. the access of users with different profiles (guests, service staff, maintenance workforce).

- KNX card holder

With a KNX card holder you can differentiate the switching to be carried out; this is not resolved by the parameters of the room temperature controller, but through the definition of scenes that are received by the device. Depending on the available device, advanced functions are possible (e.g. different user profiles).

10 Logic functions

The pushbutton interfaces EK-CG2 allow to use some useful logic functions (AND, OR, NOT and exclusive OR) in order to implement complex functions in the building automation system.

You can configure:

- 4 channels of logical functions
- 4 inputs for each channel

Each object value, if desired, can be individually inverted by inserting a NOT logic operator.

For each channel, a parameter *Delay after bus voltage recovery* is available: this parameter represents the time interval between the bus voltage recovery and the first reading of the input communication objects for evaluating the logic functions.

i In case of uncorrect connection of the input communication object or electrical trouble on bus resulting in a failed input reading request, the logic output of the corresponding channel can be calculated by setting the input values to default.

The communication function representing the logic function output is sent on the bus on event of change; alternatively, a cyclic sending can be set.

10.1 Parameter and communication object tables

The following condition has to be true: *General* $\Rightarrow$ *Logic functions* = enabled.

Parameter name	Conditions	Values
Logic function		disabled / enabled
Logic operation	Logic function = enabled	OR / AND / XOR
	<i>XOR (eXclusive OR)</i>	
Delay after bus voltage recovery		00:00:04.000 hh:mm:ss.fff [range 00:00:00.000 ... 00:10:55.350]
	<i>Time interval between the bus voltage recovery and the first reading of the input communication objects for evaluating the logic functions</i>	
Output cyclic transmission delay		no sending [other value in range 30 s ... 120 min]
	<i>No sending means that the output state of the logic function is updated on the bus only on change. Different values imply cyclic sending on the bus of the output state.</i>	
Logic object x		disabled / enabled
Negated	Logic object x = enabled	no / yes
	<i>By denying the logic object state, it is possible to implement complex combinatorial logic. Example: Output = (NOT (logic object 1) OR logic object 2).</i>	
Read at startup	Logic object x = enabled	no / yes
Default value	Logic object x = enabled	none / off / on

Object name	Conditions	Dim.	Flags	DPT	Comm. Obj. No.
Logic function X – Input 1	Logic function X = enabled Logic object 1 = enabled	1 Bit	C-W--	[1.001] switch	204,209,214,219
Logic function X – Input 2	Logic function X = enabled Logic object 2 = enabled	1 Bit	C-W--	[1.001] switch	205,210,215,220
Logic function X – Input 3	Logic function X = enabled Logic object 3 = enabled	1 Bit	C-W--	[1.001] switch	206,211,216,221
Logic function X – Input 4	Logic function X = enabled Logic object 4 = enabled	1 Bit	C-W--	[1.001] switch	207,212,217,222
Logic function X – Output	Logic function X = enabled	1 Bit	C-W--	[1.001] switch	208,213,218,223

LED output configuration

The following configuration parameters are the same for all the available LEDs.
The LED settings are listed by input, disregard of the specific mode (single or coupled).

Parameter name	Condition	Value
LED X	-	fixed closed contact was from the
Always	LED X = fixed	on / off
Off Delay	LED X = closed contact	hh:mm:ss.ff (00:02:00.00)
	LED Power Off Delay After Power On Term	
Blinking	LED X = from bus	No /Yes
Signal from Bus	LED X = from bus	Not inverted / inverted
	Specifies whether the state of the LED received from the bus should be interpreted inverted, ie es. LED lit when receiving an "off" command via a communication object. This feature is useful because the LED on can be linked to a communication about the status of other entities, which have an opposite logic.	

Object name	Conditions	Dim.	Flags	DPT	Comm. Obj. No.
LED X –	LED = from bus	1 bit	CRWTU-	[1.001] switch	70,71,72,73

12 Appendix

12.1 Summary of KNX communication objects

The following list contains the KNX communication objects for all corresponding *Data Point Types* (DPT) defined by the application program according to the performed configurations.

The list is ordered by object number; if the same object is linked to different inputs, the first input or Input is referenced

Num	CO Name	Dimension	Flag	DataPoint Type
1	TechnicalAlarm	1 Bit	-WC---	[1.1] DPT_Switch
2	Input 1A - LockCommand	1 Bit	-WC---	[1.1] DPT_Switch
3	Input 1A - SwitchingStatus1	Variable(*)	RWCTU-	Variable(**)
4	Input 1A - SwitchingStatus2	Variable(*)	RWCTU-	Variable(**)
5	Input 1A - SwitchingStatus3	Variable(*)	RWCTU-	Variable(**)
6	Input 1A - SwitchingStatus4	Variable(*)	RWCTU-	Variable(**)
7	Input 1A - SwitchingStatus5	Variable(*)	RWCTU-	Variable(**)
8	Input 1A - SwitchingStatus6	Variable(*)	RWCTU-	Variable(**)
9	Input 1A - SwitchingStatus7	Variable(*)	RWCTU-	Variable(**)
10	Input 1A - SwitchingStatus8	Variable(*)	RWCTU-	Variable(**)
11	Input 1A - SwitchingCommand-DedicatedStop	1 Bit	-WCTU-	[1.1] DPT_Switch
12	Input 1A - DimmingUpDownStopCommand	4 Bit	R-CT--	[3.7] [3.8] DPT_Dimming
13	Dummy - Input 1A - InfoStatus	1 Bit	-WC---	[1.1] DPT_Switch
14	Input 1A - StopStepUpDownCommand	1 Bit	R-CT--	[1.17] DPT_Trigger
15	Input 1A - MoveUpDownCommand	1 Bit	RWCTU-	[1.8] DPT_Up/Down
16	Input 1A - SceneNumberControl	1 Bytes	--CT--	[17.1] [18.1] DPT_Scene
16	Input 1A – CounterValue 1 Byte	1 Bytes	R-CT--	[5.1] DPT_Pulses
16	Input 1A – CounterValue 2 Byte	2 Bytes	R-CT--	[7.1] DPT_Pulses
16	Input 1A – CounterValue 4 Byte	4 Bytes	R-CT--	[12.1] DPT_Counter pulses
17	Input 1A - Counter reset command	1 Bit	-WC-U-	[1.15] DPT_Reset
18	Input 1A - Counter runout	1 Bit	RWCTU-	[1.5] DPT_Alarm
19	Input 1B - LockCommand	1 Bit	-WC---	[1.3] DPT_Enable
20	Input 1B - SwitchingStatus1	Variable(*)	RWCTU-	Variable(**)
21	Input 1B - SwitchingStatus2	Variable(*)	RWCTU-	Variable(**)
22	Input 1B - SwitchingStatus3	Variable(*)	RWCTU-	Variable(**)
23	Input 1B - SwitchingStatus4	Variable(*)	RWCTU-	Variable(**)
24	Input 1B - SwitchingStatus5	Variable(*)	RWCTU-	Variable(**)
25	Input 1B - SwitchingStatus6	Variable(*)	RWCTU-	Variable(**)
26	Input 1B - SwitchingStatus7	Variable(*)	RWCTU-	Variable(**)
27	Input 1B - SwitchingStatus8	Variable(*)	RWCTU-	Variable(**)
28	Input 1B - SwitchingCommand-DedicatedStop	1 Bit	RWCTU-	[1.1] DPT_Switch

