User manual

K-BUS® KNX Gateway for Tuya ZigBee_V1.2

BTMO-TY/00.1

BTMO-TY/00.2

KNX/EIB Home and Building Control System

Attentions

1. **Please keep devices away from strong magnetic field, high temperature, wet environment;**

2. **Do not fall the device to the ground or make them get hard impact;**

3. **Do not use wet cloth or volatile reagent to wipe the device;**

4. **Do not disassemble the devices.**

Contents

Chapter 1 Summary

KNX Gateway for Tuya ZigBee as the interface between KNX system and Tuya ZigBee system, realizing the connection between the Tuya ZigBee ecosystem and the KNX system. Based on the gateway, it can easily control the KNX devices with Tuya APP, and upload status information of KNX devices to Tuya platform, for monitoring and managing the devices conveniently.

This manual provides detailed technical information about the KNX Gateway for Tuya ZigBee, including installation and programming details, and explains how to use it in the practical examples. After setting up KNX Gateway for Tuya ZigBee on the ETS, you can smartly manage the KNX and Zigbee system products, add and use the devices through the supporting mobile APP——Tuya Smart.

KNX Gateway for Tuya ZigBee powered from KNX bus, and need a 12-30V DC auxiliary supply voltage. It is available to assign the physical address and configure the parameters by engineering design tools ETS with .knxprod (support edition ETS5.7 or higher).

The functions are summarized as followed:

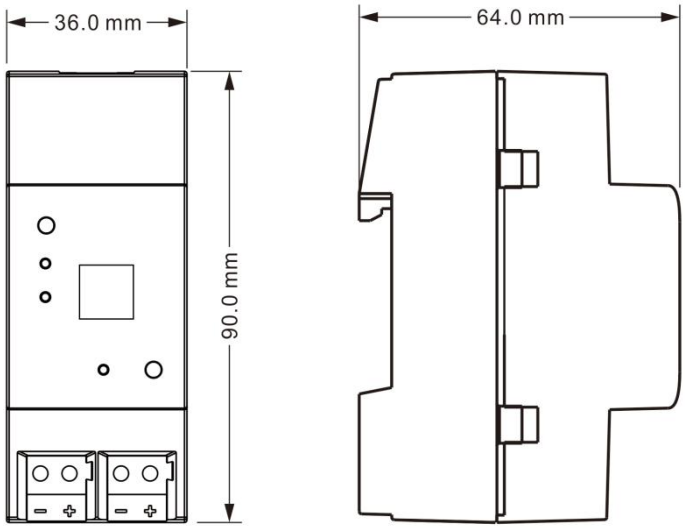
- As a ZigBee Gateway function, it can connect to Tuya Cloud, and add ZigBee device of Tuya ecosystem to the cloud
- Support uploading KNX devices to Tuya platform for management, to control KNX device and display status
- Support KNX devices functions, including Switch, Dimming, Curtains, Scenes, Colour and Colour temperature control, Audio control, HVAC control(Room temperature control, Air conditioner and Ventilation system), multiple sensors(Air Quality,Temperature and humidity, Brightness, Gas, I/O signal and etc.), and Current, Energy Metering display
- Logic functions

Chapter 2 Technical Data

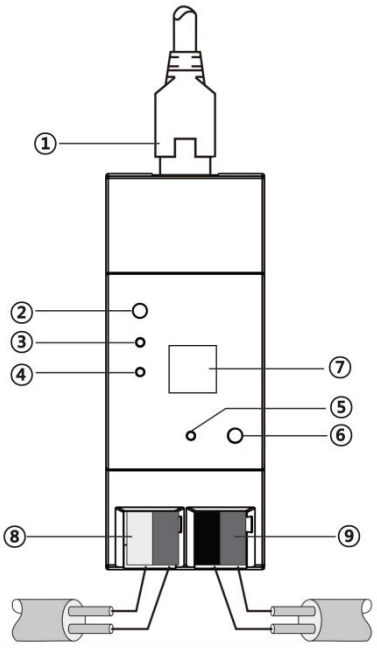
Power Supply	Bus voltage	21-30V DC，via the KNX bus
	Bus current	<4.5mA, 24V <4mA, 30V
	Bus consumption	<120mW
Auxiliary Supply	Voltage	12-30V DC
	Current	<60mA，24V <50mA，30V
	Consumption	<1.5W
Connection	KNX	Bus connection terminal(Red/Black)
	Auxiliary Supply	Bus connection terminal(Yellow/White)
	LAN	RJ45 socket for 100 Mbit and 10 Mbit BaseT, IEEE 802.3 networks
	Antenna	Rubber rod antenna or extension antenna
Operation and display	Connect button	Long press 5s will disconnect gateway
	Connect status LED	On: connecting; Off: connected Fast flashing: Tuya authorization code is abnormal(Unprogram)
	LAN LED	On: connected; Off: disconnected Flashing: Data communication
	Programming button and LED	Red on: assign physical address Green flashing: running normally
Temperature	Operation	– 5 °C ... 45 °C
	Storage	– 25 °C ... 55 °C
	Transport	– 25 °C ... 70 °C
Environment	Humidity	<93%, except dewing
Design	Standard 35mm DIN rail installation	
Dimension	36 x 90 x 64mm	
Weight	0.15kg	

Chapter 3 Dimension and Structural Diagram

3.1 Dimension Diagram



3.2 Structural Diagram



①LAN connection

②Connect button

③Connect status LED

④LAN LED

⑤Programming LED

⑥Programming button

⑦Antenna interface

⑧Auxiliary supply connection terminal

⑨KNX bus connection

Chapter 4 Project Design and Programming

Application	Maximum of communication objects	Maximum number of group addresses	Maximum number of associations
KNX Gateway for Tuya ZigBee/1.0	**3933**	**8000**	**8000**

General function

General function includes device In operation setting and IP setting.

KNX channel function

Support to upload the KNX device to the Tuya platform for management, which to control the device and display status. Up to support 150 devices. Each device can be customized its name with the engineering design tools ETS, and has In operation setting to monitor whether the device is online and upload to APP.

Top 100 device channels support the whole functions, including switch, dimming, curtain, scene switch, colour and colour temperature (RGB, RGBW and colour temperature), audio control, HVAC control (Room temperature control, Air conditioner and Ventilation), multiple sensors (Air quality, humidity and temperature, brightness, gas, I/O signal and etc.), and current, energy display.

The latter 50 device channels only support the functions, including switch, dimming, curtain, air quality sensor and energy display.

Note: Request each device status when the gateway power on.

Room temperature controller

Up to support 10 room temperature controllers. Mainly used to control the room temperature, Automatically and optimally cooling/heating control according to the room use or the occupants needs.

Support to manual switch to heating/cooling control, optionally three fan speeds and the auto fan speed, 4 operation modes: comfort, standby, economy and protection mode.

The temperature setpoint value supports the relative setting method, and is adjustable range setting. Support 2 points and PI control.

Ventilation controller

Support 1 ventilation controller, and fan speed can be linked to PM2.5/CO2/VOC, optionally output types of 1bit of 1byte.

Logic function

Up to support 8 channels of logic, each channel up to support 8 inputs and 1 logic result.

Logic function support functions, including AND, OR, XOR, Gate forwarding, Threshold comparator, Format convert, Gate function, Delay function and Staircase lighting.

Zigbee channel function (Reserves the function, not explained in this document)

Chapter 5 Parameter setting description in the ETS

5.1. Parameter window “General”

5.1.1. Parameter window “General setting”

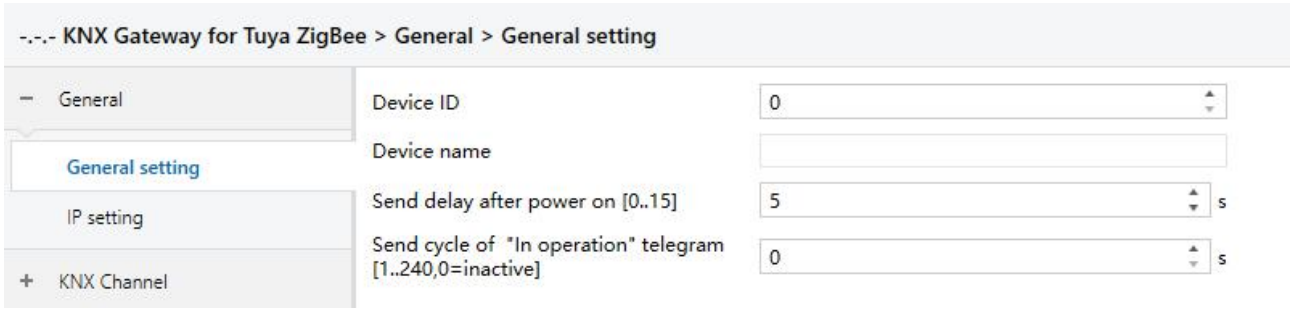


Fig.5.1.1 “General setting” Parameter window

Parameter “Device ID”

This parameter is for setting the device ID. Options: **0…65535**

Parameter “Device name”

This parameter is for setting the device name. Up to input 32 characters.

Parameter “Send delay after power on [0..15]s”

This parameter is for setting the delay time to send to bus after the gateway power on. Options: **0..15**

The setting dose not contain the gateway initialization time, and bus telegrams received during delay time will be recorded.

Parameter “Send cycle of “In operation” telegram [1...240s, 0 = inactive]”

This parameter is for setting the time interval when this device cycle send telegrams through the bus to indicate this module in normal operation. When set to “0”, the object “in operation” will not send a telegram. If the setting is not “0”, the object “In operation” will send a telegram according to the set period time with logic “1” to the bus. Options: **0...240s, 0= inactive**

As to reduce the bus load as much as possible, the maximum time interval should be selected according to actual needs.

5.1.2. Parameter window “IP setting”

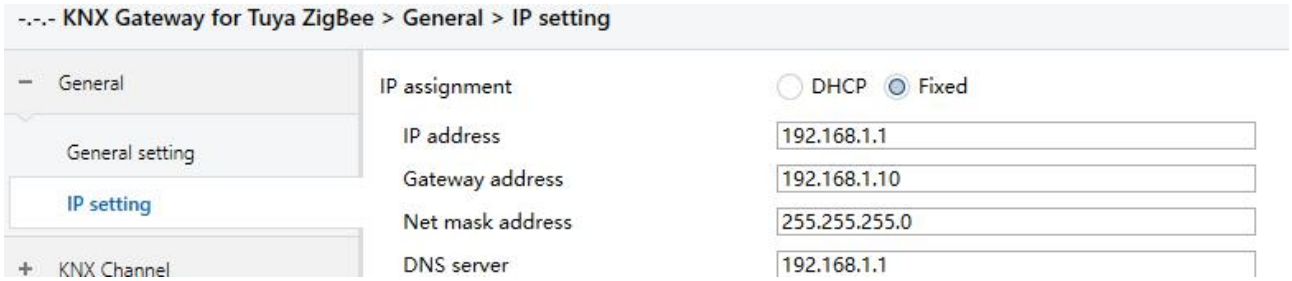


Fig.5.1.2 “IP setting” Parameter window

Parameter “IP assignment”

This parameter is for setting the assignment of IP address. Options:

DHCP

Fixed

Fixed: the assignment of IP address is fixed, can be assigned address for device via the parameters as follow.

DHCP: the parameters as follow are not visible when DHCP enabled, and no need to configure.

—Parameter “IP address”

This parameter is for setting the IP address of device, determines that this IP address is unique in the LAN, otherwise causing a conflict between IP.

Use string with IPv4 to input address, for example: 192.168.1.1

Options: **0-255.0-255.0-255.0-255**

—Parameter “Gateway address”

This parameter is for setting the gateway address of device, which must be in the same segment as the IP address.

Use string with IPv4 to input address, for example: 192.168.1.10

Options: **0-255.0-255.0-255.0-255**

—Parameter “Net mask address”

This parameter is for setting the net mask address of device, which must be must be set in a way that reflects the structure and number of subnets.

Use string with IPv4 to input address, for example: 255.255.255.0 (a simplest small network subnet)

Options: **0-255.0-255.0-255.0-255**

—Parameter “DNS server”

This parameter is for setting the DNS server of device.

Use string with IPv4 to input address, for example: 192.168.1.1

Options: **0-255.0-255.0-255.0-255**

5.2. Parameter window “KNX Channel ”

5.2.1. Parameter window “General setting”

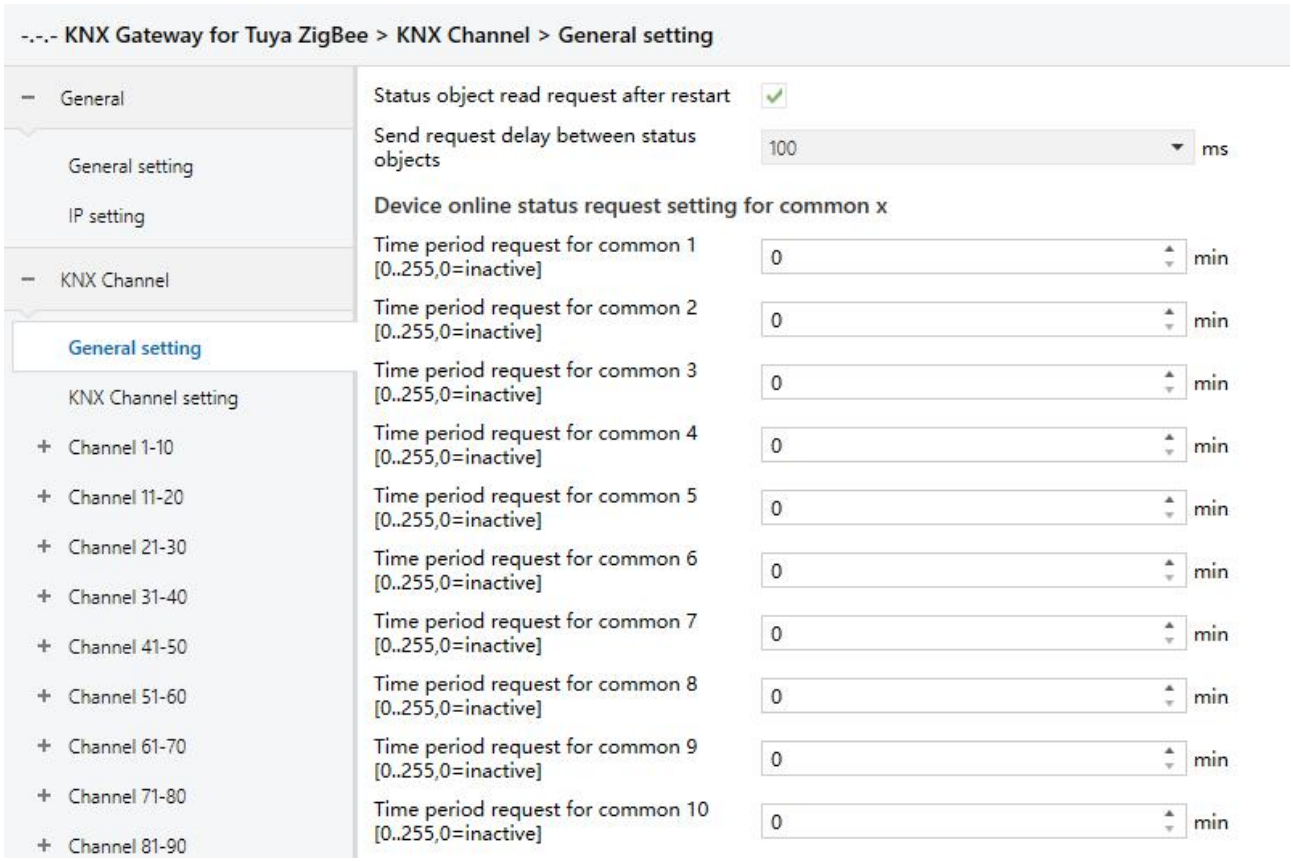


Fig.5.2.1 “General setting” Parameter window

Parameter “Status object read request after restart”

This parameter is for setting whether to send read request telegram of status after restart device.

——Parameter “Send request delay between status objects”

This parameter is visible when previous parameter is enabled. Set the interval time for sending between request telegrams when power on. Options:

50ms

100ms

200ms

Device online status request setting for common x

Parameter “Time period request for common x [0..255,0=inactive] min”(x=1~10)

This parameter is for setting the read request period to send to query the online status of the common device,

0 is not enable query function. Options: **0..255**

The function is used to query the online status of the KNX device. Requests begin after the gateway power-up send delay time complete.

5.2.2. Parameter window “KNX Channel setting”

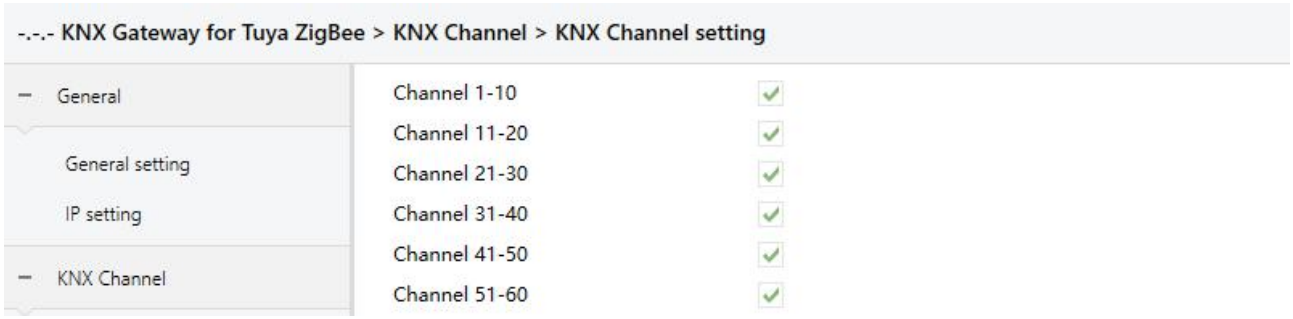


Fig.5.2.2(1) “KNX Channel setting” Parameter window

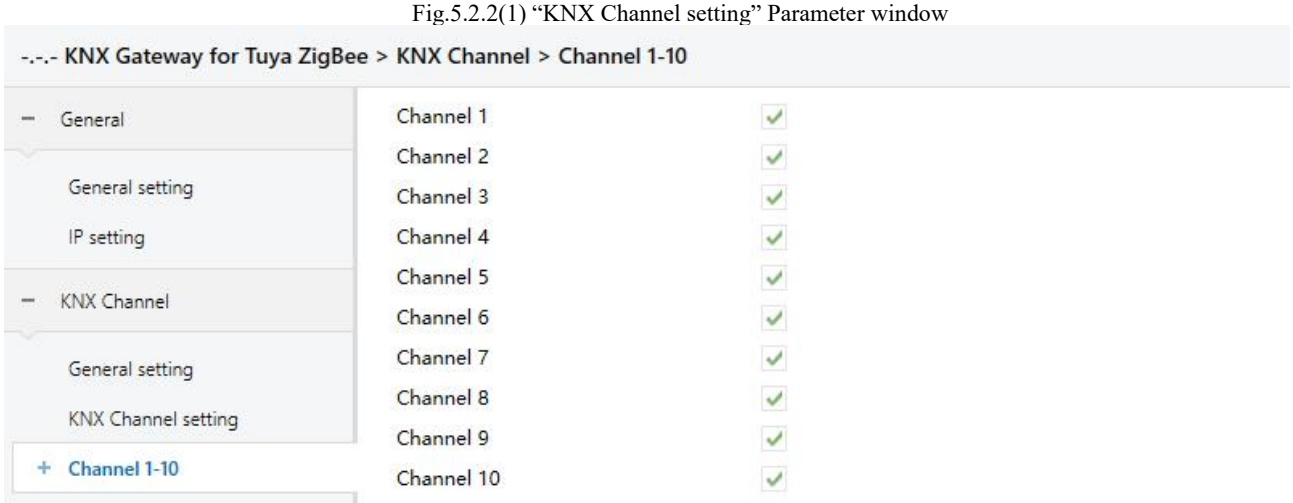


Fig.5.2.2(2) “Channel 1-10” Parameter window

Parameter “Channel 1-10/11-20/21-30/...”

This parameter is for setting the enable KNX device. Display 10 KNX channels per page when enabled. Up to support 150 channels.

Parameter “Channel 1/2/3/...”

This parameter is for setting the KNX device setting. Display corresponding window when enabled.

5.2.3. Parameter window “Channel x”(x=1~150)

Fig.5.2.3 “Channel 1” Parameter window

Parameters as follow is general setting parameters for KNX device type. Later chapters will not be repeated.

Parameter “Device type”

This parameter is for setting the device type of KNX channel. Options:

Switch
Switch/Dimming
RGB dimming
RGBW dimming
Color temperature
Curtain step/move
Roller blind step/move
Curtain position
Roller blind position
Venetian blind position and slat
Value sender
Scene switch
Air conditioner
Air conditioner(with swing)
Room temperature unit
Room temperature unit(with operation mode)
Room temperature unit(with operation mode & fan speed)
Ventilation system
Audio control
Audio control(with on/off)
Air quality sensor
CO2 sensor
PM2.5 sensor
VOC sensor
Presence sensor
Motion sensor
Brightness sensor
I/O signal
Current metering
Energy metering

Note: Channel 1~100 support all of the above function options, but channel 101~150 only support the functions as follow:

Switch
Switch/Dimming
Curtain step/move
Roller blind step/move
Air quality sensor
Energy metering

Parameter "Description (max 30char.)"

This parameter is for setting the name description for current channel device, up to input 30 characters.

Parameter "Device online status reference by"

This parameter is for setting the reference type sending read request to KNX device, you can select request for individual or common device, and optionally 10 common device requests. Options:

Common 1

...

Common 10

Individual

Always online

When select "Common ...", common device request, apply to the device with multiple circuits. For example, multiple channels of gateway may be multiple circuits that control the same KNX device, so each channel can share one request.

When select "Individual", apply to a device only is controlled by gateway single channel.

When select "Always online", apply to the KNX device without heartbeat pack, especially scene. That is, once configure the device, it will always online.

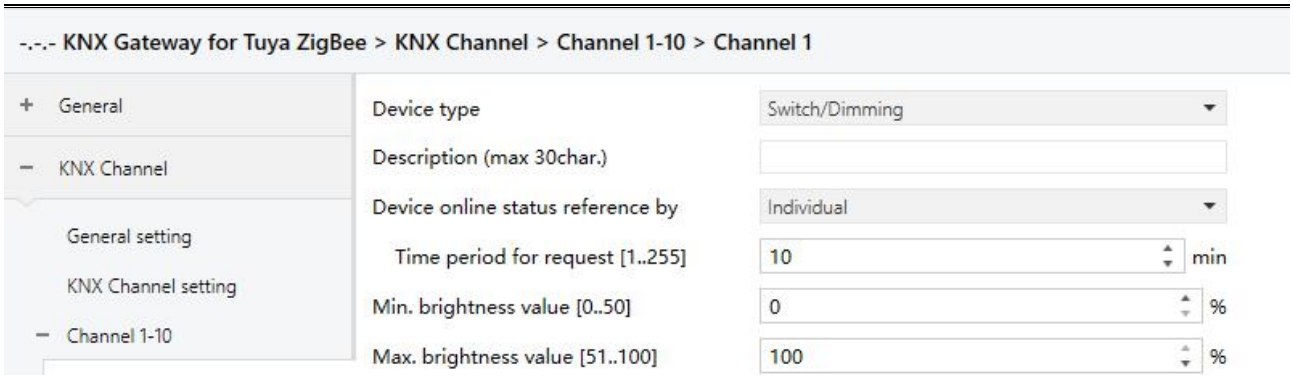
——Parameter "Time period for request [1..255] min"

This parameter is visible when "Individual" is selected, set the time period for the online status request of single device. Options: **1..255**

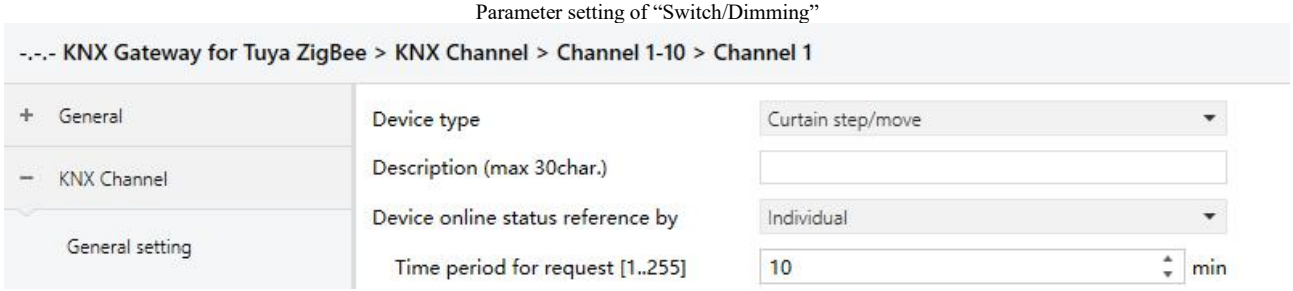
5.2.1.1. Parameter of basic function

This chapter explains the basic functions parameters of KNX channel, including switch, dimming, curtain, colour, colour temperature, value sender and KNX scene switch.

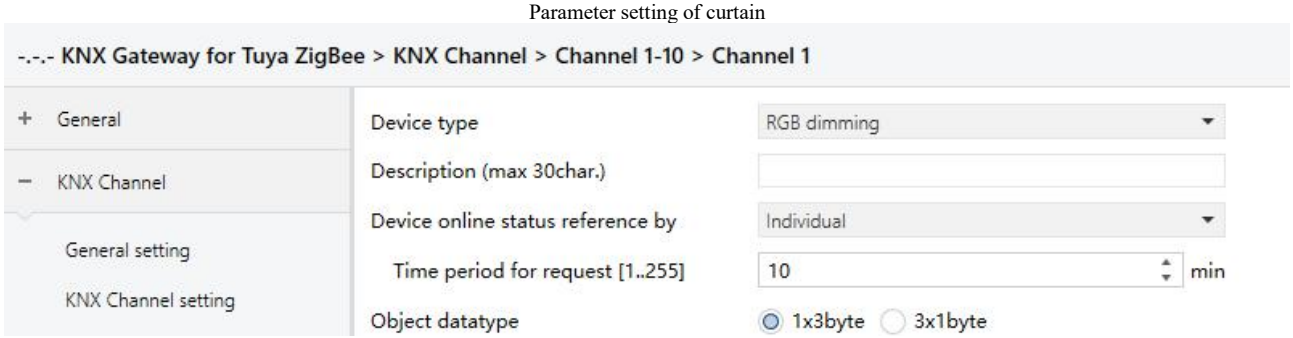
Parameter setting of "Switch"



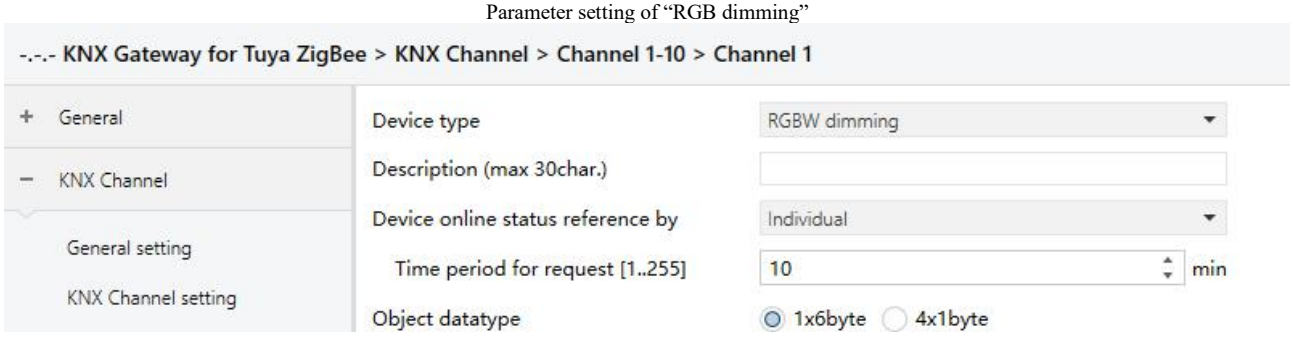
Parameter setting of "Switch/Dimming"



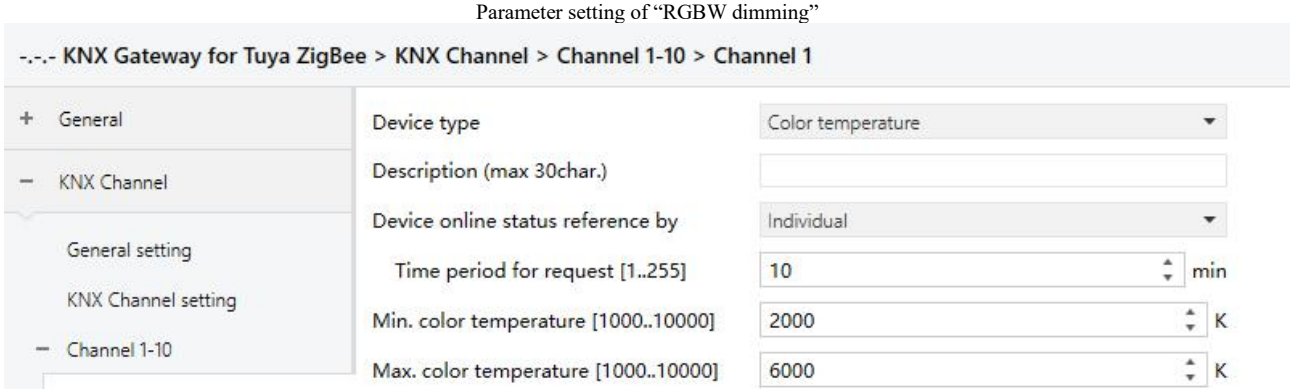
Parameter setting of curtain



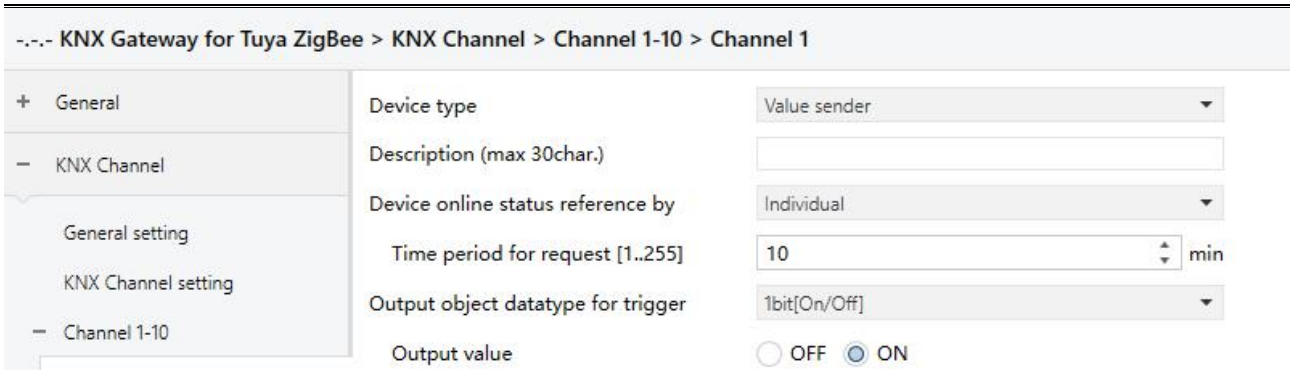
Parameter setting of "RGB dimming"



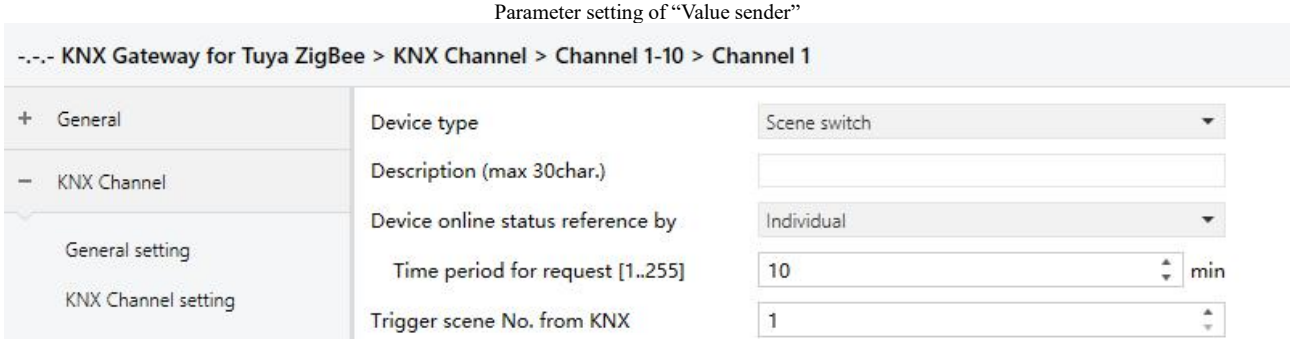
Parameter setting of "RGBW dimming"



Parameter setting of "Colour temperature"



Parameter setting of "Value sender"



Parameter setting of "Scene switch"

Fig.5.2.1.1 Parameter setting of basic function

Parameter "Min. brightness value [0..50]%"

Parameter "Max. brightness value [51..100]%"

These two parameters are visible when device type is selected "Switch/Dimming". Set the upper and lower limit threshold value of brightness.

The lower limit threshold value options: **0..50**; the upper limit threshold value options: **51..100**

Parameter "Object datatype"

This parameter is visible when device type is selected "RGB dimming" or "RGBW dimming". Set the object datatype of RGB or RGBW dimming.

Suitable for RGB type:

1x3byte

3x1byte

Suitable for RGBW type:

1x6byte

4x1byte

Parameter "Min. color temperature [1000..10000]K"

Parameter "Max. color temperature [1000..10000]K"

These two parameters are visible when device type is selected "Color temperature". Set the upper and lower

limit threshold value of colour temperature. Options: **1000..10000**

When the minimum value and maximum value are set incorrect, The selected range is the whole range, for example the minimum value is larger than the maximum value. There is only one value when equal.

Parameter "Output object datatype for trigger"

This parameter is visible when device type is selected "Value sender". Set telegram type for sending to KNX bus when trigger calling command on the APP. Options:

1bit[On/Off]

2bit[0..3]

1byte[0..100%]

1byte[0..255]

1byte[scene control]

2byte[Float]

2byte[0..65535]

Parameter "Output value"

This parameter is visible when device type is selected "Value sender", and corresponding datatype is selected. Set the telegram value for sending to KNX bus when trigger calling command on the APP. Options are according to the datatype:

OFF ON(1bit) / 0..3(2bit) / 0..100(1byte) / 0..255(1byte) / 1..64(1byte) /

-671 088.64..670 760.96(2byte) / 0..65535(2byte)

Parameter "Trigger scene No. from KNX"

This parameter is visible when device type is selected "Scene switch". Set the scene calling command of KNX system to receive. Options:**1..64**

5.2.1.2. Parameter of air condition

This chapter explains the air condition function of KNX channel, including the basic and air conditioning control with swing. Parameters as follow are visible when “Air conditioner” or “Air conditioner(with swing)” is selected.

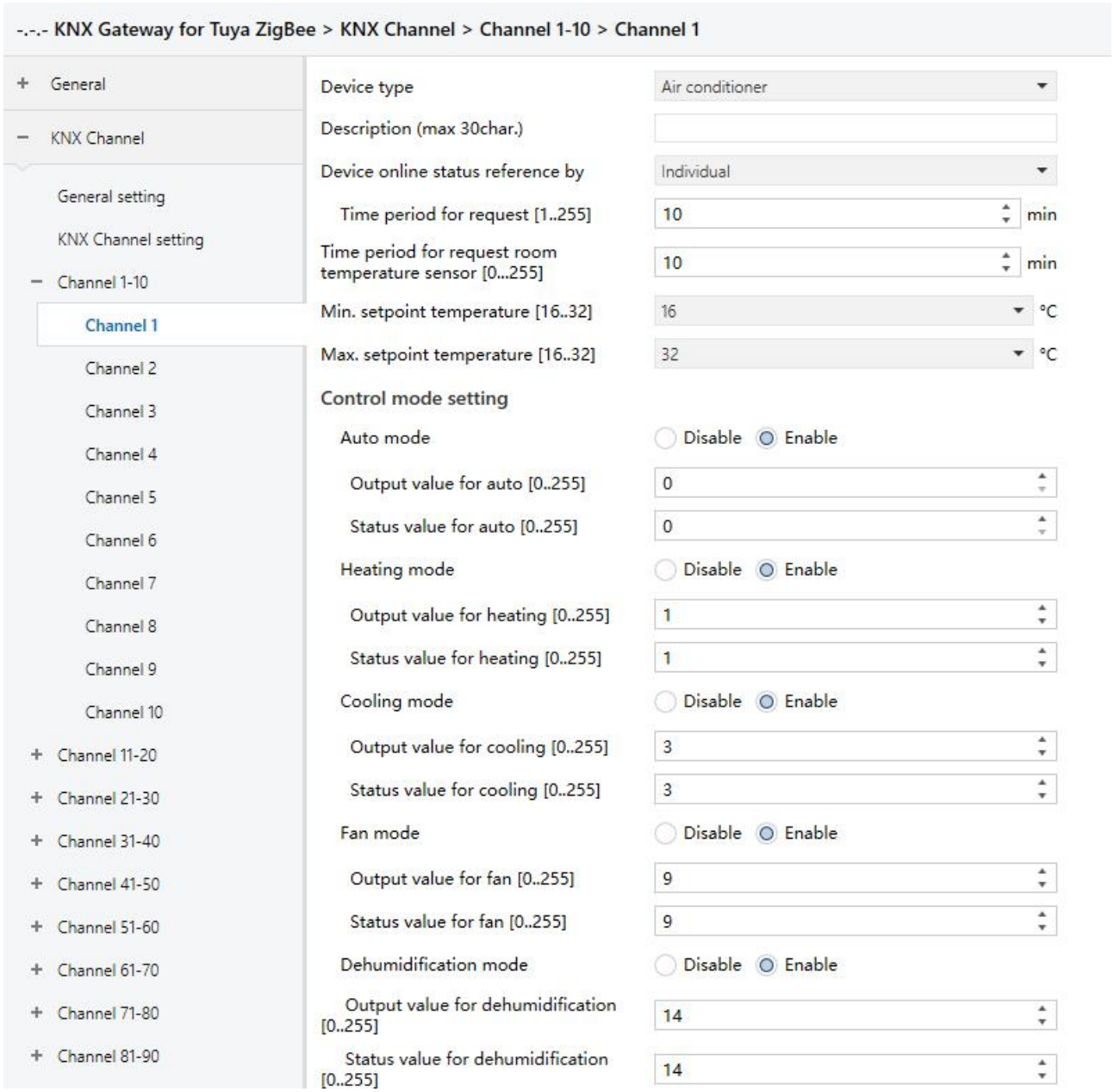


Fig.5.2.1.2(1) Parameter setting of air condition function

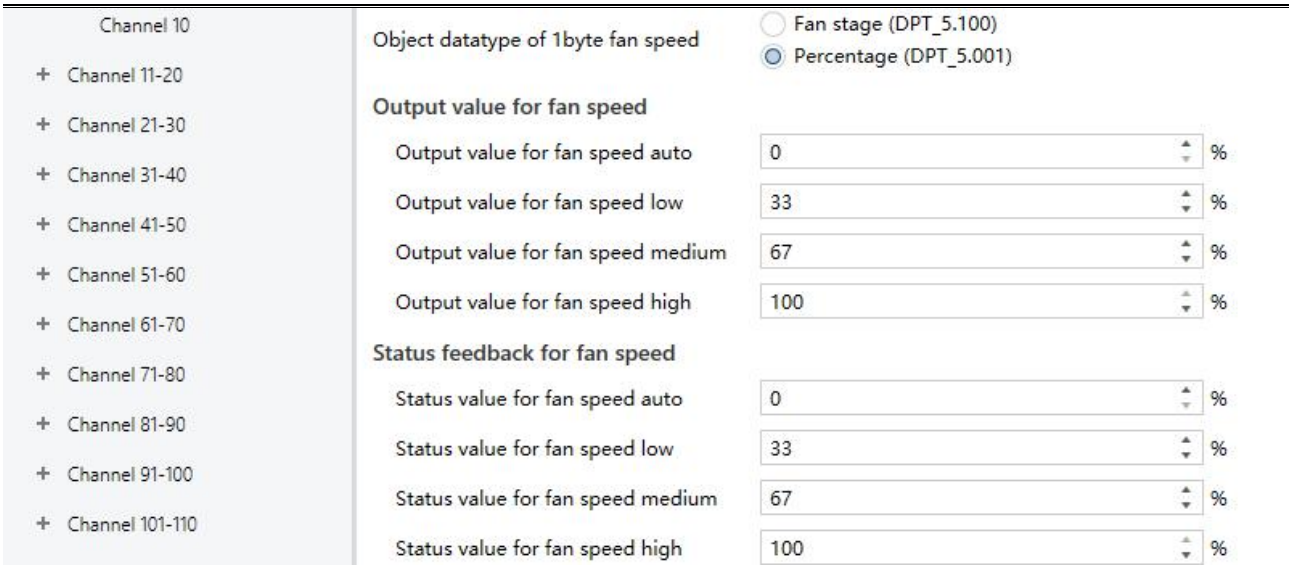


Fig.5.2.1.2(2) Parameter setting of air condition function

Parameter "Time period for request room temperature sensor [0...255]min"

This parameter is for setting the time period for read request room temperature sensor. Options: **0..255**

Send read request as default when the device voltage recovery.