29	Input 1B - DimmingUpDownStopCommand	4 Bit	R-CT--	[3.7] [3.8] DPT_Dimming
30	Dummy - Input 1B - InfoStatus	1 Bit	-WC---	[1.1] DPT_Switch
31	Input 1B - StopStepUpDownCommand	1 Bit	R-CT--	[1.17] DPT_Trigger
32	Input 1B - MoveUpDownCommand	1 Bit	RWCTU-	[1.8] DPT_Up/Down
33	Input 1B - SceneNumberControl	1 Bytes	--CT--	[17.1] [18.1] DPT_Scene
33	Input 1B – CounterValue 1 Byte	1 Bytes	R-CT--	[5.1] DPT_Pulses
33	Input 1B – CounterValue 2 Byte	2 Bytes	R-CT--	[7.1] DPT_Pulses
33	Input 1B– CounterValue 4 Byte	4 Bytes	R-CT--	[12.1] DPT_Counter pulses
34	Input 1B - Counter reset command	1 Bit	-WC-U-	[1.15] DPT_Reset
35	Input 1B - Counter runout	1 Bit	RWCTU-	[1.5] DPT_Alarm
36	Input 2A - LockCommand	1 Bit	-WC---	[1.3] DPT_Enable
37	Input 2A - SwitchingStatus1	Variable(*)	RWCTU-	Variable(**)
38	Input 2A - SwitchingStatus2	Variable(*)	RWCTU-	Variable(**)
39	Input 2A - SwitchingStatus3	Variable(*)	RWCTU-	Variable(**)
40	Input 2A - SwitchingStatus4	Variable(*)	RWCTU-	Variable(**)
41	Input 2A - SwitchingStatus5	Variable(*)	RWCTU-	Variable(**)
42	Input 2A - SwitchingStatus6	Variable(*)	RWCTU-	Variable(**)
43	Input 2A - SwitchingStatus7	Variable(*)	RWCTU-	Variable(**)
44	Input 2A - SwitchingStatus8	Variable(*)	RWCTU-	Variable(**)
45	Input 2A - SwitchingCommand-DedicatedStop	1 Bit	RWCTU-	[1.1] DPT_Switch
46	Input 2A - DimmingUpDownStopCommand	4 Bit	R-CT--	[3.7] [3.8] DPT_Dimming
47	Dummy - Input 2A - InfoStatus	1 Bit	-WC---	[1.1] DPT_Switch
48	Input 2A - StopStepUpDownCommand	1 Bit	R-CT--	[1.17] DPT_Trigger
49	Input 2A - MoveUpDownCommand	1 Bit	RWCTU-	[1.8] DPT_Up/Down
50	Input 2A - SceneNumberControl	1 Bytes	--CT--	[17.1] [18.1] DPT_Scene
50	Input 2A – CounterValue 1 Byte	1 Bytes	R-CT--	[5.1] DPT_Pulses
50	Input 2A – CounterValue 2 Byte	2 Bytes	R-CT--	[7.1] DPT_Pulses
50	Input 2A – CounterValue 4 Byte	4 Bytes	R-CT--	[12.1] DPT_Counter pulses
51	Input 2A - Counter reset command	1 Bit	-WC-U-	[1.15] DPT_Reset
52	Input 2A - Counter runout	1 Bit	RWCTU-	[1.5] DPT_Alarm
53	Input 2B - LockCommand	1 Bit	-WC---	[1.3] DPT_Enable
54	Input 2B - SwitchingStatus1	Variable(*)	RWCTU-	Variable(**)
55	Input 2B - SwitchingStatus2	Variable(*)	RWCTU-	Variable(**)
56	Input 2B - SwitchingStatus3	Variable(*)	RWCTU-	Variable(**)
57	Input 2B - SwitchingStatus4	Variable(*)	RWCTU-	Variable(**)
58	Input 2B - SwitchingStatus5	Variable(*)	RWCTU-	Variable(**)
59	Input 2B - SwitchingStatus6	Variable(*)	RWCTU-	Variable(**)
60	Input 2B - SwitchingStatus7	Variable(*)	RWCTU-	Variable(**)
61	Input 2B - SwitchingStatus8	Variable(*)	RWCTU-	Variable(**)
62	Input 2B - SwitchingCommand-DedicatedStop	1 Bit	RWCTU-	[1.1] DPT_Switch
63	Input 2B - DimmingUpDownStopCommand	4 Bit	R-CT--	[3.7] [3.8] DPT_Dimming

64	Dummy - Input 2B - InfoStatus	1 Bit	-WC---	[1.1] DPT_Switch
65	Input 2B - StopStepUpDownCommand	1 Bit	R-CT--	[1.17] DPT_Trigger
66	Input 2B - MoveUpDownCommand	1 Bit	RWCTU-	[1.8] DPT_Up/Down
67	Input 2B - SceneNumberControl	1 Bytes	--CT--	[17.1] [18.1] DPT_Scene
67	Input 2B – CounterValue 1 Byte	1 Bytes	R-CT--	[5.1] DPT_Pulses
67	Input 2B – CounterValue 2 Byte	2 Bytes	R-CT--	[7.1] DPT_Pulses
67	Input 2B – CounterValue 4 Byte	4 Bytes	R-CT--	[12.1] DPT_Counter pulses
68	Input 2B - Counter reset command	1 Bit	-WC-U-	[1.15] DPT_Reset
69	Input 2B - Counter runout	1 Bit	RWCTU-	[1.5] DPT_Alarm
70	Led 1 - Command	1 Bit	RWCTU-	[1.1] DPT_Switch
71	Led 2 - Command	1 Bit	RWCTU-	[1.1] DPT_Switch
72	Led 3 - Command	1 Bit	RWCTU-	[1.1] DPT_Switch
73	Led 4 - Command	1 Bit	RWCTU-	[1.1] DPT_Switch
74	Logic function 1 - Input 1	1 Bit	-WCTU-	[1.1] DPT_Switch
75	Logic function 1 - Input 2	1 Bit	-WCTU-	[1.1] DPT_Switch
76	Logic function 1 - Input 3	1 Bit	-WCTU-	[1.1] DPT_Switch
77	Logic function 1 - Input 4	1 Bit	-WCTU-	[1.1] DPT_Switch
78	Logic function 1 - Output	1 Bit	R-CT--	[1.1] DPT_Switch
79	Logic function 2 - Input 1	1 Bit	-WCTU-	[1.1] DPT_Switch
80	Logic function 2 - Input 2	1 Bit	-WCTU-	[1.1] DPT_Switch
81	Logic function 2 - Input 3	1 Bit	-WCTU-	[1.1] DPT_Switch
82	Logic function 2 - Input 4	1 Bit	-WCTU-	[1.1] DPT_Switch
83	Logic function 2 - Output	1 Bit	R-CT--	[1.1] DPT_Switch
84	Logic function 3 - Input 1	1 Bit	-WCTU-	[1.1] DPT_Switch
85	Logic function 3 - Input 2	1 Bit	-WCTU-	[1.1] DPT_Switch
86	Logic function 3 - Input 3	1 Bit	-WCTU-	[1.1] DPT_Switch
87	Logic function 3 - Input 4	1 Bit	-WCTU-	[1.1] DPT_Switch
88	Logic function 3 - Output	1 Bit	R-CT--	[1.1] DPT_Switch
89	Logic function 4 - Input 1	1 Bit	-WCTU-	[1.1] DPT_Switch
90	Logic function 4 - Input 2	1 Bit	-WCTU-	[1.1] DPT_Switch
91	Logic function 4 - Input 3	1 Bit	-WCTU-	[1.1] DPT_Switch
92	Logic function 4 - Input 4	1 Bit	-WCTU-	[1.1] DPT_Switch
93	Logic function 4 - Output	1 Bit	R-CT--	[1.1] DPT_Switch
94	Thermostat 1 - Temperature value	2 Bytes	R-CT--	[9.1] DPT_Value_Temp
95	Thermostat 1 - Temperature threshold 1 - Switch	1 Bit	R-CT--	[1.1] DPT_Switch
96	Thermostat 1 - Temperature threshold 2 - Switch	1 Bit	R-CT--	[1.1] DPT_Switch
97	Thermostat 1 - Room temperature (from bus)	2 Bytes	-WCTU-	[9.1] DPT_Value_Temp
98	Thermostat 1 - Humidity value (2 bytes, from bus)	2 Bytes	-WCTU-	[9.7] DPT_Value_Humidity
99	Thermostat 1 - Humidity value (1 byte, from bus)	1 Byte	-WCTU-	[5.1] DPT_Scaling
100	Thermostat 1 - Antistratification temperature (from bus)	2 Bytes	-WCTU-	[9.1] DPT_Value_Temp

101	Thermostat 1 - Outdoor temperature (from bus)	2 Bytes	-WCTU-	[9.1] DPT_Value_Temp
102	Thermostat 1 - Coil temperature (from bus)	2 Bytes	-WCTU-	[9.1] DPT_Value_Temp
103	Thermostat 1 - Floor temperature (from bus)	2 Bytes	-WCTU-	[9.1] DPT_Value_Temp
104	Thermostat 1 - Flow temperature (from bus)	2 Bytes	-WCTU-	[9.1] DPT_Value_Temp
105	Thermostat 1 - Windows contact sensor 1 (from bus)	1 Bit	-WCTU-	[1.19] DPT_Window_Door
106	Thermostat 1 - Windows contact sensor 2 (from bus)	1 Bit	-WCTU-	[1.19] DPT_Window_Door
107	Thermostat 1 - Presence sensor 1 (from bus)	1 Bit	-WCTU-	[1.18] DPT_Occupancy
108	Thermostat 1 - Presence sensor 2 (from bus)	1 Bit	-WCTU-	[1.18] DPT_Occupancy
109	Thermostat 1 - Contact of card holder (from bus)	1 Bit	-WCTU-	[1.18] DPT_Occupancy
110	Thermostat 1 - Anticondensation (from bus)	1 Bit	-WCTU-	[1.1] DPT_Switch
111	Thermostat 1 - Weighted temperature	2 Bytes	R-CT--	[9.1] DPT_Value_Temp
112	Thermostat 1 - Heating/cooling status out	1 Bit	R-CT--	[1.100] DPT_Heat_Cool
113	Thermostat 1 - Heating/cooling status in	1 Bit	-WC---	[1.100] DPT_Heat_Cool
114	Thermostat 1 - HVAC mode in	1 Byte	-WC---	[20.102] DPT_HVACMode
115	Thermostat 1 - HVAC forced mode in	1 Byte	-WC---	[20.102] DPT_HVACMode
116	Thermostat 1 - HVAC mode out	1 Byte	R-CT--	[20.102] DPT_HVACMode
117	Thermostat 1 - HVAC manual mode	1 Byte	RWCTU-	[20.102] DPT_HVACMode
118	Thermostat 1 - Chrono status	1 Bit	R-CT--	[1.11] DPT_State
119	Thermostat 1 - Actual setpoint	2 Bytes	R-CT--	[9.1] DPT_Value_Temp
120	Thermostat 1 - Manual setpoint	2 Bytes	-WC---	[9.1] DPT_Value_Temp
121	Thermostat 1 - Comfort setpoint (heating)	2 Bytes	RWCTU-	[9.1] DPT_Value_Temp
121	Thermostat 1 - Setpoint in	2 Bytes	-WC---	[9.1] DPT_Value_Temp
122	Thermostat 1 - Comfort setpoint (cooling)	2 Bytes	RWCTU-	[9.1] DPT_Value_Temp
123	Thermostat 1 - Standby setpoint (heating)	2 Bytes	RWCTU-	[9.1] DPT_Value_Temp
123	Thermostat 1 - Offset standby (heating)	2 Bytes	RWCTU-	[9.2] DPT_Value_Tempd
124	Thermostat 1 - Standby setpoint (cooling)	2 Bytes	RWCTU-	[9.1] DPT_Value_Temp
124	Thermostat 1 - Offset standby (cooling)	2 Bytes	RWCTU-	[9.2] DPT_Value_Tempd
125	Thermostat 1 - Economy setpoint (heating)	2 Bytes	RWCTU-	[9.1] DPT_Value_Temp
125	Thermostat 1 - Offset economy (heating)	2 Bytes	RWCTU-	[9.2] DPT_Value_Tempd
126	Thermostat 1 - Economy setpoint (cooling)	2 Bytes	RWCTU-	[9.1] DPT_Value_Temp
126	Thermostat 1 - Offset economy (cooling)	2 Bytes	RWCTU-	[9.2] DPT_Value_Tempd
127	Thermostat 1 - Building protection setpoint (heating)	2 Bytes	RWCTU-	[9.1] DPT_Value_Temp
128	Thermostat 1 - Building protection setpoint (cooling)	2 Bytes	RWCTU-	[9.1] DPT_Value_Temp
129	Thermostat 1 - Room temperature controller status	1 Bit	R-CT--	[1.3] DPT_Enable
130	Thermostat 1 - Heating out command	1 Byte	R-CT--	[5.1] DPT_Scaling
130	Thermostat 1 - Heating out command 1bit	1 Bit	-WC---	[1.1] DPT_Switch
130	Thermostat 1 - Heating and Cooling out command 1 byte	1 Byte	R-CT--	[5.1] DPT_Scaling
130	Thermostat 1 - Heating and Cooling out command 1 bit	1 Bit	-WC---	[1.1] DPT_Switch