Parameter "Min./Max. setpoint temperature [16..32]° C"

These two parameters are for setting the adjustable range of the setpoint temperature, the minimum value should be less than the maximum value. If the setpoint temperature beyond the limited range, the will output the limited temperature. Options:

16°C

17°C

...

32°C

Control mode setting

Parameter "Auto/Heating/Cooling/Fan/Dehumidification mode"

Corresponding mode setting is visible when these parameters are enabled.

--Parameter "Output value for auto/heating/cooling/fan/dehumidification [0..255]"

These parameters are visible when the modes are enabled. Set the output value for switching to each mode.

Options:**0..255**

——Parameter "Status value for auto/heating/cooling/fan/dehumidification [0..255]"

These parameters are visible when the modes are enabled. Set the status feedback value of each mode.

Options:**0..255**

Parameter "Object datatype of 1byte fan speed"

This parameter is for setting the object datatype of 1byte fan speed. Options:

Fan stage (DPT 5.100)

Percentage (DPT 5.001)

Output value for fan speed

——Parameter "Output value for fan speed auto/low/medium/high"

These parameters are for setting the output value for switching to each fan speed, support 4 fan speeds: auto, low, medium, high. Options are according to the object datatype of previous parameter: **0..255/0..100**

Status feedback for fan speed

——Parameter "Status value for fan speed auto/low/medium/high"

These parameters are for setting the status feedback value of each fan speed, support 4 fan speeds: auto, low, medium, high. Options are according to the object datatype of previous parameter: **0..255/0..100**

5.2.1.3. Parameter of room temperature unit

This chapter explains the room temperature unit function of KNX channel, including the basic, control with control mode, and with fan speed. Parameters as follow are visible when “Room temperature unit”, “Room temperature unit(with operation mode)” or “Room temperature unit(with operation mode & fan speed)” is selected.

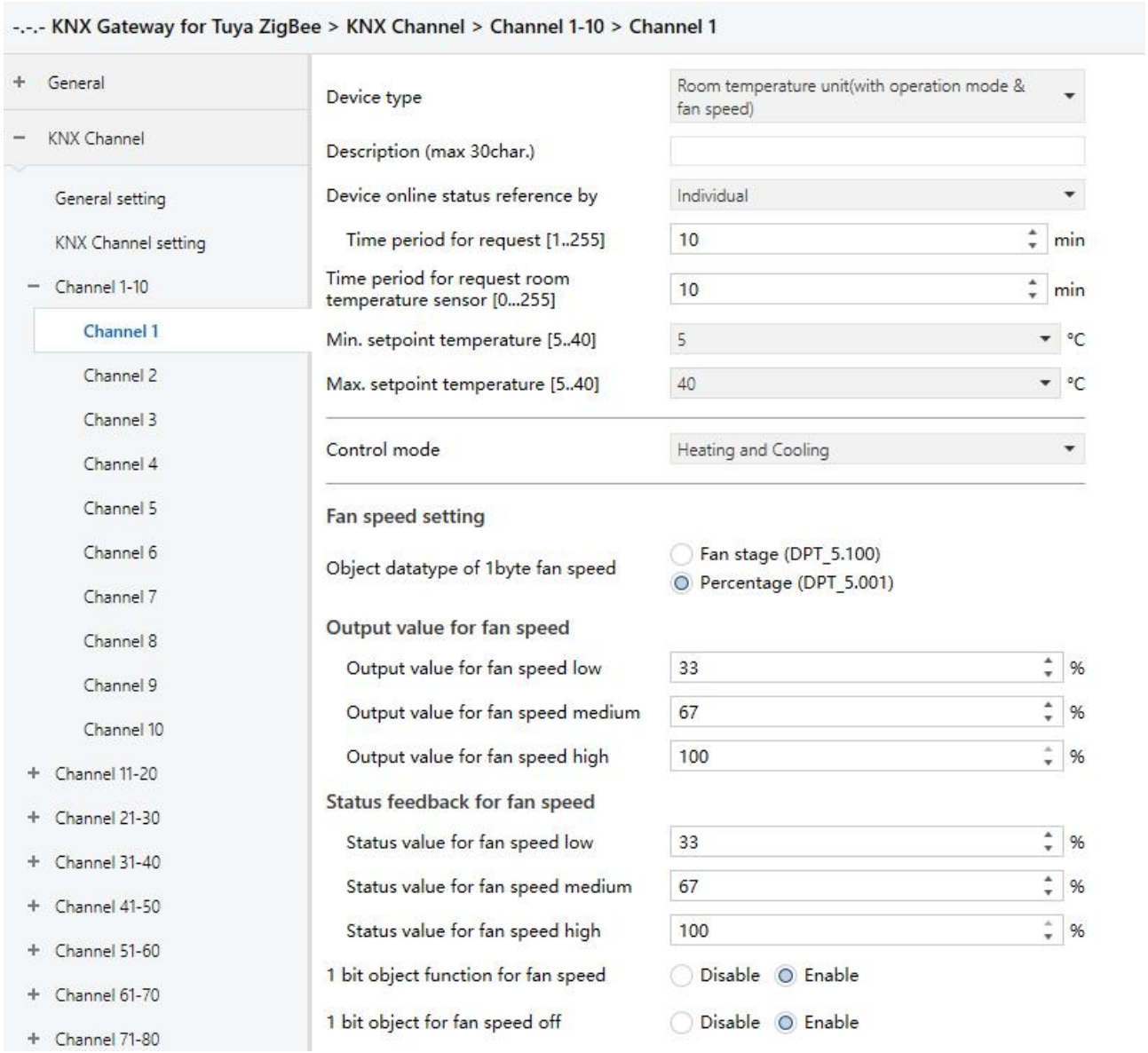


Fig.5.2.1.3 Parameter setting of room temperature unit

Parameter “Time period for request room temperature sensor [0...255]min”

This parameter is for setting the time period for read request room temperature sensor. Options: **0..255**

Send read request as default when the device voltage recovery.

Parameter "Min./Max. setpoint temperature [5..40] ° C"

These two parameters are for setting the adjustable range of the setpoint temperature, the minimum value should be less than the maximum value. If the setpoint temperature beyond the limited range, the will output the limited temperature. Options:

5°C

6°C

...

40°C

Parameter "Control mode"

This parameter is for setting temperature control mode, support 3 types: heating, cooling and heating/cooling. Options:

Heating

Cooling

Heating and Cooling

Fan speed setting

This setting is visible when "Room temperature unit(with operation mode & fan speed)" is selected

Parameter "Object datatype of 1byte fan speed"

This parameter is for setting the object datatype of 1byte fan speed. Options:

Fan stage (DPT 5.100)

Percentage (DPT 5.001)

Output value for fan speed

——Parameter "Output value for fan speed low/medium/high"

These parameters are for setting the output value for switching to each fan speed, support 3 fan speeds: low, medium, high. Options are according to the object datatype of previous parameter: **0..255/0..100**

Status feedback for fan speed

——Parameter "Status value for fan speed low/medium/high"

These parameters are for setting the status feedback value of each fan speed, support 3 fan speeds: low, medium, high. Options are according to the object datatype of previous parameter: **0..255/0..100**

Parameter "1 bit object function for fan speed"

This parameter is for setting whether to enable the object datatype of 1bit fan speed. When enabled, the 1 bit object of each fan speed is visible. When three objects value is 0, turn off the fan.

——Parameter "1 bit object for fan speed off"

This parameter is visible when previous parameter is enabled. Set whether to enable the object "1bit fan speed off" to visible.

5.2.1.4. Parameter of ventilation system

This chapter explains the ventilation system function of KNX channel. Parameters as follow are visible when "Ventilation system" is selected.

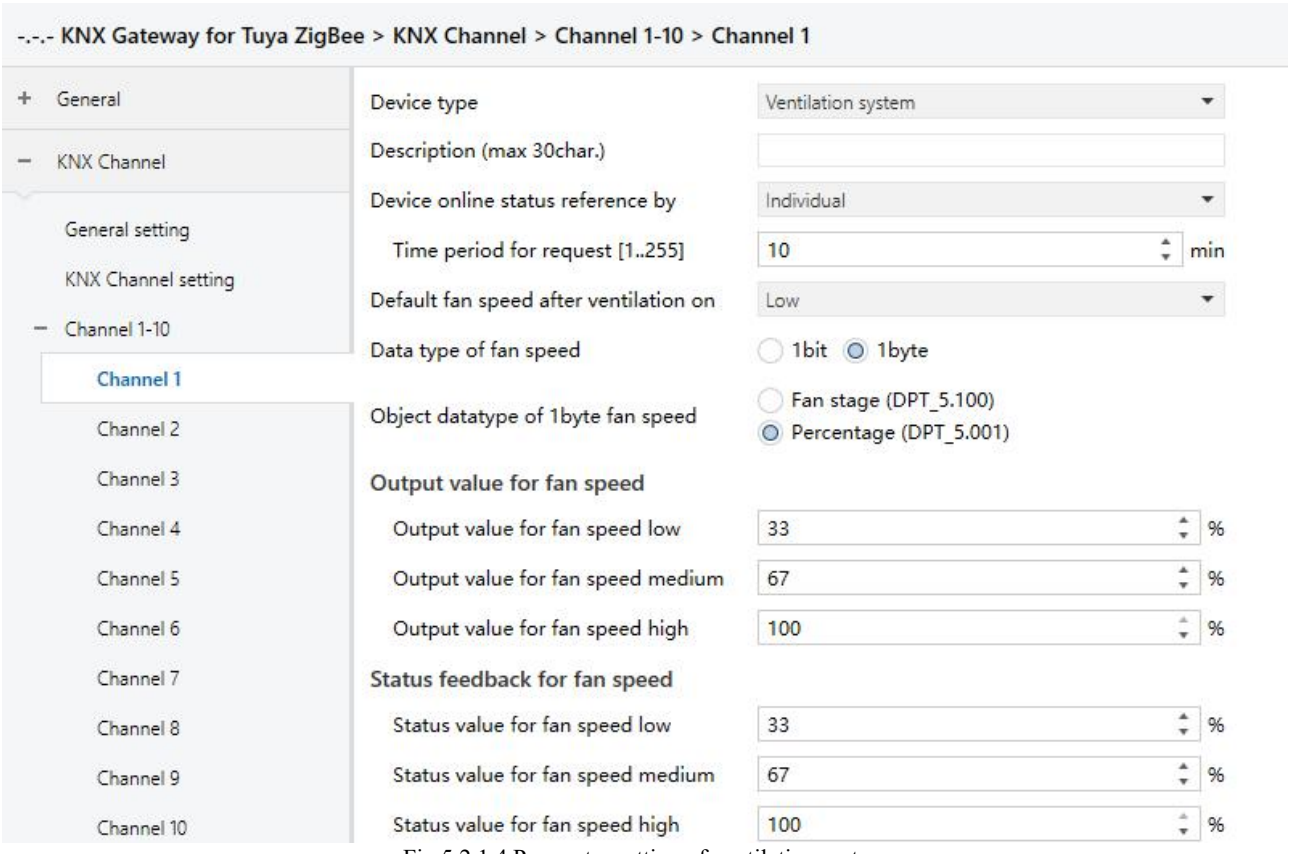


Fig.5.2.1.4 Parameter setting of ventilation system

Parameter "Default fan speed after ventilation on"

This parameter is for setting the initial fan speed after ventilation on. Switch status of ventilation is read from bus after download or reset, and default as off if can not read. Options:

Low

Medium

High

Last status

Parameter "Data type of fan speed"

This parameter is for setting the data type of ventilation fan speed. Options:

1bit

1byte

——Parameter "Object datatype of 1byte fan speed"

This parameter is visible when fan speed datatype is selected "1byte". Set the datatype of 1byte fan speed object. Options:

Fan stage (DPT 5.100)

Percentage (DPT 5.001)

Output value for fan speed

——Parameter "Output value for fan speed low/medium/high"

These parameters are for setting the output value for switching to each fan speed, support 3 fan speeds: low, medium, high. Value=0 is fan speed off. Options are according to the object datatype of previous parameter: **0..255/0..100**

Status feedback for fan speed

——Parameter "Status value for fan speed low/medium/high"

These parameters are for setting the status feedback value of each fan speed, support 3 fan speeds: low, medium, high. Value=0 is fan speed off. Options are according to the object datatype of previous parameter: **0..255/0..100**

——Parameter "Object value of fan speed off/low/medium/high"

This parameter is visible when fan speed datatype is selected "1bit". Set the control value for switching to each fan speed and the status feedback value. support 4 fan speeds: off, low, medium, high. Device will be update the fan speed display according to the feedback value. Options:

Low=0,Medium=0,High=0

Low=1,Medium=0,High=0

Low=0,Medium=1,High=0

Low=1,Medium=1,High=0

Low=0,Medium=0,High=1

Low=1,Medium=0,High=1

Low=0,Medium=1,High=1

Low=1,Medium=1,High=1

——Parameter "Delay between fan speed switch [0..100]*50ms "

This parameter is visible when fan speed datatype is selected "1bit". Set the delay time between fan speeds switchover. Options: **0..100**

Turn off fan speed before switch the fan speed, and turn on it after the delay time. When delay time is set as 0, it will not turn off first then turn on, but switch to next fan speed directly.

5.2.1.5. Parameter of audio control

This chapter explains the audio control function of KNX channel, including the basic and audio control with power on/off. Parameters as follow are visible when "Audio control" or "Audio control(with on/off)" is selected.

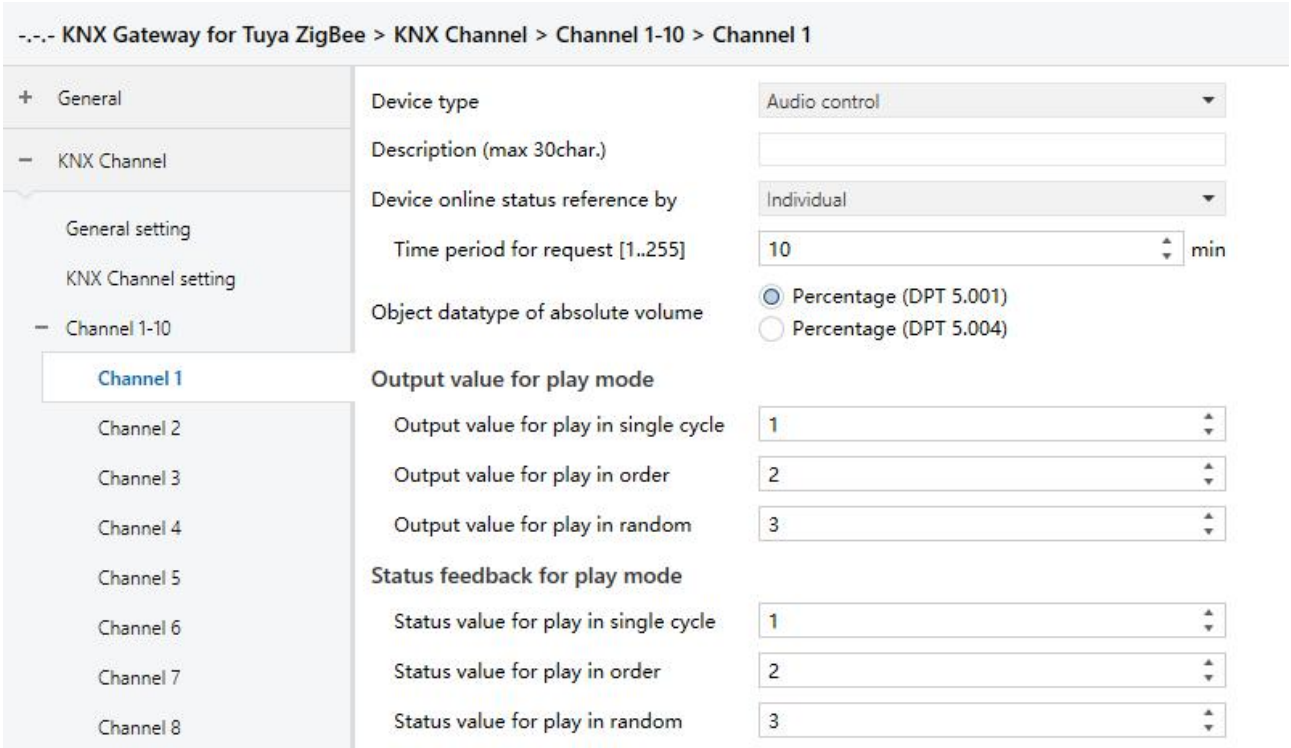


Fig.5.2.1.5 Parameter setting of audio control

Parameter "Object datatype of absolute volume"

This parameter is for setting the datatype of audio control object. Options:

Percentage (DPT 5.001)

Percentage (DPT 5.004)

Output value for play mode

——Parameter "Output value for play in single cycle/order/random"

These parameters are for setting the control value of each mode, including single cycle/order/random play.