131	Thermostat 1 - Cooling out command	1 Byte	R-CT--	[5.1] DPT_Scaling
131	Thermostat 1 - Cooling out command 1 bit	1 Bit	-WC---	[1.1] DPT_Switch
132	Thermostat 1 - Auxiliary heating output command	1 Bit	R-CT--	[1.1] DPT_Switch
132	Thermostat 1 - Auxiliary heating and cooling output command	1 Bit	R-CT--	[1.1] DPT_Switch
133	Thermostat 1 - Auxiliary cooling output command	1 Bit	R-CT--	[1.1] DPT_Switch
134	Thermostat 1 - Auxiliary heating disable	1 Bit	-WC---	[1.3] DPT_Enable
135	Thermostat 1 - Auxiliary cooling disable	1 Bit	-WC---	[1.3] DPT_Enable
136	Thermostat 1 - Fan continuous speed	1 Byte	R-CT--	[5.1] DPT_Scaling
137	Thermostat 1 - Fan speed 1	1 Bit	R-CT--	[1.1] DPT_Switch
138	Thermostat 1 - Fan speed 2	1 Bit	R-CT--	[1.1] DPT_Switch
139	Thermostat 1 - Fan speed 3	1 Bit	R-CT--	[1.1] DPT_Switch
140	Thermostat 1 - Fan control disable	1 Bit	-WC---	[1.2] DPT_Bool
141	Dummy - Thermostat 1 - Manual mode lock	1 Bit	-WC---	[1.3] DPT_Enable
142	Thermostat 1 - Manual mode status	1 Bit	RWCTU-	[1.11] DPT_State
143	Thermostat 1 - Fan step manual speed	1 Byte	RWCTU-	[5.10] DPT_Value_1_Ucount
144	Thermostat 1 - Status fan step speed	1 Byte	R-CT--	[5.10] DPT_Value_1_Ucount
145	Thermostat 1 - Ventilation manual operation	1 Bit	R-CT--	[1.11] DPT_State
146	Thermostat 1 - HVAC scene number	1 Byte	-WC---	[17.1] DPT_SceneNumber, [18.1] DPT_SceneControl
147	Thermostat 1 - Disable room temperature controller	1 Bit	-WC---	[1.1] DPT_Switch
148	Thermostat 1 - Dew-point temperature	2 Bytes	R-CT--	[9.1] DPT_Value_Temp
149	Thermostat 1 - Relative humidity setpoint for dehumidification	2 Bytes	RWCTU-	[9.7] DPT_Value_Humidity
150	Thermostat 1 - Relative humidity setpoint for humidification	2 Bytes	RWCTU-	[9.7] DPT_Value_Humidity
151	Thermostat 1 - Dehumidification command	1 Bit	R-CT--	[1.1] DPT_Switch
152	Thermostat 1 - Dehumidification water battery command	1 Bit	R-CT--	[1.1] DPT_Switch
153	Thermostat 1 - Dehumidification integration control	1 Bit	R-CT--	[1.1] DPT_Switch
154	Thermostat 1 - Dehumidification control disable	1 Bit	-WC---	[1.2] DPT_Bool
155	Thermostat 1 - Humidification command	1 Bit	R-CT--	[1.1] DPT_Switch
156	Thermostat 1 - Humidification control disable	1 Bit	-WC---	[1.2] DPT_Bool
157	Thermostat 1 - Anticondensation alarm	1 Bit	R-CT--	[1.5] DPT_Alarm
158	Thermostat 1 - Thermal generator lock	1 Bit	-WC---	[1.5] DPT_Alarm
159	Thermostat 1 - HVAC building protection mode activated	1 Bit	R-CT--	[1.1] DPT_State
160	Thermostat 1 - Fan manual speed percentage	1 Byte	R-CT--	[5.1] DPT_Scaling
161	Thermostat 1 - Fan manual off status	1 Bit	R-CT--	[1.1] DPT_State
162	Thermostat 2 - Temperature value	2 Bytes	R-CT--	[9.1] DPT_Value_Temp
163	Thermostat 2 - Temperature threshold 1 - Switch	1 Bit	R-CT--	[1.1] DPT_Switch
164	Thermostat 2 - Temperature threshold 2 - Switch	1 Bit	R-CT--	[1.1] DPT_Switch
165	Thermostat 2 - Room temperature (from bus)	2 Bytes	-WCTU-	[9.1] DPT_Value_Temp

166	Thermostat 2 - Humidity value (2 bytes, from bus)	2 Bytes	-WCTU-	[9.7] DPT_Value_Humidity
167	Thermostat 2 - Humidity value (1 byte, from bus)	1 Byte	-WCTU-	[5.1] DPT_Scaling
168	Thermostat 2 - Antistratification temperature (from bus)	2 Bytes	-WCTU-	[9.1] DPT_Value_Temp
169	Thermostat 2 - Outdoor temperature (from bus)	2 Bytes	-WCTU-	[9.1] DPT_Value_Temp
170	Thermostat 2 - Coil temperature (from bus)	2 Bytes	-WCTU-	[9.1] DPT_Value_Temp
171	Thermostat 2 - Floor temperature (from bus)	2 Bytes	-WCTU-	[9.1] DPT_Value_Temp
172	Thermostat 2 - Flow temperature (from bus)	2 Bytes	-WCTU-	[9.1] DPT_Value_Temp
173	Thermostat 2 - Windows contact sensor 1 (from bus)	1 Bit	-WCTU-	[1.19] DPT_Window_Door
174	Thermostat 2 - Windows contact sensor 2 (from bus)	1 Bit	-WCTU-	[1.19] DPT_Window_Door
175	Thermostat 2 - Presence sensor 1 (from bus)	1 Bit	-WCTU-	[1.18] DPT_Occupancy
176	Thermostat 2 - Presence sensor 2 (from bus)	1 Bit	-WCTU-	[1.18] DPT_Occupancy
177	Thermostat 2 - Contact of card holder (from bus)	1 Bit	-WCTU-	[1.18] DPT_Occupancy
178	Thermostat 2 - Anticondensation (from bus)	1 Bit	-WCTU-	[1.1] DPT_Switch
179	Thermostat 2 - Weighted temperature	2 Bytes	R-CT--	[9.1] DPT_Value_Temp
180	Thermostat 2 - Heating/cooling status out	1 Bit	R-CT--	[1.100] DPT_Heat_Cool
181	Thermostat 2 - Heating/cooling status in	1 Bit	-WC---	[1.100] DPT_Heat_Cool
182	Thermostat 2 - HVAC mode in	1 Byte	-WC---	[20.102] DPT_HVACMode
183	Thermostat 2 - HVAC forced mode in	1 Byte	-WC---	[20.102] DPT_HVACMode
184	Thermostat 2 - HVAC mode out	1 Byte	R-CT--	[20.102] DPT_HVACMode
185	Thermostat 2 - HVAC manual mode	1 Byte	RWCTU-	[20.102] DPT_HVACMode
186	Thermostat 2 - Chrono status	1 Bit	R-CT--	[1.11] DPT_State
187	Thermostat 2 - Actual setpoint	2 Bytes	R-CT--	[9.1] DPT_Value_Temp
188	Thermostat 2 - Manual setpoint	2 Bytes	-WC---	[9.1] DPT_Value_Temp
189	Thermostat 2 - Comfort setpoint (heating)	2 Bytes	RWCTU-	[9.1] DPT_Value_Temp
189	Thermostat 2 - Setpoint in	2 Bytes	-WC---	[9.1] DPT_Value_Temp
190	Thermostat 2 - Comfort setpoint (cooling)	2 Bytes	RWCTU-	[9.1] DPT_Value_Temp
191	Thermostat 2 - Standby setpoint (heating)	2 Bytes	RWCTU-	[9.1] DPT_Value_Temp
191	Thermostat 2 - Offset standby (heating)	2 Bytes	RWCTU-	[9.2] DPT_Value_Tempd
192	Thermostat 2 - Standby setpoint (cooling)	2 Bytes	RWCTU-	[9.1] DPT_Value_Temp
192	Thermostat 2 - Offset standby (cooling)	2 Bytes	RWCTU-	[9.2] DPT_Value_Tempd
193	Thermostat 2 - Economy setpoint (heating)	2 Bytes	RWCTU-	[9.1] DPT_Value_Temp
193	Thermostat 2 - Offset economy (heating)	2 Bytes	RWCTU-	[9.2] DPT_Value_Tempd
194	Thermostat 2 - Economy setpoint (cooling)	2 Bytes	RWCTU-	[9.1] DPT_Value_Temp
194	Thermostat 2 - Offset economy (cooling)	2 Bytes	RWCTU-	[9.2] DPT_Value_Tempd
195	Thermostat 2 - Building protection setpoint (heating)	2 Bytes	RWCTU-	[9.1] DPT_Value_Temp
196	Thermostat 2 - Building protection setpoint (cooling)	2 Bytes	RWCTU-	[9.1] DPT_Value_Temp
197	Thermostat 2 - Room temperature controller status	1 Bit	R-CT--	[1.3] DPT_Enable
198	Thermostat 2 - Heating out command	1 Byte	R-CT--	[5.1] DPT_Scaling

198	Thermostat 2 - Heating out command 1bit	1 Bit	-WC---	[1.1] DPT_Switch
198	Thermostat 2 - Heating and Cooling out command 1 byte	1 Byte	R-CT--	[5.1] DPT_Scaling
198	Thermostat 2 - Heating and Cooling out command 1 bit	1 Bit	-WC---	[1.1] DPT_Switch
199	Thermostat 2 - Cooling out command	1 Byte	R-CT--	[5.1] DPT_Scaling
199	Thermostat 2 - Cooling out command 1 bit	1 Bit	-WC---	[1.1] DPT_Switch
200	Thermostat 2 - Auxiliary heating output command	1 Bit	R-CT--	[1.1] DPT_Switch
200	Thermostat 2 - Auxiliary heating and cooling output command	1 Bit	R-CT--	[1.1] DPT_Switch
201	Thermostat 2 - Auxiliary cooling output command	1 Bit	R-CT--	[1.1] DPT_Switch
202	Thermostat 2 - Auxiliary heating disable	1 Bit	-WC---	[1.3] DPT_Enable
203	Thermostat 2 - Auxiliary cooling disable	1 Bit	-WC---	[1.3] DPT_Enable
204	Thermostat 2 - Fan continuous speed	1 Byte	R-CT--	[5.1] DPT_Scaling
205	Thermostat 2 - Fan speed 1	1 Bit	R-CT--	[1.1] DPT_Switch
206	Thermostat 2 - Fan speed 2	1 Bit	R-CT--	[1.1] DPT_Switch
207	Thermostat 2 - Fan speed 3	1 Bit	R-CT--	[1.1] DPT_Switch
208	Thermostat 2 - Fan control disable	1 Bit	-WC---	[1.2] DPT_Bool
209	Dummy - Thermostat 2 - Manual mode lock	1 Bit	-WC---	[1.3] DPT_Enable
210	Thermostat 2 - Manual mode status	1 Bit	RWCTU-	[1.11] DPT_State
211	Thermostat 2 - Fan step manual speed	1 Byte	RWCTU-	[5.10] DPT_Value_1_Ucount
212	Thermostat 2 - Status fan step speed	1 Byte	R-CT--	[5.10] DPT_Value_1_Ucount
213	Thermostat 2 - Ventilation manual operation	1 Bit	R-CT--	[1.11] DPT_State
214	Thermostat 2 - HVAC scene number	1 Byte	-WC---	[17.1] DPT_SceneNumber, [18.1] DPT_SceneControl
215	Thermostat 2 - Disable room temperature controller	1 Bit	-WC---	[1.1] DPT_Switch
216	Thermostat 2 - Dew-point temperature	2 Bytes	R-CT--	[9.1] DPT_Value_Temp
217	Thermostat 2 - Relative humidity setpoint for dehumidification	2 Bytes	RWCTU-	[9.7] DPT_Value_Humidity
218	Thermostat 2 - Relative humidity setpoint for humidification	2 Bytes	RWCTU-	[9.7] DPT_Value_Humidity
219	Thermostat 2 - Dehumidification command	1 Bit	R-CT--	[1.1] DPT_Switch
220	Thermostat 2 - Dehumidification water battery command	1 Bit	R-CT--	[1.1] DPT_Switch
221	Thermostat 2 - Dehumidification integration control	1 Bit	R-CT--	[1.1] DPT_Switch
222	Thermostat 2 - Dehumidification control disable	1 Bit	-WC---	[1.2] DPT_Bool
223	Thermostat 2 - Humidification command	1 Bit	R-CT--	[1.1] DPT_Switch
224	Thermostat 2 - Humidification control disable	1 Bit	-WC---	[1.2] DPT_Bool
225	Thermostat 2 - Anticondensation alarm	1 Bit	R-CT--	[1.5] DPT_Alarm
226	Thermostat 2 - Thermal generator lock	1 Bit	-WC---	[1.5] DPT_Alarm
227	Thermostat 2 - HVAC building protection mode activated	1 Bit	R-CT--	[1.1] DPT_State
228	Thermostat 2 - Fan manual speed percentage	1 Byte	R-CT--	[5.1] DPT_Scaling
229	Thermostat 2 - Fan manual off status	1 Bit	R-CT--	[1.1] DPT_State