Options: **0..255**

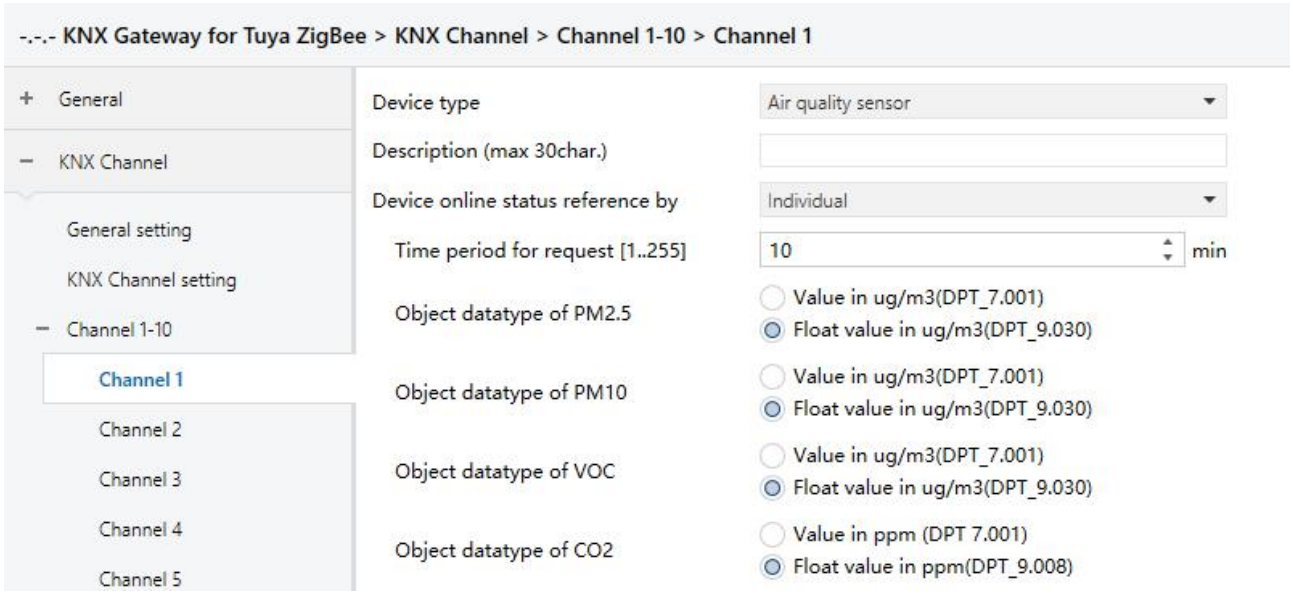
Status feedback for play mode

——Parameter "Status value for for play in single cycle/order/random"

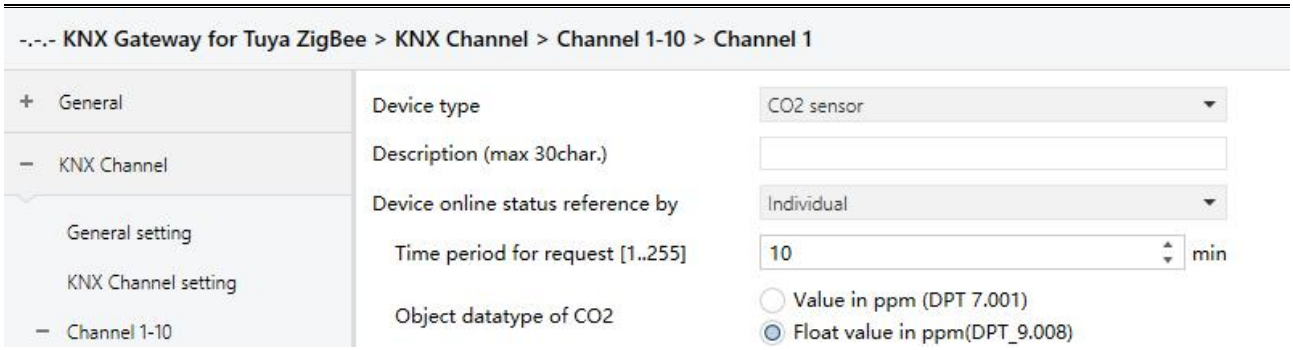
These parameters are for setting the status value of each mode, including single cycle/order/random play. Device will be update the play mode display according to the feedback value. Options: **0..255**

5.2.1.6. Parameter of sensor function

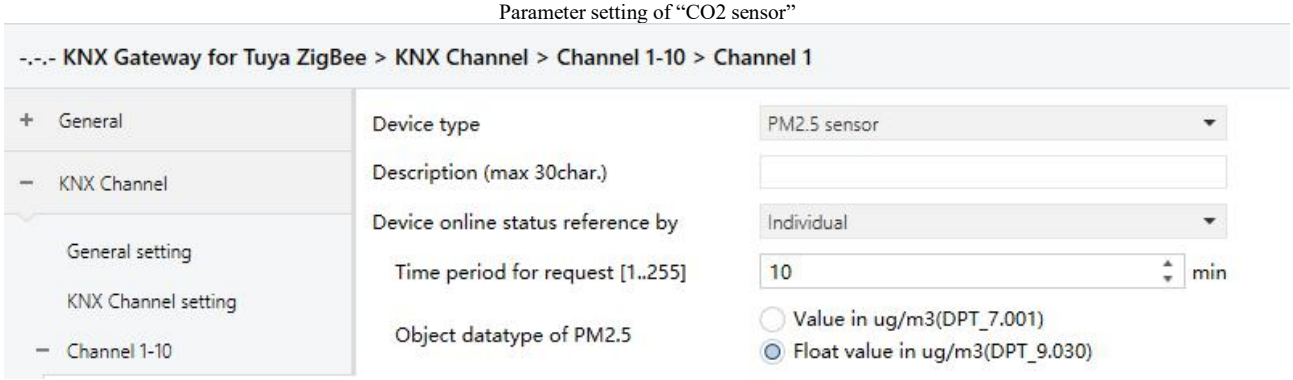
This chapter explains the sensor function of KNX channel, including air quality sensor, presence sensor, motion sensor, brightness sensor, I/O sensor and etc.



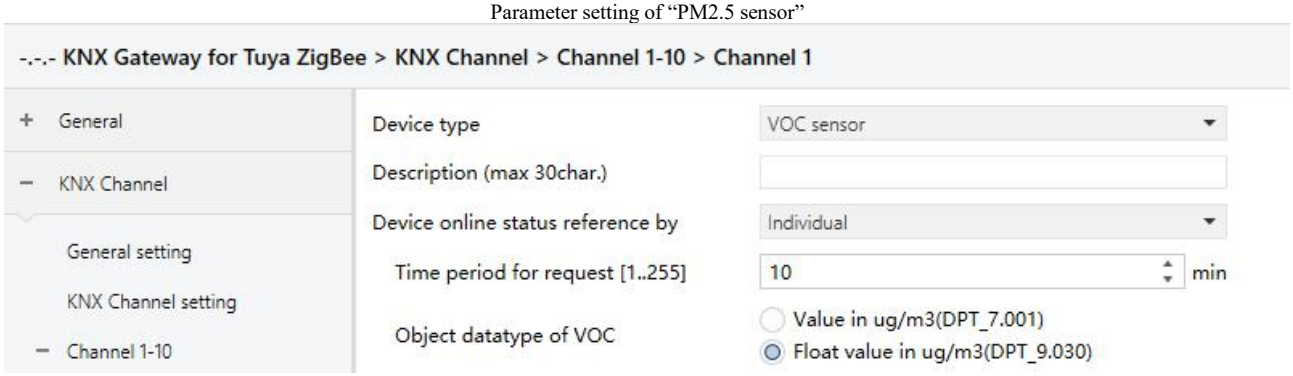
Parameter setting of "Air quality sensor"



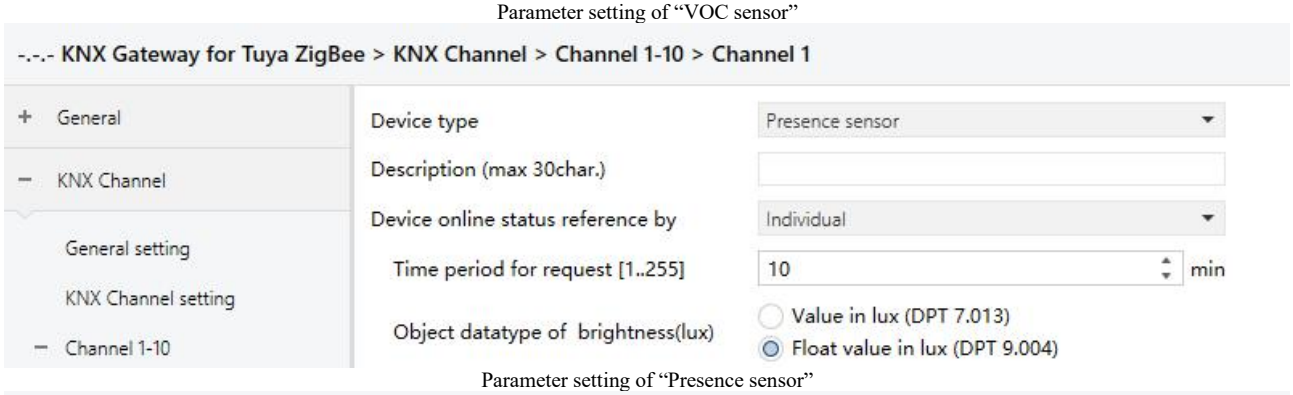
Parameter setting of "CO2 sensor"



Parameter setting of "PM2.5 sensor"

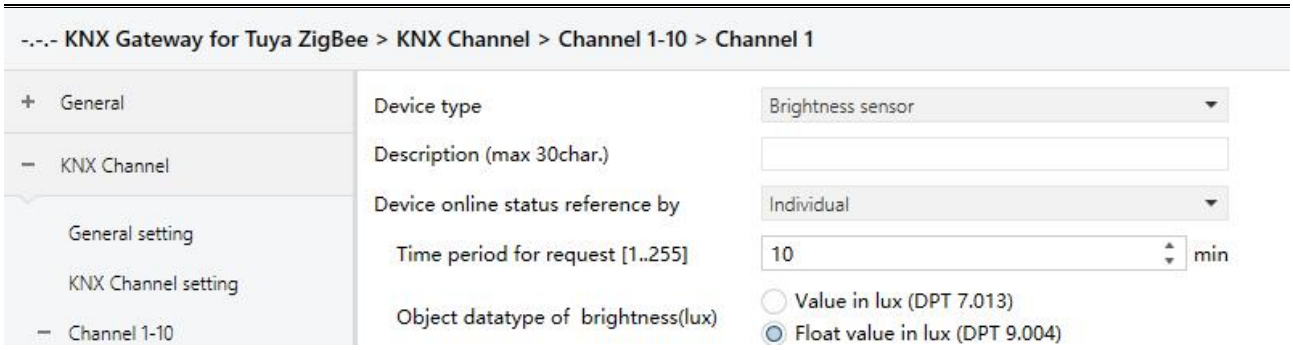


Parameter setting of "VOC sensor"

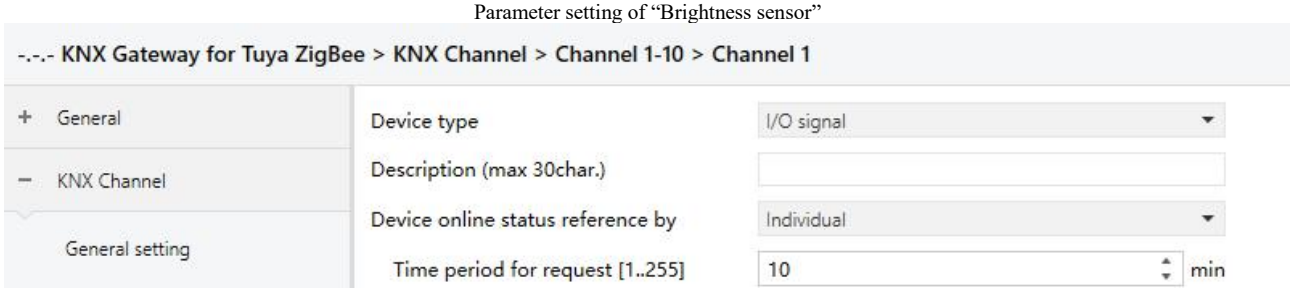


Parameter setting of "Presence sensor"

Parameter setting of "Motion sensor"



Parameter setting of "Brightness sensor"



Parameter setting of "I/O signal"

Fig.5.2.1.6 Parameter setting of sensor function

Parameter "Object datatype of PM2.5"

This parameter is visible when device type is selected "Air quality sensor" or "PM2.5 sensor". Set the object datatype of PM2.5. Options:

Value in ug/m3 (DPT 7.001)

Float value in ug/m3 (DPT 9.030)

Parameter "Object datatype of PM10"

This parameter is visible when device type is selected "Air quality sensor". Set the object datatype of PM10.

Options:

Value in ug/m3 (DPT 7.001)

Float value in ug/m3 (DPT 9.030)

Parameter "Object datatype of VOC"

This parameter is visible when device type is selected "Air quality sensor" or "VOC sensor". Set the object datatype of VOC. Options:

Value in ug/m3 (DPT 7.001)

Float value in ug/m3 (DPT 9.030)

Parameter "Object datatype of CO2"

This parameter is visible when device type is selected "Air quality sensor" or "CO2 sensor". Set the object datatype of CO2. Options:

Value in ppm (DPT 7.001)

Float value in ppm (DPT 9.008)

Parameter "Object datatype of brightness(lux)"

This parameter is visible when device type is selected "Presence sensor" or "Brightness sensor". Set the object datatype of brightness. Options:

Value in lux (DPT 7.013)

Float value in lux (DPT 9.004)

5.2.1.7. Parameter of current metering function

This chapter explains the current metering function of KNX channel. Parameters as follow are visible when device type is selected "Current metering".

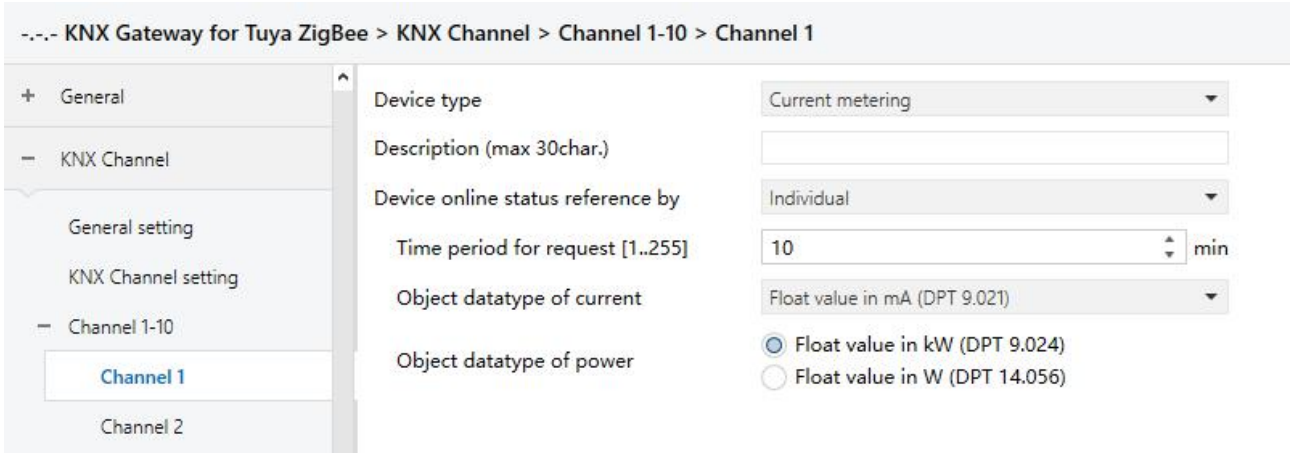


Fig.5.2.1.7 Parameter setting of current metering function

Parameter "Object datatype of current"

This parameter is for setting the object datatype of current. Options:

Value in mA (DPT 7.012)

Float value in mA (DPT 9.021)

Float value in A (DPT 14.019)

Parameter "Object datatype of power"

This parameter is for setting the object datatype of power. Options:

Float value in kW (DPT 9.024)

Float value in W (DPT 14.056)

5.2.1.8. Parameter of energy metering function

This chapter explains the energy metering function of KNX channel. Parameters as follow are visible when device type is selected “Energy metering”.

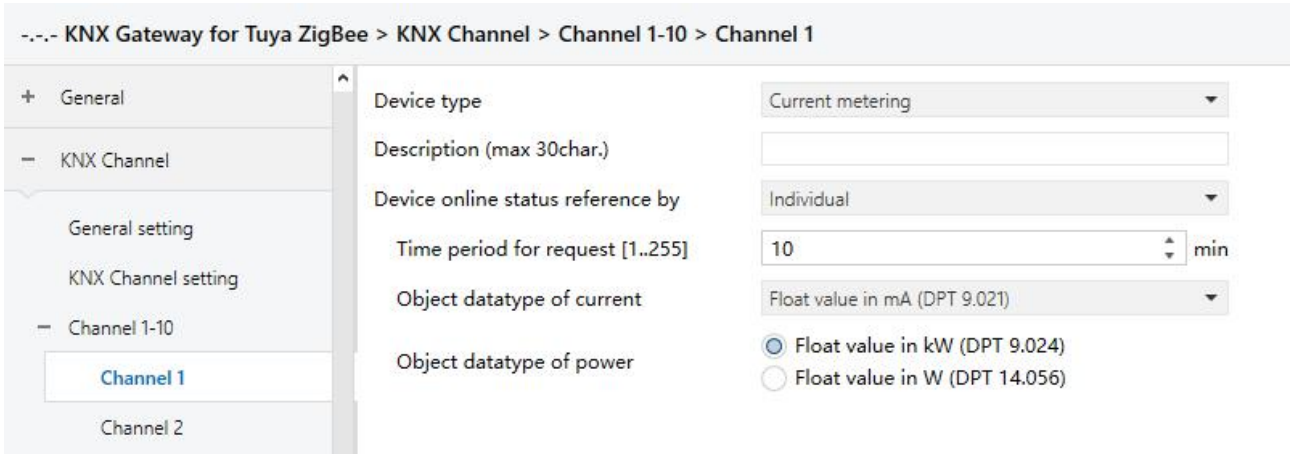


Fig.5.2.1.8 Parameter setting of energy metering function

Parameter “Object datatype of current”

This parameter is for setting the object datatype of current. Options:

Value in mA (DPT 7.012)

Float value in mA (DPT 9.021)

Float value in A (DPT 14.019)

Parameter “Object datatype of voltage”

This parameter is for setting the object datatype of voltage. Options:

Float value in mV (DPT 9.020)

Float value in V (DPT 14.027)

Parameter “Object datatype of power”

This parameter is for setting the object datatype of power. Options:

Float value in kW (DPT 9.024)

Float value in W (DPT 14.056)

Parameter “Object datatype of energy”

This parameter is for setting the object datatype of energy. Options:

Value in Wh (DPT 13.010)

Value in kWh (DPT 13.013)

5.3. Parameter window “Room temperature controller”

5.3.1. Parameter window “RTC Channel setting”

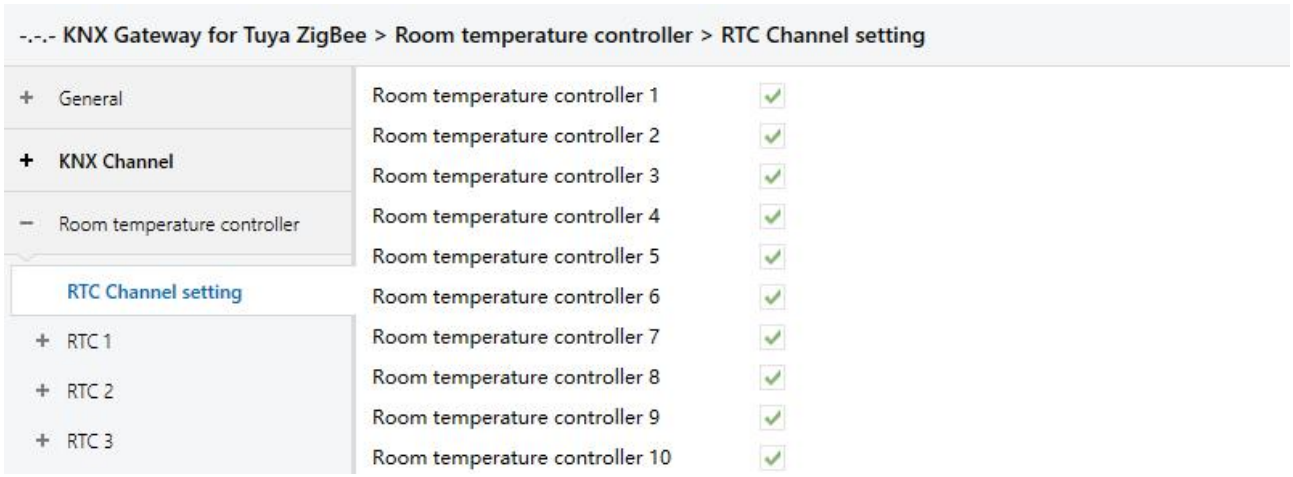


Fig.5.3.1 “RTC Channel setting” Parameter window

Parameter “Room temperature controller 1/2/3/...”

This parameter is for setting whether to enable the setting interface of RTC device, display corresponding interface. Up to enable 10 channels.

5.3.2. Parameter window “RTC x”(x=1~10)

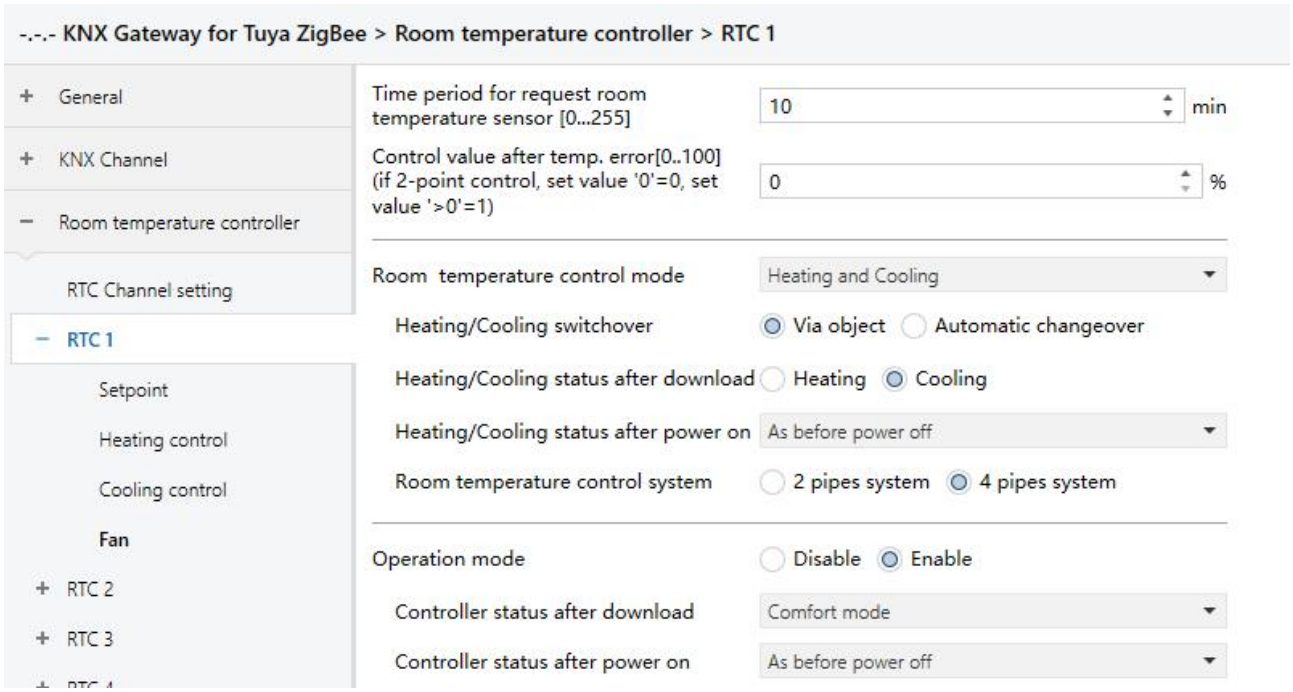


Fig.5.3.2(1) “RTC 1” parameter window

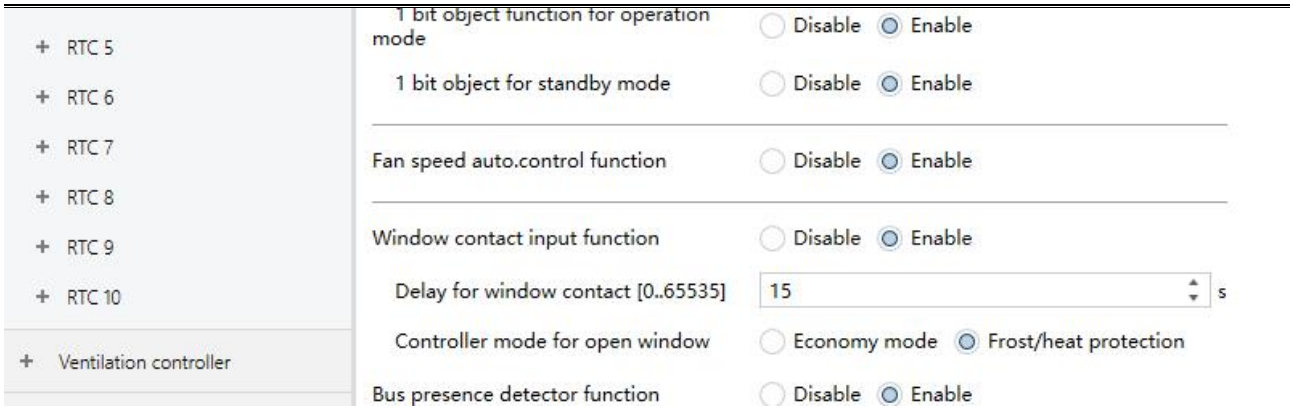


Fig.5.3.2(2) "RTC 1" parameter window

Parameter "Time period for request room temperature sensor [0..255]min"

This parameter is for setting the time period for read request external temperature sensor. Send read request to external temperature sensor after the device voltage recovery or finish programming. Options: **0..255**

Parameter "Control value after temp. error[0..100]% (if 2-point control, set value '0'=0, set value '>0'=1)"

This parameter is for setting the control value when temperature error occur. Options: **0..100**

If 2-Point control, then the parameter value is 0, as well as the control value; if the parameter value is more than 0, then the control value will be 1.

Parameter "Room temperature control mode"

This parameter is for setting RTC control mode, support 3 types: heating, cooling and heating/cooling.

Options:

Heating

Cooling

Heating and Cooling

Heating and Cooling: heating and cooling are available. At the same time, four parameters as follow are visible.

——Parameter "Heating/Cooling switchover"

This parameter is for setting the switchover way of Heating/Cooling. Options:

Via object

Automatic changeover

——Parameter "Heating/Cooling status after download"

This parameter is for setting the heating/cooling control mode of device when power on RTC after download.

Options:

Heating

Cooling

——Parameter "Heating/Cooling status after power on"

This parameter is for setting the heating/cooling control mode of device when power on RTC after voltage recovery. Options:

Heating

Cooling

As before power off

As before voltage failure: When the device is reset after power on, the control mode will recover as before voltage failure or restart. If it is the first time the device is used or a newly enabled function page, the control mode after the device is started is in an uncertain state, and it needs to be manually selected at this time.

——Parameter "Room temperature control system"

This parameter is for setting the type of RTC control system, that is, pipe types of fan coil water inlet/outlet. Options:

2 pipes system

4 pipes system

2 pipes system: Shares an inlet and outlet pipe for heating and cooling, that is, both hot and cold water are controlled by a valve.

4 pipes system: Has its own inlet and outlet pipes for heating and cooling, and two valves are needed to control the entry and exit of hot water and cold water respectively.

Parameter "Operation mode"

This parameter is for setting whether to enable RTC operation mode. Options:

Disable

Enable

When enable, support 4 modes: comfort, standby, economy and frost/heat protection. Support datatype of 1bit and 1byte, and preset a operation mode when download and voltage recovery.

Four parameters as follow are visible when RTC operation mode enabled.

——Parameter "Controller status after download"

This parameter is for setting the operation mode when power on RTC after download. Options:

Standby mode

Comfort mode

Economy mode

——Parameter "Controller status after power on"

This parameter is for setting the operation mode when power on RTC after voltage recovery. Options:

Standby mode

Comfort mode

Economy mode

Frost/heat protection

As before power off

——Parameter "1 bit object function for operation mode"

This parameter is for setting whether to enable 1 bit objects of operation mode are visible. Options:

Disable

Enable

——Parameter "1 bit object for standby mode"

This parameter is visible when previous parameter enabled. Set whether to enable 1 bit object of standby mode is visible. Options:

Disable

Enable

Three parameters as follow are visible when RTC operation mode disabled.

——Parameter " Initial setpoint temperature (° C)"

This parameter is for setting the initial value of setpoint temperature. Options:

10.0

10.5

...

35.0

Automatic H/C mode changeover dead zone

——Parameter " Upper/Lower dead zone"

These two parameters are visible when control mode "Heating and Cooling" is selected, and "Automatic changeover" is selected. Setting the dead zone range of auto switchover heating/cooling. Options:

0.5°C

1.0°C

...

10°C

Under heating control, when the actual temperature(T) > or = the setpoint temperature + the upper dead zone, then mode heating switch to cooling;

Under cooling control, when the actual temperature(T) < or = the setpoint temperature + the upper dead zone, then mode cooling switch to heating.

Parameter "Fan speed auto.control function"

This parameter is for setting whether to enable fan auto control interface is visible. Options:

Disable

Enable

Parameter "Window contact input function"

This parameter is visible when RTC operation mode enabled. Set whether to link to window contact status.

Options:

Disable

Enable

——Parameter "Delay for window contact [0..65535]s"

This parameter is visible when RTC operation mode and window contact input function are enabled. Set the delay time to window contact detection. That is, when the window is open within the set value, the window is not open. If the time is out of the set value, the window is open. Options: **0..65535**

——Parameter "Controller mode for open window"

This parameter is visible when RTC operation mode and window contact input function are enabled. If window status is open, perform corresponding operation according to configuration. Options:

Economy mode

Frost/heat protection

Operable switch and set temperature, and heating/cooling mode. For the operation mode, it is recorded in the background if a control telegram is received, and perform after the window close; it is stay current operation mode if no telegram is received.

Parameter "Bus presence detector function"

This parameter is visible when RTC operation mode enabled. Set whether to link to bus presence detector status. Options:

Disable

Enable

If presence is detected, enter the comfort mode and recovery original mode after leaving. If there is a telegram/manual to adjust the mode, it will not recovery the previous mode after leaving. (If receive presence status cyclically, no comfort mode retriggered, and only can be after leaving.)

Parameter "Min./Max. setpoint temperature [5..40][°C]"

These parameters are visible when RTC operation mode disabled. Set to limit the adjustable range of the setpoint temperature, the minimum value should be less than the maximum value. If the setpoint temperature beyond the limited range, the will output the limited temperature. Options:

5°C

6°C

...

40°C

5.3.1.1. Parameter window “Setpoint”

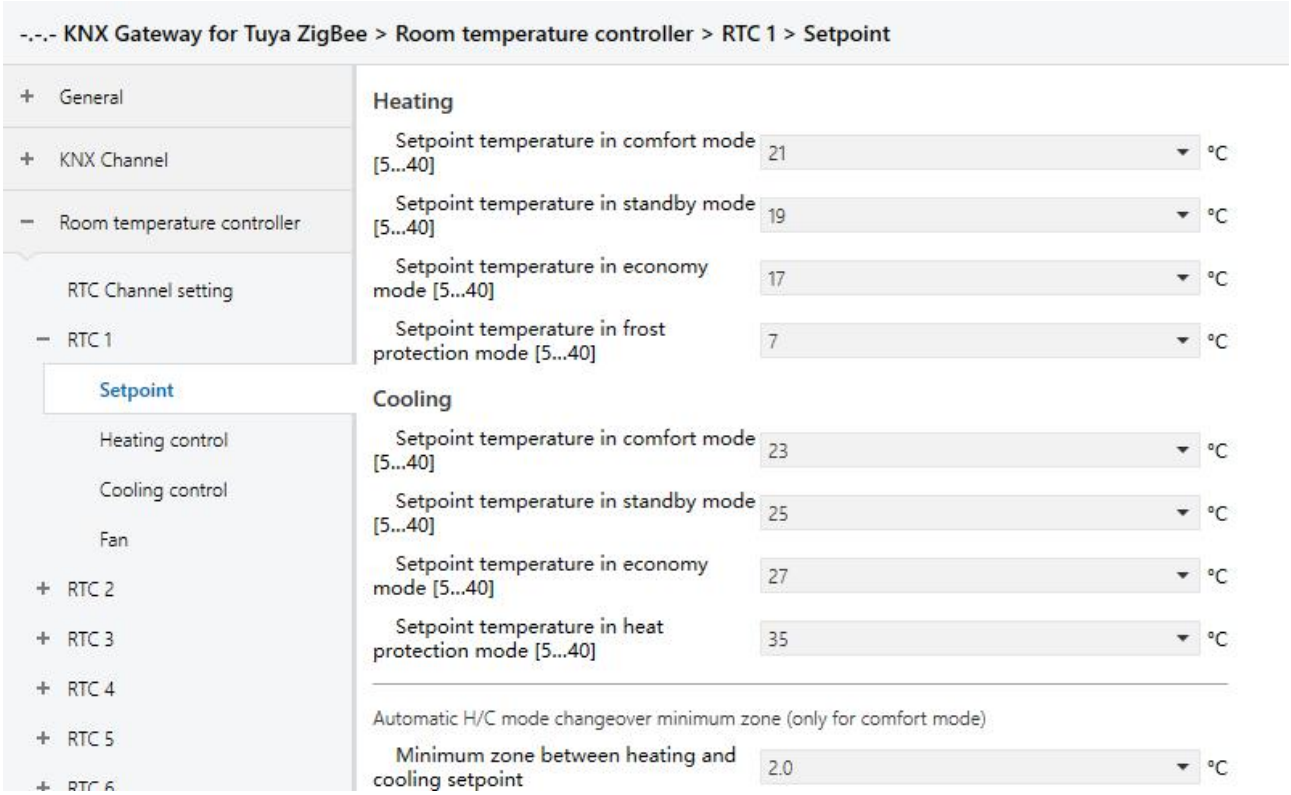


Fig.5.3.1.1 “Setpoint” parameter window

Parameters of this window are visible when RTC operation mode enabled, display according to control mode. The temperature setpoint value uses the way of absolute adjustment.

Parameter “Setpoint temperature in comfort mode [5...40]° C”

Parameter “Setpoint temperature in standby mode [5...40]° C”

Parameter “Setpoint temperature in economy mode [5...40]° C”

Parameter “Setpoint temperature in frost protection mode [5...40]° C”(for heating)

Parameter “Setpoint temperature in heat protection mode [5...40]° C”(for cooling)

These parameters are for setting the temperature setpoint value of each mode. Options:

5°C

6°C

...

40°C

Automatic H/C mode changeover minimum zone (only for comfort mode)

Parameter "Minimum zone between heating and cooling setpoint"

This parameter is visible when control mode selects "Heating and Cooling", and "Automatic changeover" is selected. Setting the minimum zone of temperature between heating and cooling setpoint. Options:

0.5°C

1.0°C

...

10°C

Heating/cooling auto switchover according to temperature setpoint value of comfort mode:

Auto switchover to cooling when actual temperature higher than the temperature setpoint value of comfort mode;

Auto switchover to heating when actual temperature lower than the temperature setpoint value of comfort mode.

Parameter "Min./Max. setpoint temperature [5..40] ° C"

These two parameters are for setting to limit the adjustable range of the setpoint temperature, the minimum value should be less than the maximum value. If the setpoint temperature beyond the limited range, the will output the limited temperature. Options:

5°C

6°C

...

40°C

5.3.1.2. Parameter window “Heating/Cooling control”

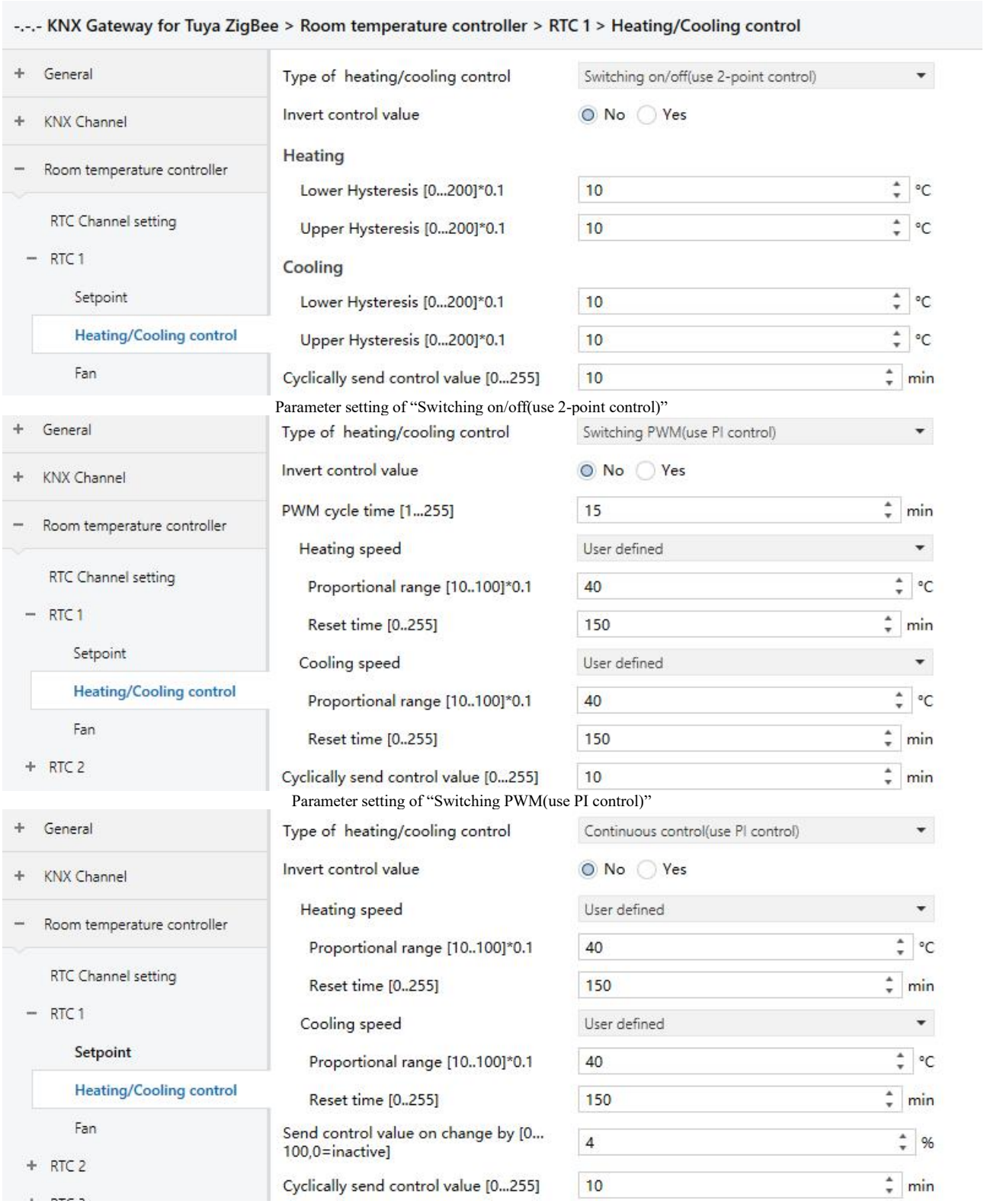


Fig.5.3.1.2(1) “Heating/Cooling control” parameter window

Parameters of this window display according to control mode and control system(2 pipe or 4pipe).

Parameter "Type of heating/cooling control"

This parameter is for setting the type of heating/cooling control. Different control types are suitable for controlling different temperature controllers. Options:

Switching on/off(use 2-point control)

Switching PWM(use PI control)

Continuous control(use PI control)

Parameter "Invert control value"

This parameter is for setting whether to invert control value or normal sending control value, so that the control value will be suitable for the valve type. Options:

No

Yes

Yes: Sending the control value to the bus through objects after inverting the control value.

Two parameters as follow are suitable for 2 point control:

——Parameter "Lower Hysteresis [0...200]*0.1°C"

——Parameter "Upper Hysteresis [0...200]*0.1°C"

These two parameters are for setting the lower/upper hysteresis temperature in HVAC heating or cooling. Options: **0..200**

Under heating control,

When the actual temperature(T) > the setting temperature + the upper hysteresis temperature, then will stop heating;

When the actual temperature(T) < the setting temperature - the lower hysteresis temperature, then will start heating.

For example, the lower hysteresis temperature is 1℃, the upper hysteresis temperature is 2℃, the setting temperature is 22℃, if T is higher than 24℃, then it will stop heating; if T is lower than 24℃,then it will start heating; if T is between 21~24℃, then it will maintain the previous status.

Under the cooling control,

When the actual temperature (T) < the setting temperature -the lower hysteresis temperature, then will stop cooling;

When the actual temperature (T) > the setting temperature +the upper hysteresis temperature, then will start cooling.

For example, the lower hysteresis temperature is 1℃, the upper hysteresis temperature is 2℃, the setting temperature is 26℃, if T is lower than 25℃, then it will stop cooling; if T is lower than 28℃, then it will start cooling; if T is between 28~25℃, then it will maintain the previous status.

2-point control mode is a very simple control mode. When adopting this control mode, it is necessary to set the upper hysteresis temperature and the lower hysteresis temperature through parameters. When setting the hysteresis temperature, the following effects need to be considered:

1. When hysteresis interval is small, the temperature range will be small, however, frequent sending of control value will bring large load to the bus;

2. When hysteresis interval is large, the switch switching frequency will be low, but it is easy to cause uncomfortable temperature change.