230	Thermostat 3 - Temperature value	2 Bytes	R-CT--	[9.1] DPT_Value_Temp
231	Thermostat 3 - Temperature threshold 1 - Switch	1 Bit	R-CT--	[1.1] DPT_Switch
232	Thermostat 3 - Temperature threshold 2 - Switch	1 Bit	R-CT--	[1.1] DPT_Switch
233	Thermostat 3 - Room temperature (from bus)	2 Bytes	-WCTU-	[9.1] DPT_Value_Temp
234	Thermostat 3 - Humidity value (2 bytes, from bus)	2 Bytes	-WCTU-	[9.7] DPT_Value_Humidity
235	Thermostat 3 - Humidity value (1 byte, from bus)	1 Byte	-WCTU-	[5.1] DPT_Scaling
236	Thermostat 3 - Antistratification temperature (from bus)	2 Bytes	-WCTU-	[9.1] DPT_Value_Temp
237	Thermostat 3 - Outdoor temperature (from bus)	2 Bytes	-WCTU-	[9.1] DPT_Value_Temp
238	Thermostat 3 - Coil temperature (from bus)	2 Bytes	-WCTU-	[9.1] DPT_Value_Temp
239	Thermostat 3 - Floor temperature (from bus)	2 Bytes	-WCTU-	[9.1] DPT_Value_Temp
240	Thermostat 3 - Flow temperature (from bus)	2 Bytes	-WCTU-	[9.1] DPT_Value_Temp
241	Thermostat 3 - Windows contact sensor 1 (from bus)	1 Bit	-WCTU-	[1.19] DPT_Window_Door
242	Thermostat 3 - Windows contact sensor 2 (from bus)	1 Bit	-WCTU-	[1.19] DPT_Window_Door
243	Thermostat 3 - Presence sensor 1 (from bus)	1 Bit	-WCTU-	[1.18] DPT_Occupancy
244	Thermostat 3 - Presence sensor 2 (from bus)	1 Bit	-WCTU-	[1.18] DPT_Occupancy
245	Thermostat 3 - Contact of card holder (from bus)	1 Bit	-WCTU-	[1.18] DPT_Occupancy
246	Thermostat 3 - Anticondensation (from bus)	1 Bit	-WCTU-	[1.1] DPT_Switch
247	Thermostat 3 - Weighted temperature	2 Bytes	R-CT--	[9.1] DPT_Value_Temp
248	Thermostat 3 - Heating/cooling status out	1 Bit	R-CT--	[1.100] DPT_Heat_Cool
249	Thermostat 3 - Heating/cooling status in	1 Bit	-WC---	[1.100] DPT_Heat_Cool
250	Thermostat 3 - HVAC mode in	1 Byte	-WC---	[20.102] DPT_HVACMode
251	Thermostat 3 - HVAC forced mode in	1 Byte	-WC---	[20.102] DPT_HVACMode
252	Thermostat 3 - HVAC mode out	1 Byte	R-CT--	[20.102] DPT_HVACMode
253	Thermostat 3 - HVAC manual mode	1 Byte	RWCTU-	[20.102] DPT_HVACMode
254	Thermostat 3 - Chrono status	1 Bit	R-CT--	[1.11] DPT_State
255	Thermostat 3 - Actual setpoint	2 Bytes	R-CT--	[9.1] DPT_Value_Temp
256	Thermostat 3 - Manual setpoint	2 Bytes	-WC---	[9.1] DPT_Value_Temp
257	Thermostat 3 - Comfort setpoint (heating)	2 Bytes	RWCTU-	[9.1] DPT_Value_Temp
257	Thermostat 3 - Setpoint in	2 Bytes	-WC---	[9.1] DPT_Value_Temp
258	Thermostat 3 - Comfort setpoint (cooling)	2 Bytes	RWCTU-	[9.1] DPT_Value_Temp
259	Thermostat 3 - Standby setpoint (heating)	2 Bytes	RWCTU-	[9.1] DPT_Value_Temp
259	Thermostat 3 - Offset standby (heating)	2 Bytes	RWCTU-	[9.2] DPT_Value_Tempd
260	Thermostat 3 - Standby setpoint (cooling)	2 Bytes	RWCTU-	[9.1] DPT_Value_Temp
260	Thermostat 3 - Offset standby (cooling)	2 Bytes	RWCTU-	[9.2] DPT_Value_Tempd
261	Thermostat 3 - Economy setpoint (heating)	2 Bytes	RWCTU-	[9.1] DPT_Value_Temp
261	Thermostat 3 - Offset economy (heating)	2 Bytes	RWCTU-	[9.2] DPT_Value_Tempd
262	Thermostat 3 - Economy setpoint (cooling)	2 Bytes	RWCTU-	[9.1] DPT_Value_Temp
262	Thermostat 3 - Offset economy (cooling)	2 Bytes	RWCTU-	[9.2] DPT_Value_Tempd