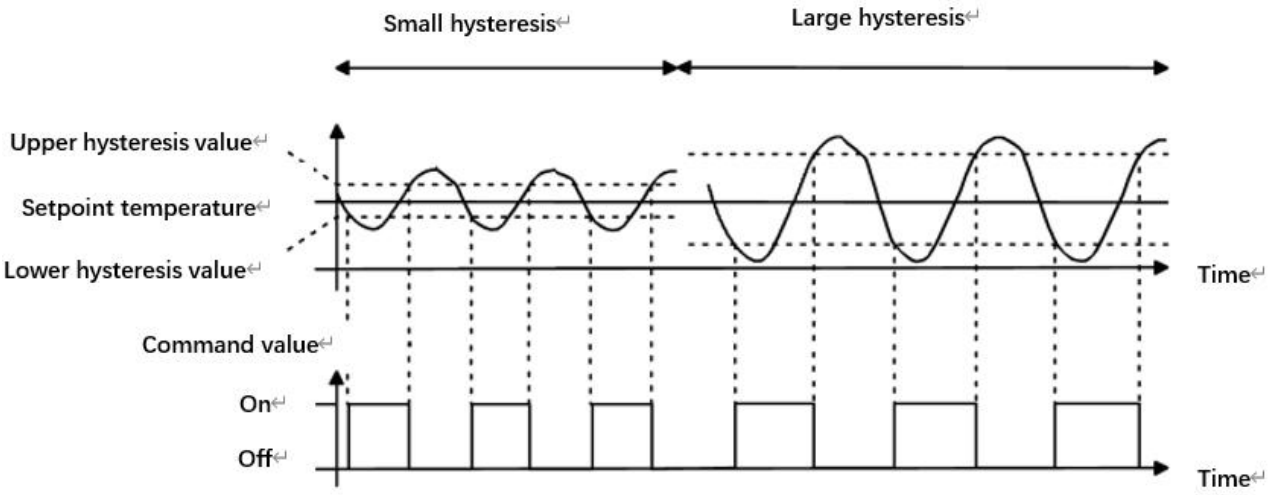


Fig.5.4.1.2(2) Effects of hysteresis on control value switch action(heating) under2-point control mode

Two parameters as follow are suitable for PI control:

——Parameter "Heating speed "

——Parameter "Cooling speed"

These two parameters are for setting the responding speed of heating or cooling PI controller. Different responding speeds are suitable for different environments.

Options:

Hot water heating (5K/150min)

Underfloor heating (5K/240 min)

Electrical heating (4K/100min)

Split unit (4K/90min)

Fan coil unit (4K/90min)

User defined

Options:

Cooling ceiling (5K/240min)

Split unit (4K/90min)

Fan coil unit(4K/90min)

User defined

——Parameter "Proportional range [10..100]*0.1° C"(P value)

——Parameter "Reset time [0..255]min"(I value)

These two parameters are visible when "User defined" is selected. Set the PI value of PI controller.

Options: **10..100 (P value)**

Options: **0..255 (I value)**

——Parameter "PWM cycle time [1..255]min"

This parameter is only visible when the control type is "Switching PWM(use PI control)". Set the period of the control object cycle to send the switch value, the object sends the switch value according to the duty cycle of the control value. For example, if the set period is 10 min and the control value is 80%, then the object will send an open telegram for 8 min. If the control value is changed, the time duty ratio of the on/ off telegram of the object will also change, but the period is still the time of parameter setting.

Options: **1..255**

The PI values of "Switching PWM (use PI control)" and "Continuous control (use PI control)" are the same, only different in control objects, the control object of "Continuous control" output PI value(1byte) directly, while the control value of "Switching PWM" output a "on/off" telegram according to the duty cycle of the control value.

---Parameter "Send control value on change by [0...100.0=inactive]%"

This parameter is visible when control type is "Continuous control (use PI control)", for setting the changing value of the control value to be sent to the bus. Options: **0..100, 0＝inactive**

Parameter "Cyclically send control value [0...255]min"

This parameter is for setting the period for cyclically sending the control value to the bus. Options: **0..255**

In PI control mode, the predefined control parameters of each PI controller in heating or cooling system are recommended as follows:

(1) Heating

Heating type	**P value**	**I value(integration time)**	**Recommended PI control type**	**Recommended PWM period**
Hot water Heating	5K	150min	Continuous/PWM	15min
Underfloor heating	5K	240min	PWM	15-20min
Electrical heating	4K	100min	PWM	10-15min
Split unit	4K	90min	PWM	10-15min
Fan coil unit	4K	90min	Continuous	--

(2) Cooling

Cooling type	**P value**	**I value(integration time)**	**Recommended PI control type**	**Recommended PWM period**
Cooling ceiling	5K	240min	PWM	15-20mIn
Split unit	4K	90min	PWM	10-15min
Fan coil unit	4K	90min	Continuous	--

(3) User defined

When the parameter"Heating/Cooling speed" is set to "User defined", the parameter value of P (scale factor) and I (integration time) can be set through the parameter. When adjusting the parameters, refer to the fixed PI value mentioned in the above table. Even if the control parameters are adjusted slightly, the control behavior will

be significantly different.

In addition, the integration time should be set properly. If the integration time is too long, the adjustment will be slow, and the oscillation will not be obvious; if the integration time is too small, the adjustment will be fast, but the oscillation will occur. 0 means the integral term is not used.

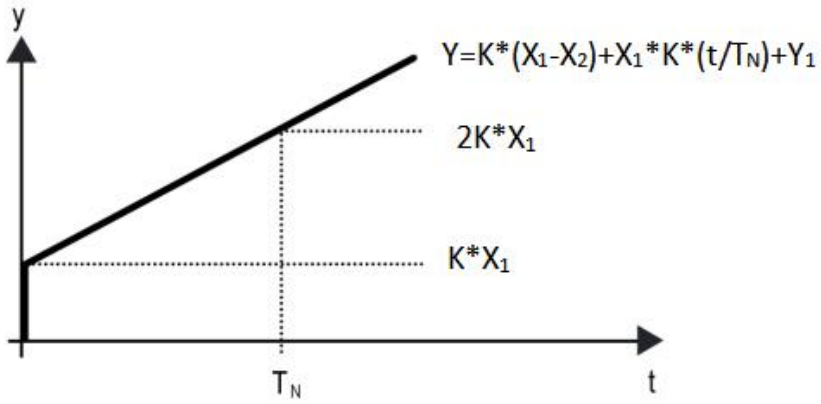


Fig.5.3.3 (7) control value of PI control mode

Y: control value

Y1: last control value

X1: temperature deviation = set temperature - actual temperature

X2: last temperature deviation = set temperature - actual temperature

T_N: integration time

K: scale factor (the scale factor is not zero)

PI control algorithm: $Y = K * (X1-X2) + X1 * K * t / T_N + Y1$

When the integration time is set to zero, the PI control algorithm is: $Y = K (X1-X2) + Y2$

Setting and influence of user-defined parameters:

Parameter setting	Effect
K: If the scale range is too small	Quick adjustment, and overshoot will occur
K: If the scale range is too small	Slow adjustment, but no overshoot
T_N: If the integration time is too short	Quick adjustment, but there will be oscillation
T_N: If the integration time is too long	Slow adjustment, no obvious oscillation

5.3.1.3. Parameter window “Fan”

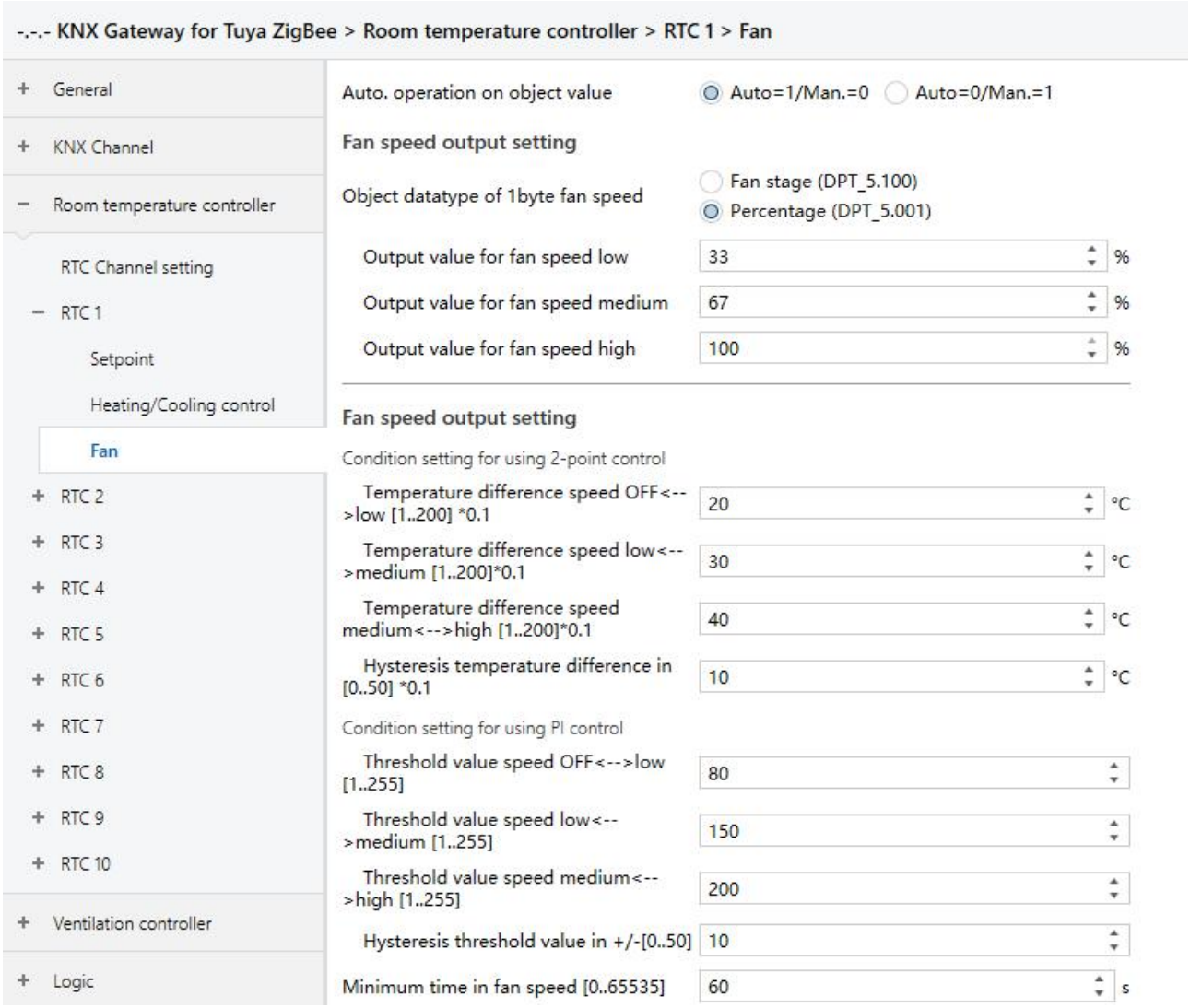


Fig.5.3.1.3 “Fan” parameter window

Parameters of this window are visible when fan auto control enabled.

Parameter “Auto. operation on object value”

This parameter is for setting the telegram value to activate automatic operation. Options:

Auto=1/Man.=0

Auto=0/Man.=1

Auto=1/Man.=0: When the object “Fan automatic operation” receives the telegram value “0”, activate the automatic operation, when receive “1”, exit the automatic operation.

Auto=0/Man.=1: When the object “Fan automatic operation” receives the telegram value “1”, activate the automatic operation, when receive “0”, exit the automatic operation.

After power-on, automatic operation is not activated by default.

Fan speed output setting

Parameter "Object datatype of 1byte fan speed"

This parameter is for setting the object datatype of 1 byte fan speed. Options:

Fan stage (DPT 5.100)

Percentage (DPT 5.001)

——Parameter "Output value for fan speed low/medium/high"

These three parameters are for setting the value sent for each fan speed switchover. Fan speed off when value is 0. Options according to fan object datatype: **1..255 /1..100**

Fan speed control setting

Condition setting for using 2-point control

Under 2-point control, controller will decide the fan power on/off or fan speed according to the temperature difference between the actual temperature and setpoint temperature.

Cooling: Temperature difference = actual temperature - setpoint temperature;

Heating: Temperature difference = setpoint temperature - actual temperature.

Parameter "Temperature difference speed OFF<-->low [1..200] *0.1° C"

This parameter is for setting the temperature difference between off-fan and low-level fan speeds.

Options: **1..200**

If the temperature difference is greater than or equal to this setting temperature difference, low-level fan speed will start running; if less than this setting temperature difference, the fan will be turned off.

Parameter "Temperature difference speed low<-->medium [1..200]*0.1° C"

Define the temperature difference for switching the fan speed to medium fan speed, if the control value is greater than or equal to this setting temperature difference, the medium fan speed will start running.

Options: **1..200**

Parameter "Temperature difference speed medium<-->high [1..200]*0.1° C"

Define the temperature difference for switching the fan speed to high fan speed, if the control value is greater than or equal to this setting temperature difference, the high fan speed will start running. Options: **1..200**

Parameter "Hysteresis temperature difference in [0..50] *0.1° C"

This parameter is for setting the hysteresis value of the temperature difference, which can avoid the

unnecessary action of the fan when the control value fluctuates near the temperature difference. Options: **0..50**

If value is 0, no hysteresis. Fan switch to speed once control value greater than temperature difference;

Suppose that hysteresis value is 0.5℃ and the temperature difference is 1℃, then the upper limit temperature difference 1.5℃ (Temperature difference+Hysteresis value) and the lower limit temperature difference 0.5℃ (Temperature difference-Hysteresis value). When the control value is between 0.5℃~1.5℃, fan action will not be caused, and the previous status will still be maintained. Only less than 0.5℃ or greater than or equal to 1.5℃ will change the running status of the fan.

Condition setting for using PI control

Under PI control, control value is PI operated within program, controller will power on/off fan or switch fan speed according to the threshold range of the control values.

Parameter "Threshold value speed OFF<–>low [1..255]"

Define threshold value for off-fan and low-level fan speeds, options: **1..255**

If the control value is greater than or equal to this setting threshold value, low-level fan speed will start running; if the control value is less than this setting threshold value, the fan will be turned off.

Parameter "Threshold value speed low<–>medium [1..255]"

Define the threshold value for switching the fan speed to medium fan speed, if the control value is greater than or equal to this setting threshold, the medium fan speed will start running. Options: **1..255**

Parameter "Threshold value speed medium<–>high [1..255]"

Define the threshold for switching the fan speed to high fan speed, if the control value is greater than or equal to this setting threshold, the high fan speed will start running. Options: **1..255**

Tip: The controller evaluates the threshold in ascending order.

First check →OFF <–>low fan speed threshold →low fan speed <–>medium fan speed →medium fan speed <–>high fan speed.

The correctness of functional execution is guaranteed only in this case:

The threshold of OFF <–> low fan speed is lower than that of low fan speed <–> medium fan speed, and the threshold of low fan speed <–> medium fan speed is lower than that of medium fan speed <–> high fan speed.

Parameter "Hysteresis threshold value in +/-[0..50]"

This parameter is for setting the hysteresis value of the threshold value, which can avoid the unnecessary action of the fan when the control value fluctuates near the threshold. Options: **0..50**

If value is 0, no hysteresis. Fan switch to speed once control value greater than threshold value;

Suppose that hysteresis value is 10 and the threshold is 50, then the upper limit threshold 60 (Threshold value+Hysteresis value) and the lower limit threshold 40 (Threshold value-Hysteresis value). When the control value is between 40 ~60, fan action will not be caused, and the previous status will still be maintained. Only less than 40 or greater than or equal to 60 will change the running status of the fan.

Parameter "Minimum time in fan speed [0..65535]s"

Defines the residence time of the fan from the current fan speed to a higher fan speed or lower fan speed, that is, the minimum time for a fan speed operation.

If you need to switch to another fan speed, you need to wait for this period of time before switching.

If the current fan speed has been running long enough, the fan speed can be changed quickly.

Options: **0..65535**

0: there is no minimum running time, but the delay switching time of fan speed still needs to be considered.

Note: The residence time for this parameter setting is only enabled in Auto mode.

5.4. Parameter window “Ventilation controller”

5.4.1. Parameter window “Control setting”

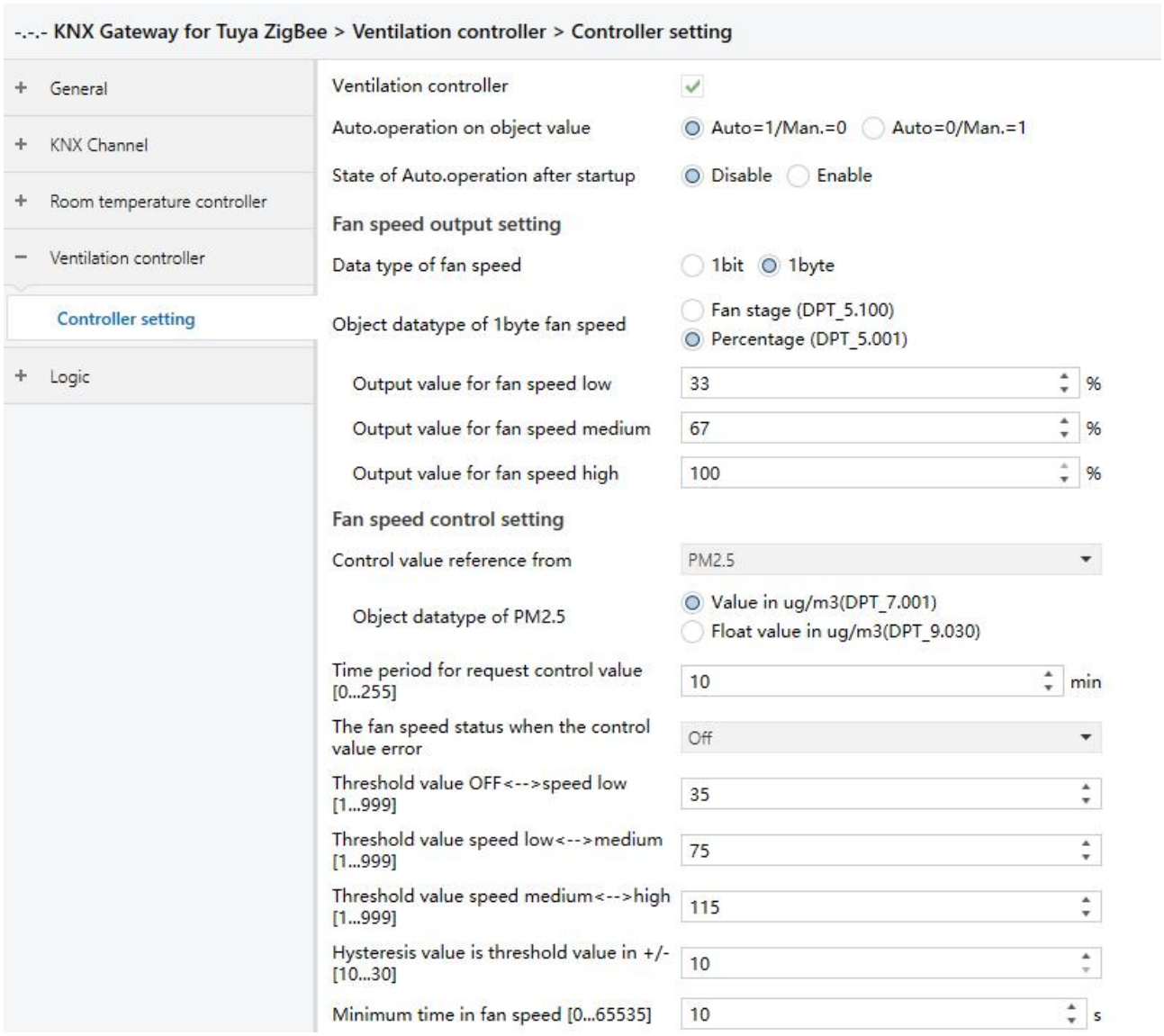


Fig.5.4.1 “Control setting” parameter window

Parameter “Ventilation controller”

This parameter is for setting whether to enable ventilation controller, to realize the automatic control of the fan speed, and link the fan speed to the detection value of PM2.5 or CO2, VOC. Parameters as follow are visible when enabled.