263	Thermostat 3 - Building protection setpoint (heating)	2 Bytes	RWCTU-	[9.1] DPT_Value_Temp
264	Thermostat 3 - Building protection setpoint (cooling)	2 Bytes	RWCTU-	[9.1] DPT_Value_Temp
265	Thermostat 3 - Room temperature controller status	1 Bit	R-CT--	[1.3] DPT_Enable
266	Thermostat 3 - Heating out command	1 Byte	R-CT--	[5.1] DPT_Scaling
266	Thermostat 3 - Heating out command 1bit	1 Bit	-WC---	[1.1] DPT_Switch
266	Thermostat 3 - Heating and Cooling out command 1 byte	1 Byte	R-CT--	[5.1] DPT_Scaling
266	Thermostat 3 - Heating and Cooling out command 1 bit	1 Bit	-WC---	[1.1] DPT_Switch
267	Thermostat 3 - Cooling out command	1 Byte	R-CT--	[5.1] DPT_Scaling
267	Thermostat 3 - Cooling out command 1 bit	1 Bit	-WC---	[1.1] DPT_Switch
268	Thermostat 3 - Auxiliary heating output command	1 Bit	R-CT--	[1.1] DPT_Switch
268	Thermostat 3 - Auxiliary heating and cooling output command	1 Bit	R-CT--	[1.1] DPT_Switch
269	Thermostat 3 - Auxiliary cooling output command	1 Bit	R-CT--	[1.1] DPT_Switch
270	Thermostat 3 - Auxiliary heating disable	1 Bit	-WC---	[1.3] DPT_Enable
271	Thermostat 3 - Auxiliary cooling disable	1 Bit	-WC---	[1.3] DPT_Enable
272	Thermostat 3 - Fan continuous speed	1 Byte	R-CT--	[5.1] DPT_Scaling
273	Thermostat 3 - Fan speed 1	1 Bit	R-CT--	[1.1] DPT_Switch
274	Thermostat 3 - Fan speed 2	1 Bit	R-CT--	[1.1] DPT_Switch
275	Thermostat 3 - Fan speed 3	1 Bit	R-CT--	[1.1] DPT_Switch
276	Thermostat 3 - Fan control disable	1 Bit	-WC---	[1.2] DPT_Bool
277	Dummy - Thermostat 3 - Manual mode lock	1 Bit	-WC---	[1.3] DPT_Enable
278	Thermostat 3 - Manual mode status	1 Bit	RWCTU-	[1.11] DPT_State
279	Thermostat 3 - Fan step manual speed	1 Byte	RWCTU-	[5.10] DPT_Value_1_Ucount
280	Thermostat 3 - Status fan step speed	1 Byte	R-CT--	[5.10] DPT_Value_1_Ucount
281	Thermostat 3 - Ventilation manual operation	1 Bit	R-CT--	[1.11] DPT_State
282	Thermostat 3 - HVAC scene number	1 Byte	-WC---	[17.1] DPT_SceneNumber, [18.1] DPT_SceneControl
283	Thermostat 3 - Disable room temperature controller	1 Bit	-WC---	[1.1] DPT_Switch
284	Thermostat 3 - Dew-point temperature	2 Bytes	R-CT--	[9.1] DPT_Value_Temp
285	Thermostat 3 - Relative humidity setpoint for dehumidification	2 Bytes	RWCTU-	[9.7] DPT_Value_Humidity
286	Thermostat 3 - Relative humidity setpoint for humidification	2 Bytes	RWCTU-	[9.7] DPT_Value_Humidity
287	Thermostat 3 - Dehumidification command	1 Bit	R-CT--	[1.1] DPT_Switch
288	Thermostat 3 - Dehumidification water battery command	1 Bit	R-CT--	[1.1] DPT_Switch
289	Thermostat 3 - Dehumidification integration control	1 Bit	R-CT--	[1.1] DPT_Switch
290	Thermostat 3 - Dehumidification control disable	1 Bit	-WC---	[1.2] DPT_Bool
291	Thermostat 3 - Humidification command	1 Bit	R-CT--	[1.1] DPT_Switch
292	Thermostat 3 - Humidification control disable	1 Bit	-WC---	[1.2] DPT_Bool
293	Thermostat 3 - Anticondensation alarm	1 Bit	R-CT--	[1.5] DPT_Alarm

294	Thermostat 3 - Thermal generator lock	1 Bit	-WC---	[1.5] DPT_Alarm
295	Thermostat 3 - HVAC building protection mode activated	1 Bit	R-CT--	[1.1] DPT_State
296	Thermostat 3 - Fan manual speed percentage	1 Byte	R-CT--	[5.1] DPT_Scaling
297	Thermostat 3 - Fan manual off status	1 Bit	R-CT--	[1.1] DPT_State
298	Thermostat 4 - Temperature value	2 Bytes	R-CT--	[9.1] DPT_Value_Temp
299	Thermostat 4 - Temperature threshold 1 - Switch	1 Bit	R-CT--	[1.1] DPT_Switch
300	Thermostat 4 - Temperature threshold 2 - Switch	1 Bit	R-CT--	[1.1] DPT_Switch
301	Thermostat 4 - Room temperature (from bus)	2 Bytes	-WCTU-	[9.1] DPT_Value_Temp
302	Thermostat 4 - Humidity value (2 bytes, from bus)	2 Bytes	-WCTU-	[9.7] DPT_Value_Humidity
303	Thermostat 4 - Humidity value (1 byte, from bus)	1 Byte	-WCTU-	[5.1] DPT_Scaling
304	Thermostat 4 - Antistratification temperature (from bus)	2 Bytes	-WCTU-	[9.1] DPT_Value_Temp
305	Thermostat 4 - Outdoor temperature (from bus)	2 Bytes	-WCTU-	[9.1] DPT_Value_Temp
306	Thermostat 4 - Coil temperature (from bus)	2 Bytes	-WCTU-	[9.1] DPT_Value_Temp
307	Thermostat 4 - Floor temperature (from bus)	2 Bytes	-WCTU-	[9.1] DPT_Value_Temp
308	Thermostat 4 - Flow temperature (from bus)	2 Bytes	-WCTU-	[9.1] DPT_Value_Temp
309	Thermostat 4 - Windows contact sensor 1 (from bus)	1 Bit	-WCTU-	[1.19] DPT_Window_Door
310	Thermostat 4 - Windows contact sensor 2 (from bus)	1 Bit	-WCTU-	[1.19] DPT_Window_Door
311	Thermostat 4 - Presence sensor 1 (from bus)	1 Bit	-WCTU-	[1.18] DPT_Occupancy
312	Thermostat 4 - Presence sensor 2 (from bus)	1 Bit	-WCTU-	[1.18] DPT_Occupancy
313	Thermostat 4 - Contact of card holder (from bus)	1 Bit	-WCTU-	[1.18] DPT_Occupancy
314	Thermostat 4 - Anticondensation (from bus)	1 Bit	-WCTU-	[1.1] DPT_Switch
315	Thermostat 4 - Weighted temperature	2 Bytes	R-CT--	[9.1] DPT_Value_Temp
316	Thermostat 4 - Heating/cooling status out	1 Bit	R-CT--	[1.100] DPT_Heat_Cool
317	Thermostat 4 - Heating/cooling status in	1 Bit	-WC---	[1.100] DPT_Heat_Cool
318	Thermostat 4 - HVAC mode in	1 Byte	-WC---	[20.102] DPT_HVACMode
319	Thermostat 4 - HVAC forced mode in	1 Byte	-WC---	[20.102] DPT_HVACMode
320	Thermostat 4 - HVAC mode out	1 Byte	R-CT--	[20.102] DPT_HVACMode
321	Thermostat 4 - HVAC manual mode	1 Byte	RWCTU-	[20.102] DPT_HVACMode
322	Thermostat 4 - Chrono status	1 Bit	R-CT--	[1.11] DPT_State
323	Thermostat 4 - Actual setpoint	2 Bytes	R-CT--	[9.1] DPT_Value_Temp
324	Thermostat 4 - Manual setpoint	2 Bytes	-WC---	[9.1] DPT_Value_Temp
325	Thermostat 4 - Comfort setpoint (heating)	2 Bytes	RWCTU-	[9.1] DPT_Value_Temp
325	Thermostat 4 - Setpoint in	2 Bytes	-WC---	[9.1] DPT_Value_Temp
326	Thermostat 4 - Comfort setpoint (cooling)	2 Bytes	RWCTU-	[9.1] DPT_Value_Temp
327	Thermostat 4 - Standby setpoint (heating)	2 Bytes	RWCTU-	[9.1] DPT_Value_Temp
327	Thermostat 4 - Offset standby (heating)	2 Bytes	RWCTU-	[9.2] DPT_Value_Tempd
328	Thermostat 4 - Standby setpoint (cooling)	2 Bytes	RWCTU-	[9.1] DPT_Value_Temp
328	Thermostat 4 - Offset standby (cooling)	2 Bytes	RWCTU-	[9.2] DPT_Value_Tempd