Parameter “Auto. operation on object value”

This parameter is for setting the telegram value to activate automatic operation. Options:

Auto=1/Man.=0

Auto=0/Man.=1

Auto=1/Man.=0: When the object "Fan automatic operation" receives the telegram value "0", activate the automatic operation, when receive "1", exit the automatic operation.

Auto=0/Man.=1: When the object "Fan automatic operation" receives the telegram value "1", activate the automatic operation, when receive "0", exit the automatic operation.

After power-on, automatic operation is not activated by default.

Parameter "State of Auto.operation after startup"

This parameter is for setting whether to enable state of Auto.operation after startup the device. Options:

Disable

Enable

Fan speed output setting

Parameter "Data type of fan speed"

This parameter is for setting the data type of fan speed. Options:

1bit

1byte

——Parameter "Object datatype of 1byte fan speed"

This parameter is visible when "1byte" is selected. Set the object datatype of 1byte fan speed. Options:

Fan stage (DPT 5.100)

Percentage (DPT 5.001)

——Parameter "Output value for fan speed low/medium/high"

This parameter is visible when "1byte" is selected. Set the value sent for each fan speed switchover. Fan speed off when value is 0. Options according to fan object datatype: **1..255 /1..100**

——Parameter "Object value of fan speed off/low/medium/high"

This parameter is visible when 1bit" is selected. Set the value sent for each fan speed, sent by three 1 bit objects at the same time. Options:

Low=0,Medium=0,High=0

Low=1,Medium=0,High=0

Low=0,Medium=1,High=0

Low=1,Medium=1,High=0

Low=0,Medium=0,High=1

Low=1,Medium=0,High=1

Low=0,Medium=1,High=1

Low=1,Medium=1,High=1

——Parameter "Delay between fan speed switch [0..100]*50ms"

This parameter is visible when 1bit" is selected. Set the delay time between fan speed switchover, and consider it according to the fan technical characters. Options: **0..100**

Turn off fan speed before switch the fan speed, and turn on it after the delay time. When delay time is set as 0, it will not turn off first then turn on, but switch to next fan speed directly.

Fan speed control setting

Parameter "Control value reference from"

This parameter is for setting the reference of control value under automatic operation. Options:

PM2.5

CO2

VOC

——Parameter "Object datatype of PM2.5/VOC"

These parameters are for setting the datatype of PM2.5/VOC. Datatype determines object type, select it according to the docking PM2.5 or VOC sensor data type. Options:

Value in ug/m3(DPT 7.001)

Float value in ug/m3(DPT 9.030)

DPT_7.001: Suitable for integrated value.

DPT_9.030: Suitable for float value.

——Parameter "Object datatype of CO2"

This parameter is for setting the datatype of CO2. Datatype determines object type, select it according to the docking CO2 sensor data type. Options:

Value in ppm(DPT 7.001)

Float value in ppm(DPT 9.008)

DPT_7.001: Suitable for integrated value.

DPT_9.008: Suitable for float value.

Parameter "Time period for request control value [0..255]min"

This parameter is for setting the time period for device to send a control value read request to external sensor after bus recovery or finish programming (**After stabilization time 2min, then read**). Options: **0..255**

Parameter "The fan speed status when the control value error"

This parameter is for setting the default fan speed of ventilation system when control value is error. Options:

Off

Low

Medium

High

Parameter "Threshold value speed OFF<–>low [1..999]/ [1…4000]"

Define threshold value for off-fan and low-level fan speeds, options: **1..999/1..4000**

If the control value is greater than or equal to this setting threshold value, low-level fan speed will start running; if the control value is less than this setting threshold value, the fan will be turned off.

Parameter "Threshold value speed low<–>medium [1..999]/ [1…4000]"

Define the threshold value for switching the fan speed to medium fan speed, if the control value is greater than or equal to this setting threshold, the medium fan speed will start running. Options: **1..999/1..4000**

Parameter "Threshold value speed medium<–>high [1..999]/ [1…4000]"

Define the threshold for switching the fan speed to high fan speed, if the control value is greater than or equal to this setting threshold, the high fan speed will start running. Options: **1..999/1..4000**

Tip: The controller evaluates the threshold in ascending order.

First check →OFF <->low fan speed threshold →low fan speed <->medium fan speed →medium fan speed <->high fan speed.

The correctness of functional execution is guaranteed only in this case:

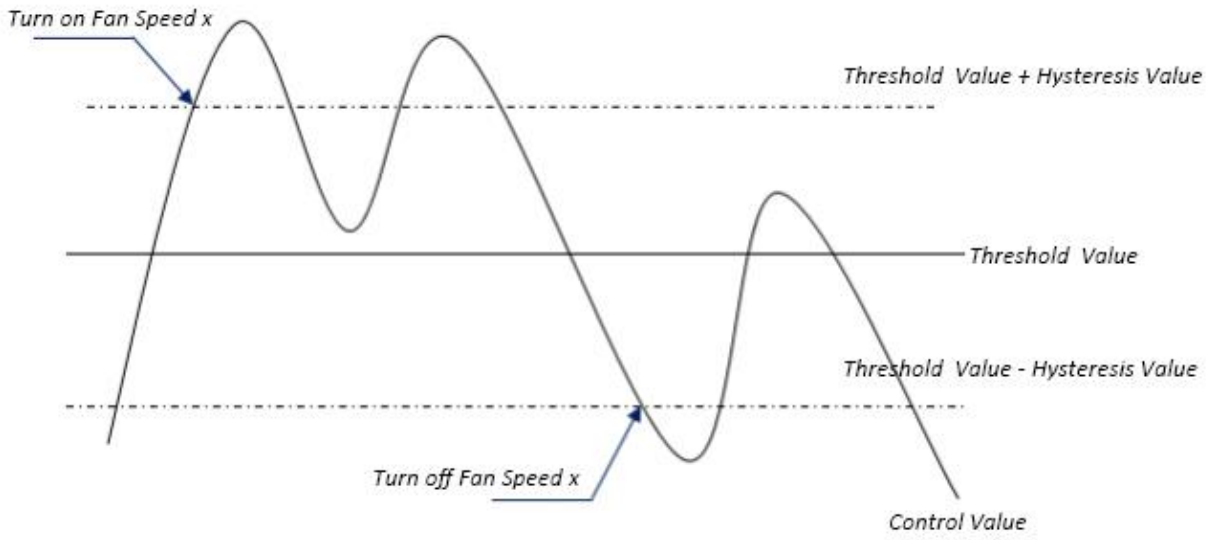
The threshold of OFF <-> low fan speed is lower than that of low fan speed <-> medium fan speed, and

the threshold of low fan speed <-> medium fan speed is lower than that of medium fan speed <-> high fan speed.

Parameter "Hysteresis threshold value in +/- [10...30]/[100..400]"

This parameter is for setting the hysteresis value of the threshold value, which can avoid the unnecessary action of the fan when the control value fluctuates near the threshold. Options: **10..30/100..400**

For example, the control type is CO2, the Hysteresis value is 100 and the threshold is 450, then the upper limit threshold 550 (Threshold value+Hysteresis value) and the lower limit threshold 350 (Threshold value-Hysteresis value). When the control value is between 350 ~550, fan action will not be caused, and the previous status will still be maintained. Only less than 350 or greater than or equal to 550 will change the running status of the fan. As shown in the following figure:



Note:

When hysteresis is enabled, if the threshold overlap occurs, fan action is specified as follows:

1) Hysteresis determines the control point where Fan speed conversion occurs;

2) If Fan speed conversion occurs, new fan speed is determined by control value and threshold value, irrespective of hysteresis.

For example (1):

Take PM2.5 as an example

OFF <-> Low fan speed threshold value is 35

Low fan speed <->Medium fan speed threshold value is 55

Medium fan speed <-> High fan speed threshold value is 75

Hysteresis value is 25

The fan speed of the fan turbine increases from OFF:

Fan OFF status will change at a control value of 60 (≥25+35), and new fan speed will be the mid-fan speed (because 60 is between 55 and 75, irrespective of hysteresis at this time), so the low fan speed is ignored;

The behavior of fan speed when descending from a high fan speed:

The high fan speed will change at a control value of 50 (<75-25), and new fan speed will be low fan speed (because 50 is between 35 and 55, irrespective of hysteresis), so the fan speed is ignored.

For example(2):

Take PM2.5 as an example

OFF <-> Low fan speed threshold value is 20

Low fan speed <->Medium fan speed threshold value is 40

Medium fan speed <-> High fan speed threshold value is 70

Hysteresis value is 10

When fan speed is increasing from OFF:

The OFF status will be turned when the control value is 30（≥20+10）

When the control value 41 is received, the new speed will be at medium(because the hysteresis is ignored when the value 41 is between 40 and 70), therefore the low speed is ignored.

When the control value 39 is received, the new speed will be at low (because the hysteresis is ignored when the value 39 is between 20 and 40)

When Fan Speed decreasing from high:

The high speed will be turned when the control value is 60（<70-10）

When the control value 39 is received, the new speed will be at low(because the hysteresis is ignored when the value 39 is between 20 and 40),therefore the medium speed is ignored.

3) When the control value is 0,the fan will be off at any circumstances.

Parameter "Minimum time in fan speed [0..65535]s"

Defines the residence time of the fan from the current fan speed to a higher fan speed or lower fan speed, that is, the minimum time for a fan speed operation. Options: **0..65535**

If you need to switch to another fan speed, you need to wait for this period of time before switching.

If the current fan speed has been running long enough, the fan speed can be changed quickly.

0: there is no minimum running time, but the delay switching time of fan speed still needs to be considered.

Note: The residence time for this parameter setting is only enabled in Auto mode.

5.5. Parameter window “Logic”

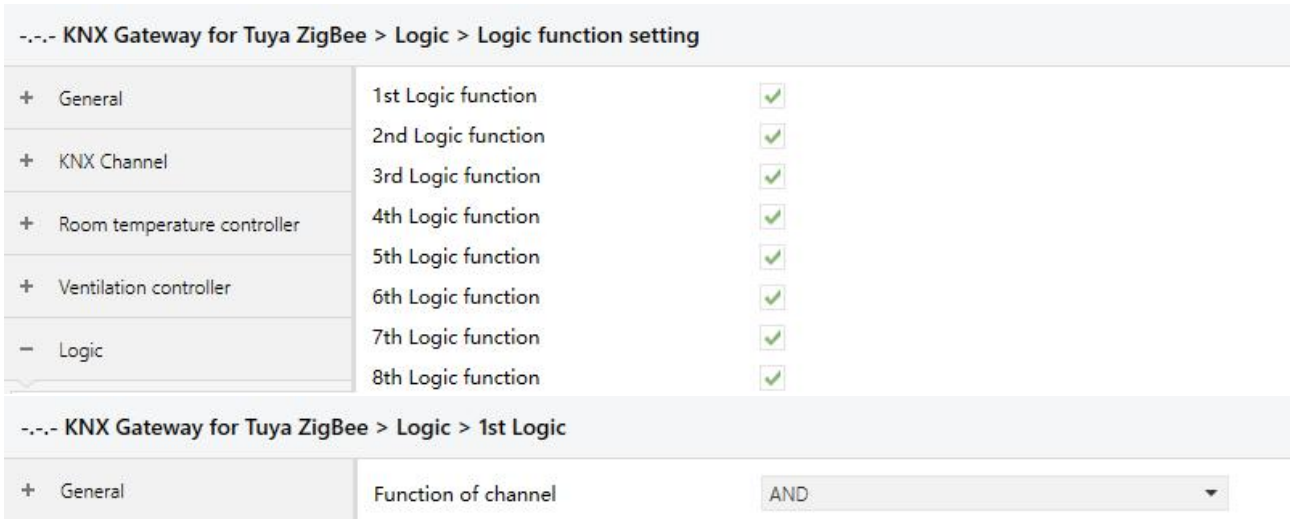


Fig.5.6 “Logic function setting” Parameter window

Parameter “1st/2nd/3rd... Logic function”

This parameter is for setting the setting interface of logic function, display corresponding logic function page when select. Up to enable 8 logic functions.

Parameter “Function of channel”

This parameter is for setting the logic function of the channel. Options:

AND

OR

XOR

Gate forwarding

Threshold comparator

Format convert

Gate function

Delay function

Staircase lighting

AND/OR/XOR: as the parameter is similar to the communication object (only the logic algorithm is different), the following parameters taking one options for example.

5.5.1. Parameter window “AND/OR/XOR”

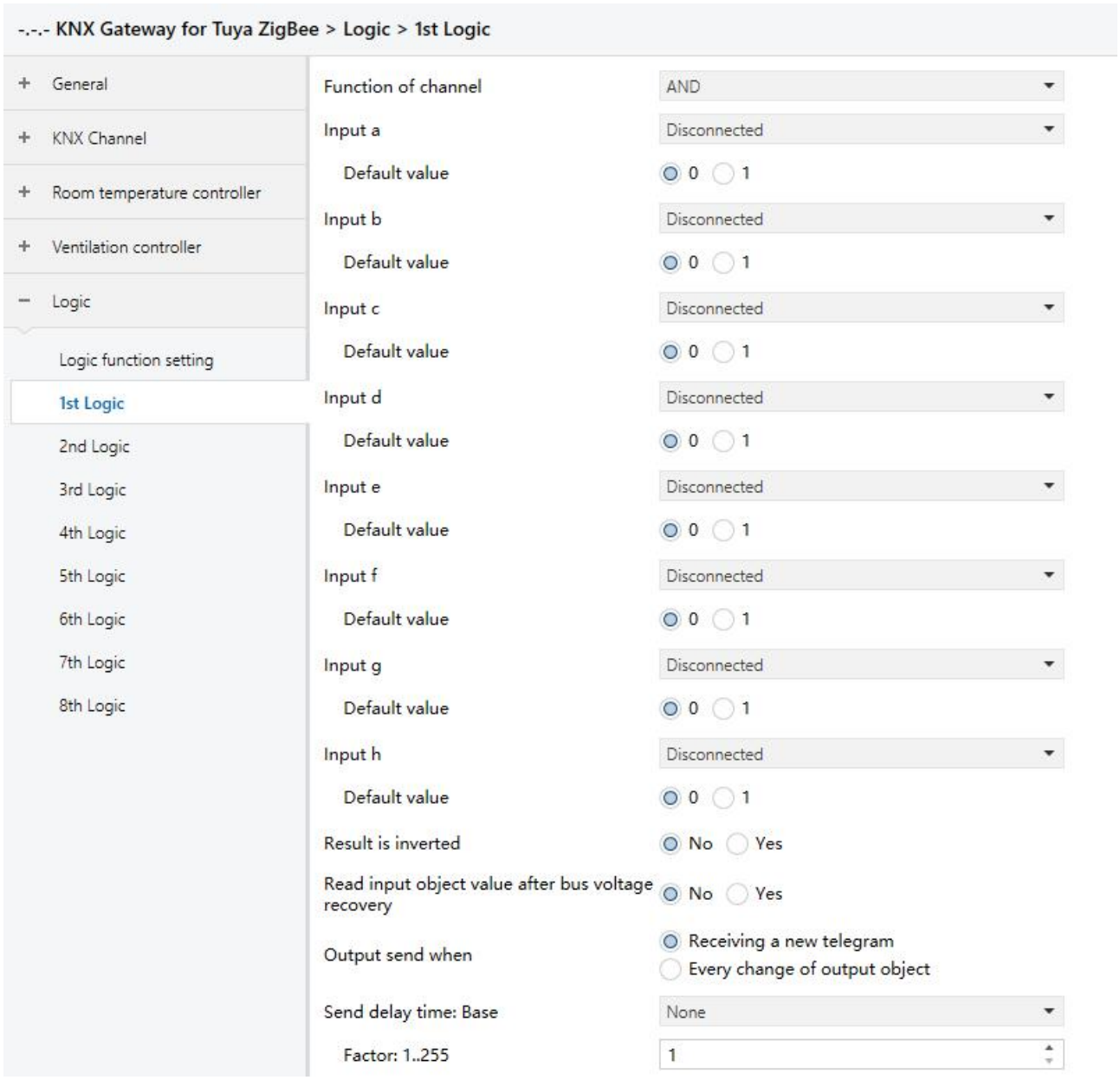


Fig.5.5.1 “AND/OR/XOR” parameter window

Parameter “Input a/b/c/d/e/f/g/h”

This parameter is for setting whether input x to calculate, whether to normally calculate or inverted calculate.Options:

Disconnected

Normal

Inverted

Disconnected: not to calculate;

Normal: to directly calculate the input value;

Inverted: invert the input value, then to calculate. **Note: not to invert the initiate value.**

——Parameter "Default value"

This parameter is for setting the initial value of logic input x. Options:

0

1

Parameter "Result is inverted"

This parameter is for setting whether to invert the logic calculation result. Options:

No

Yes

No: output directly;

Yes: output after inverting.

Parameter "Read input object value after bus voltage recovery"

This parameter is for setting whether to send the read request to the logic input object after device voltage recovery or finish programming. Options:

No

Yes

Parameter "Output send when"

This parameter is for setting the condition of sending logic result. Options:

Receiving a new telegram

Every change of output object

Receiving a new telegram：every time the object received a new input value will the logic result be sent to the bus;

Every change of output object： only when logic result has changed will it be sent to the bus.

Tip: when in the first time to logic calculate, the logic result will be sent even if it has no change.

Parameter "Send delay time"

Base: **None**

0.1s

1s

...

10s

25s

Factor: 1..255

This parameter is for setting the delay time for sending the logic calculation result to the bus. Delay time = Base × Factor, if option "None" of Base is selected, then there is no delay.

5.5.2. Parameter window "Gate forwarding"

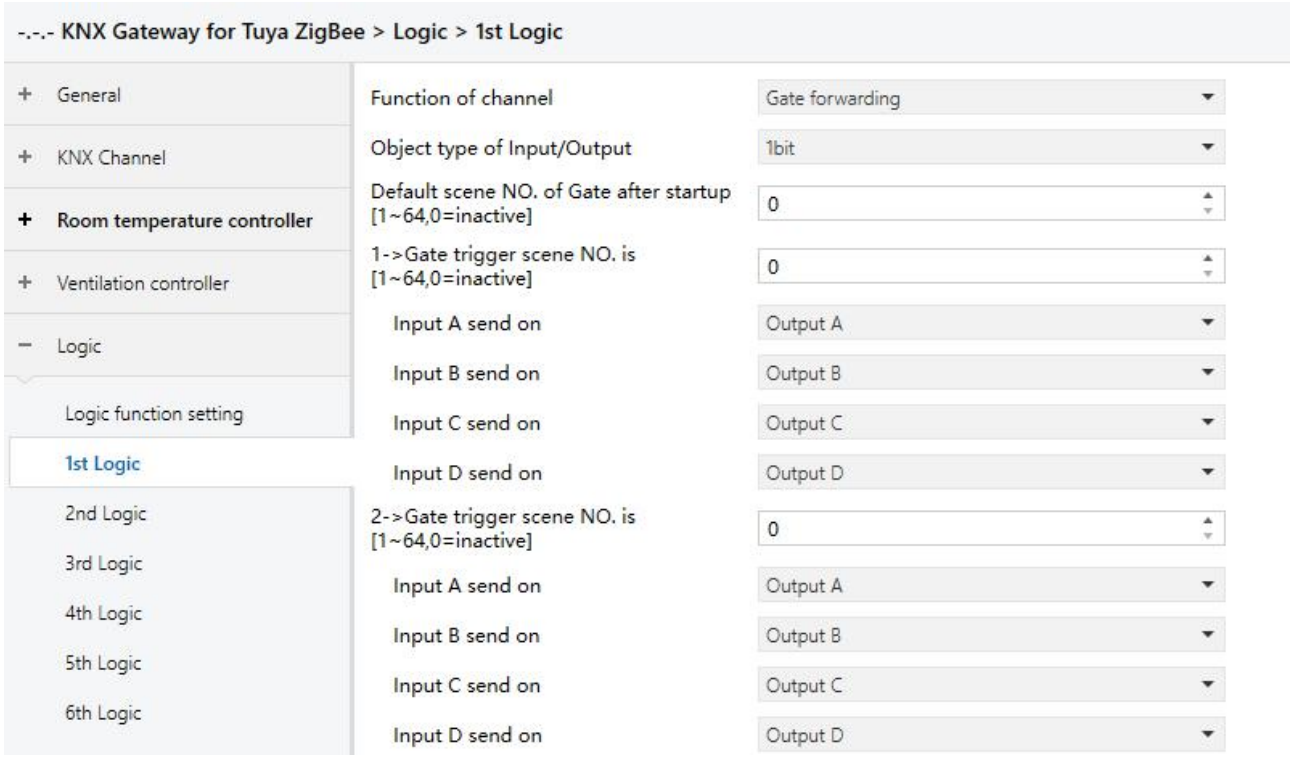


Fig.5.5.2 "Gate forwarding" parameter window

Parameter "Object type of Input/Output"

This parameter is for setting the object type of input/output. Options:

1bit

4bit

1byte

Parameter "Default scene NO. of Gate after startup [1~64,0=inactive]"

This parameter is for setting the initial scene where logical gate forwarding can be performed by default after device starts, which needs to be configured in the parameters. Options: **1..64, 0=inactive**

Note: gate scene is recommended to be selected before operating, or it will enable the initiate scene by default.