329	Thermostat 4 - Economy setpoint (heating)	2 Bytes	RWCTU-	[9.1] DPT_Value_Temp
329	Thermostat 4 - Offset economy (heating)	2 Bytes	RWCTU-	[9.2] DPT_Value_Tempd
330	Thermostat 4 - Economy setpoint (cooling)	2 Bytes	RWCTU-	[9.1] DPT_Value_Temp
330	Thermostat 4 - Offset economy (cooling)	2 Bytes	RWCTU-	[9.2] DPT_Value_Tempd
331	Thermostat 4 - Building protection setpoint (heating)	2 Bytes	RWCTU-	[9.1] DPT_Value_Temp
332	Thermostat 4 - Building protection setpoint (cooling)	2 Bytes	RWCTU-	[9.1] DPT_Value_Temp
333	Thermostat 4 - Room temperature controller status	1 Bit	R-CT--	[1.3] DPT_Enable
334	Thermostat 4 - Heating out command	1 Byte	R-CT--	[5.1] DPT_Scaling
334	Thermostat 4 - Heating out command 1bit	1 Bit	-WC---	[1.1] DPT_Switch
334	Thermostat 4 - Heating and Cooling out command 1 byte	1 Byte	R-CT--	[5.1] DPT_Scaling
334	Thermostat 4 - Heating and Cooling out command 1 bit	1 Bit	-WC---	[1.1] DPT_Switch
335	Thermostat 4 - Cooling out command	1 Byte	R-CT--	[5.1] DPT_Scaling
335	Thermostat 4 - Cooling out command 1 bit	1 Bit	-WC---	[1.1] DPT_Switch
336	Thermostat 4 - Auxiliary heating output command	1 Bit	R-CT--	[1.1] DPT_Switch
336	Thermostat 4 - Auxiliary heating and cooling output command	1 Bit	R-CT--	[1.1] DPT_Switch
337	Thermostat 4 - Auxiliary cooling output command	1 Bit	R-CT--	[1.1] DPT_Switch
338	Thermostat 4 - Auxiliary heating disable	1 Bit	-WC---	[1.3] DPT_Enable
339	Thermostat 4 - Auxiliary cooling disable	1 Bit	-WC---	[1.3] DPT_Enable
340	Thermostat 4 - Fan continuous speed	1 Byte	R-CT--	[5.1] DPT_Scaling
341	Thermostat 4 - Fan speed 1	1 Bit	R-CT--	[1.1] DPT_Switch
342	Thermostat 4 - Fan speed 2	1 Bit	R-CT--	[1.1] DPT_Switch
343	Thermostat 4 - Fan speed 3	1 Bit	R-CT--	[1.1] DPT_Switch
344	Thermostat 4 - Fan control disable	1 Bit	-WC---	[1.2] DPT_Bool
345	Dummy - Thermostat 4 - Manual mode lock	1 Bit	-WC---	[1.3] DPT_Enable
346	Thermostat 4 - Manual mode status	1 Bit	RWCTU-	[1.11] DPT_State
347	Thermostat 4 - Fan step manual speed	1 Byte	RWCTU-	[5.10] DPT_Value_1_Ucount
348	Thermostat 4 - Status fan step speed	1 Byte	R-CT--	[5.10] DPT_Value_1_Ucount
349	Thermostat 4 - Ventilation manual operation	1 Bit	R-CT--	[1.11] DPT_State
350	Thermostat 4 - HVAC scene number	1 Byte	-WC---	[17.1] DPT_SceneNumber, [18.1] DPT_SceneControl
351	Thermostat 4 - Disable room temperature controller	1 Bit	-WC---	[1.1] DPT_Switch
352	Thermostat 4 - Dew-point temperature	2 Bytes	R-CT--	[9.1] DPT_Value_Temp
353	Thermostat 4 - Relative humidity setpoint for dehumidification	2 Bytes	RWCTU-	[9.7] DPT_Value_Humidity
354	Thermostat 4 - Relative humidity setpoint for humidification	2 Bytes	RWCTU-	[9.7] DPT_Value_Humidity
355	Thermostat 4 - Dehumidification command	1 Bit	R-CT--	[1.1] DPT_Switch
356	Thermostat 4 - Dehumidification water battery command	1 Bit	R-CT--	[1.1] DPT_Switch
357	Thermostat 4 - Dehumidification integration control	1 Bit	R-CT--	[1.1] DPT_Switch

358	Thermostat 4 - Dehumidification control disable	1 Bit	-WC---	[1.2] DPT_Bool
359	Thermostat 4 - Humidification command	1 Bit	R-CT--	[1.1] DPT_Switch
360	Thermostat 4 - Humidification control disable	1 Bit	-WC---	[1.2] DPT_Bool
361	Thermostat 4 - Anticondensation alarm	1 Bit	R-CT--	[1.5] DPT_Alarm
362	Thermostat 4 - Thermal generator lock	1 Bit	-WC---	[1.5] DPT_Alarm
363	Thermostat 4 - HVAC building protection mode activated	1 Bit	R-CT--	[1.1] DPT_State
364	Thermostat 4 - Fan manual speed percentage	1 Byte	R-CT--	[5.1] DPT_Scaling
365	Thermostat 4 - Fan manual off status	1 Bit	R-CT--	[1.1] DPT_State

Table A1. Dimensions and DPT for Communication Objects with Independent Inputs:

Dimens. (*)	DPT(**)
1	[1.001] switch
1 bit	[2.*] 1-bit controlled
2 bit	[4.*] character [5.*] 8-bit unsigned value [20.*] 1-byte
1 byte unsigned value	[4.*] character [5.*] 8-bit unsigned value [20.*] 1-byte
1 byte signed value	[6.*] 8-bit signed value
2 bytes unsigned value	[7.*] 2-byte unsigned value
2 bytes signed value	[8.*] 2-byte signed value
2 bytes float value	[9.*] 2-byte float value

13 Warning

- Installation, electrical connection, configuration and commissioning of the device can only be carried out by qualified personnel.

- Opening the housing of the device causes the immediate end of the warranty period.
- ekinex® KNX defective devices must be returned to the manufacturer at the following address:

EKINEX S.p.A. Via Novara 37, I-28010 Vaprio d'Agogna (NO) Italy.

13.1 Other information

- This application manual is aimed at installers, system integrators and planners
- For further information on the product, please contact the ekinex® technical support at the e-mail address: support@ekinex.com or visit the website www.ekinex.com
- KNX® and ETS® are registered trademarks of KNX Association cvba, Brussels

© EKINEX S.p.A. The company reserves the right to make changes to this documentation without notice