Parameter "z->Gate trigger scene NO. is [1~64,0=inactive]"(z=1~8)

This parameter is for setting scene number of logic gate forwarding. Up to 8 trigger scene number can be set for each logic. Options: **1..64, 0=inactive**

——Parameter "Input A/B/C/D send on"

This parameter is for setting the output of input X (X=A/B/C/D) after gate forwarding. Options:

Output A

Output B

...

Output B,C,D

According to the options, one input can be forwarded into one or more outputs, the output value is the same as the input value.

5.5.3. Parameter window "Threshold comparator"

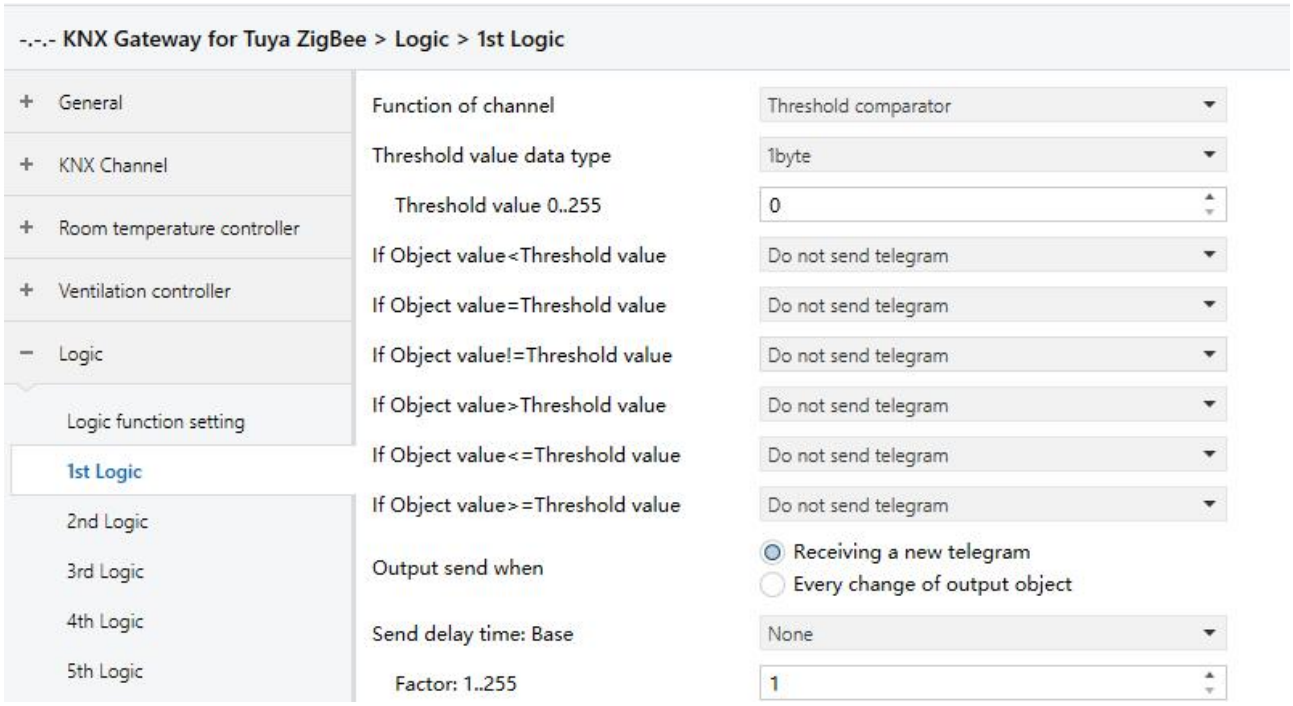


Fig.5.5.3 "Threshold comparator" parameter window

Parameter "Threshold value data type"

This parameter is for setting the threshold value data type. Options:

4bit

1byte

2byte

4byte

Parameter "Threshold value 0..255"

This parameter is for setting threshold value, the range depends on the data type. Options: **4bit 0..15/1byte 0..255/ 2byte 0..65535 /4byte 0..4294967295**

Parameter "If Object value<Threshold value"

Parameter "If Object value!=Threshold value"

Parameter "If Object value<Threshold value"

Parameter "If Object value<Threshold value"

Parameter "If Object value<Threshold value"

This parameter is for setting the logic result value that should be sent when threshold value Less than, equal to, not equal to, greater than, less than or equal to the setting valve. Options:

Do not send telegram

Send value "0"

Send value "1"

Do not send telegram: not consider to select this option;

Send value "0"/"1": when condition is satisfied, send telegram 0 or1.

If there is a conflict between the setting options between parameters, the base on the value that should be sent when reach the final parameter condition. **For example: parameter "If Object value=Threshold value" is set to be "Send value "0" "; parameter "If Object value<=Threshold value" is set to be "Send value "1" "; when object value is equal to the threshold value, then the logic result will send "1".**

Parameter "Output send when"

This parameter is for setting the condition of sending logic result. Options:

Receiving a new telegram

Every change of output object

Receiving a new telegram: every time the object received a new input value will the logic result be sent to the bus;

Every change of output object: only when logic result has changed will it be sent to the bus.

Tip: when in the first time to logic algorithm, the logic result will be sent even if it has no change.

Parameter "Send delay time"

Base: **None**

0.1s

1s

...

10s

25s

Factor: **1..255**

This parameter is for setting the delay time for sending the logic algorithm result to the bus. Delay time = Base x Factor, if option "None" of Base is selected, then there is no delay.

5.5.4. Parameter window "Format convert"

Fig.5.5.4 "Format convert" parameter window

Parameter "Function"

This parameter is for setting the format convert type. Options:

2x1bit-->1x2bit

8x1bit-->1x1byte

1x1byte-->1x2byte

2x1byte-->1x2byte

2x2byte-->1x4byte

1x1byte-->8x1bit

1x2byte-->2x1byte

1x4byte-->2x2byte

1x3byte-->3x1byte

3x1byte-->1x3byte

Parameter "Output send when"

This parameter is for setting the condition of sending logic result. Options:

Receiving a new telegram

Every change of output object

Receiving a new telegram: every time the object received a new input value will the logic result be sent to the bus;

Every change of output object: only when logic result has changed will it be sent to the bus.

Tip: when in the first time to logic algorithm, the logic result will be sent even if it has no change.

5.5.5. Parameter window "Gate function"

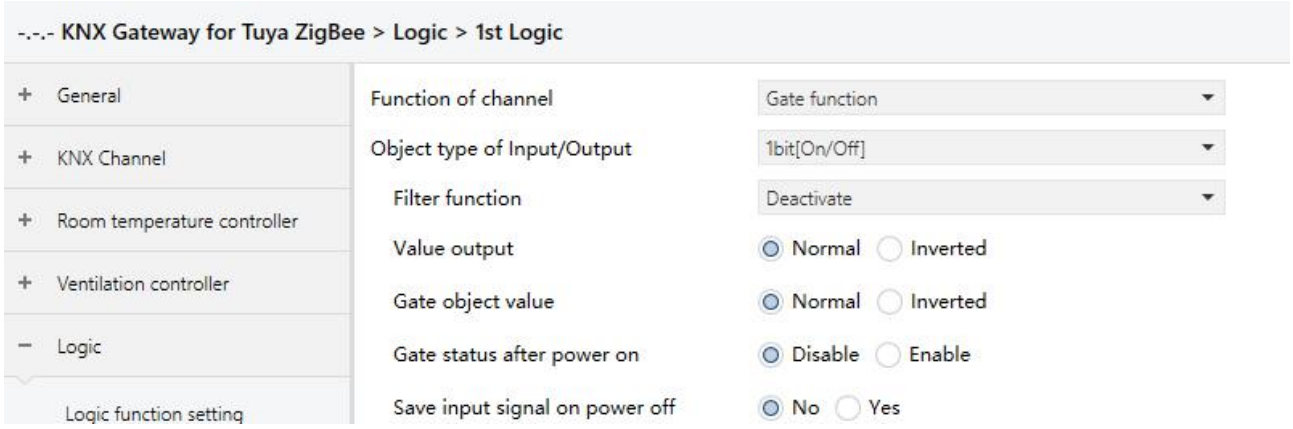


Fig.5.5.5 "Gate function" parameter window

Parameter "Object type of Input/Output"

This parameter is for setting the object type of input/output. Options:

1bit[On/Off]

1byte[0..100%]

1byte[0..255]

2byte[Float]

2byte[0..65535]

——Parameter "Filter function"

This parameter is visible when "1bit[On/Off]" is selected. Set whether to filter On or Off telegram, only pass one of them or pass all. Options:

Deactivate

On filter out

Off filter out

Deactivate: Do not filter the On or Off telegrams;

On filter out: Off can pass, On cannot pass;

Off filter out: On can pass, Off cannot pass.

——Parameter "Value output"

This parameter is visible when "1bit[On/Off]" is selected. Set whether to invert the value then output it. Options:

Normal

Inverted

Parameter "Gate object value"

This parameter is for setting whether to invert the gate object value then output it. Options:

Normal

Inverted

Parameter "Gate status after power on"

This parameter is for setting the gate status after power on. Options:

Disable

Enable

Parameter "Save input signal on power off"

This parameter is for setting whether to save input signal on power off. Options:

No

Yes

5.5.6. Parameter window “Delay function”

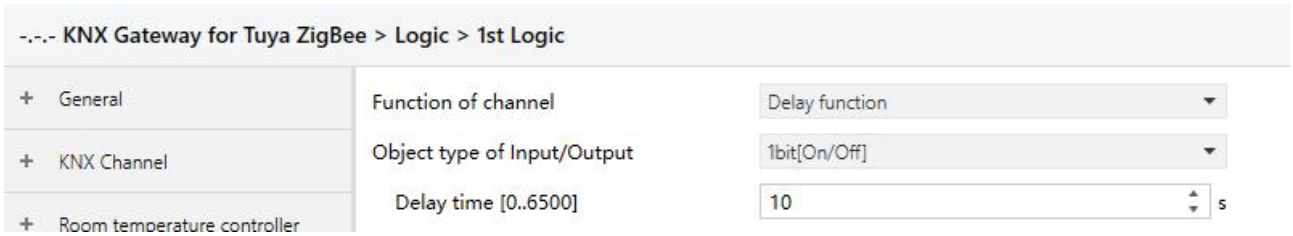


Fig.5.5.6 “Delay function” parameter window

Parameter “Object type of Input/Output”

This parameter is for setting the object type of input/output. Options:

1bit[On/Off]

1byte[0..100%]

1byte[0..255]

2byte[Float]

2byte[0..65535]

——Parameter “Delay time [0..6500]s”

This parameter is for setting the delay time that output object forwards the value when the input object receives the telegram. Options: **0..6500**

Note: Receive telegram again in delay time, re-timing.

5.5.7. Parameter window “Staircase lighting”

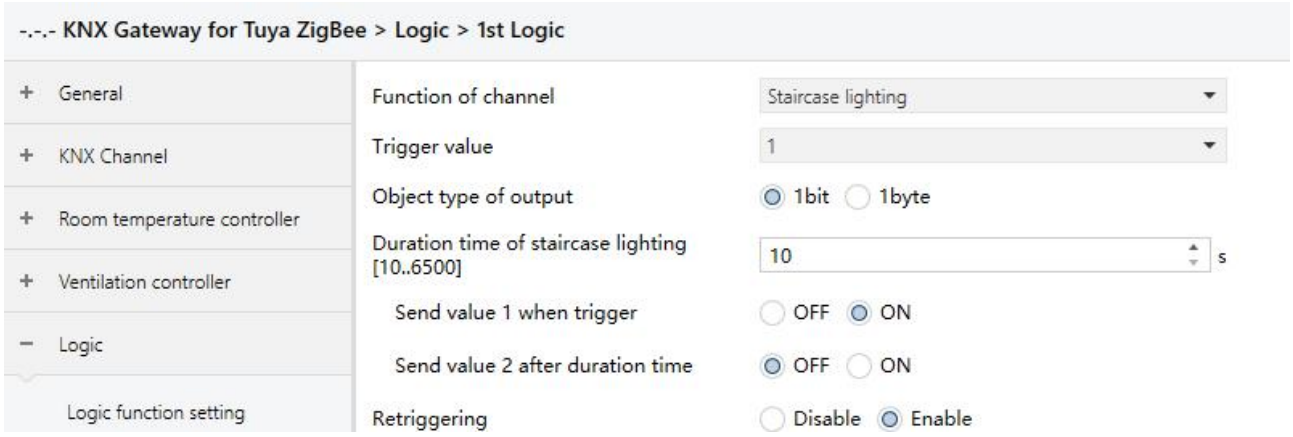


Fig.5.5.7 “Staircase lighting” parameter window

Parameter "Trigger value"

This parameter is for setting the telegram value of the object "Trigger value". Options:

0

1

0 or 1

Parameter "Object type of output"

This parameter is for setting the object type of output. Options:

1bit

1byte

Parameter "Duration time of staircase lighting[10..6500]s"

This parameter is for setting duration time of staircase lighting after the stair light power on.

Options: **10..6500**

——Parameter "Send value 1 when trigger"

——Parameter "Send value 2 after duration time"

These parameters are for setting the value to send. Send value 1 when trigger, and then send value 2 after duration time. Options display according to the output object datatype.

When 1 bit, options:

OFF

ON

When 1 byte , options: **0..255**

Parameter "Retriggering"

This parameter is for setting whether to trigger re-timing when received trigger value in delay time. Options:

Disable

Enable

Chapter 6 Description of Communication Object

The communication object is the medium to communicate other device on the bus, namely only the communication object can communicate with the bus.

NOTE: "C" in "Flag" column in the below table means enable the communication function of the object; "W" means value of object can be written from the bus; "R" means the value of the object can be read by the other devices; "T" means the object has the transmission function; "U" means the value of the object can be updated.

6.1. "General" Communication Object

Number	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
1	General	In operation			1 bit	C	R	-	T	-	switch	Low

Fig.6.1 "General" Communication Object

NO.	Object Function	Name	Data Type	Flag	DPT
1	**In operation**	**General**	**1bit**	**C,R,T**	**1.001 switch**
The communication object is used to periodically send a telegram "1" to the bus to indicate that the device is working properly.					

Table 6.1 "General" communication object table

6.2. "KNX Channel" Communication Object

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
300	KNX General	Device online common 1, status			1 bit	C	-	W	T	U	switch	Low
301	KNX General	Device online common 2, status			1 bit	C	-	W	T	U	switch	Low
302	KNX General	Device online common 3, status			1 bit	C	-	W	T	U	switch	Low
303	KNX General	Device online common 4, status			1 bit	C	-	W	T	U	switch	Low
304	KNX General	Device online common 5, status			1 bit	C	-	W	T	U	switch	Low
305	KNX General	Device online common 6, status			1 bit	C	-	W	T	U	switch	Low
306	KNX General	Device online common 7, status			1 bit	C	-	W	T	U	switch	Low
307	KNX General	Device online common 8, status			1 bit	C	-	W	T	U	switch	Low
308	KNX General	Device online common 9, status			1 bit	C	-	W	T	U	switch	Low
309	KNX General	Device online common 10, status			1 bit	C	-	W	T	U	switch	Low
310	KNX: Channel 1	Device online, status			1 bit	C	-	W	T	U	switch	Low

Fig.6.2 KNX general communication Object

NO.	Object Function	Name	Data Type	Flag	DPT
300/.../309	**Device online common x, status**	**KNX General**	**1bit**	**C,W,T,U**	**1.001 switch**
The communication object is used to send the request to the bus, for common device to query the KNX device online status, and also can receive the feedback. Telegrams:					

	0——Offline 1——Online Suitable for devices with multiple circuits. For example, multiple channels of gateway may be multiple circuits that control the same KNX device, so each channel can share one request.				
310	**Device online, status**	**KNX: {{Channel 1}}**	**1bit**	**C,W,T,U**	**1.001 switch**
	The communication object is used to send the request to the bus, for independent device to query the KNX device online status, and also can receive the feedback. Telegrams: 0——Offline 1——Online Suitable for a device that only controlled by the single channel of gateway. The name in parentheses changes with the parameter "Description (max 30 char.)". If description is empty, display "...Channel x" by default. The same below.				

Table 6.2 KNX general communication object table

6.2.1. Communication Object of basic function

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
311	KNX: Channel 1	Switch			1 bit	C	-	-	T	-	switch	Low
312	KNX: Channel 1	Switch, status			1 bit	C	-	W	T	U	switch	Low

Switch

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
311	KNX: Channel 1	Switch			1 bit	C	-	-	T	-	switch	Low
312	KNX: Channel 1	Brightness dimming			1 byte	C	-	-	T	-	percentage (0..100%)	Low
313	KNX: Channel 1	Brightness, status			1 byte	C	-	W	T	U	percentage (0..100%)	Low

Switch/Dimming

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
311	KNX: Channel 1	Switch			1 bit	C	-	-	T	-	switch	Low
312	KNX: Channel 1	Switch, status			1 bit	C	-	W	T	U	switch	Low
313	KNX: Channel 1	RGB dimming value			3 bytes	C	-	-	T	-	RGB value 3x(0..255)	Low
317	KNX: Channel 1	RGB brightness, status			3 bytes	C	-	W	T	U	RGB value 3x(0..255)	Low
313	KNX: Channel 1	Red dimming value			1 byte	C	-	-	T	-	percentage (0..100%)	Low
314	KNX: Channel 1	Green dimming value			1 byte	C	-	-	T	-	percentage (0..100%)	Low
315	KNX: Channel 1	Blue dimming value			1 byte	C	-	-	T	-	percentage (0..100%)	Low
317	KNX: Channel 1	Red brightness, status			1 byte	C	-	W	T	U	percentage (0..100%)	Low
318	KNX: Channel 1	Green brightness, status			1 byte	C	-	W	T	U	percentage (0..100%)	Low
319	KNX: Channel 1	Blue brightness, status			1 byte	C	-	W	T	U	percentage (0..100%)	Low

RGB dimming

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
311	KNX: Channel 1	Switch			1 bit	C	-	-	T	-	switch	Low
312	KNX: Channel 1	Switch, status			1 bit	C	-	W	T	U	switch	Low
313	KNX: Channel 1	RGBW dimming value			6 bytes	C	-	-	T	-	RGBW value 4x(0..100%)	Low
317	KNX: Channel 1	RGBW brightness, status			6 bytes	C	-	W	T	U	RGBW value 4x(0..100%)	Low

RGBW dimming(1)

313	KNX: Channel 1	Red dimming value		1 byte	C	-	-	T	-	percentage (0..100%)	Low
314	KNX: Channel 1	Green dimming value		1 byte	C	-	-	T	-	percentage (0..100%)	Low
315	KNX: Channel 1	Blue dimming value		1 byte	C	-	-	T	-	percentage (0..100%)	Low
316	KNX: Channel 1	White dimming value		1 byte	C	-	-	T	-	percentage (0..100%)	Low
317	KNX: Channel 1	Red brightness, status		1 byte	C	-	W	T	U	percentage (0..100%)	Low
318	KNX: Channel 1	Green brightness, status		1 byte	C	-	W	T	U	percentage (0..100%)	Low
319	KNX: Channel 1	Blue brightness, status		1 byte	C	-	W	T	U	percentage (0..100%)	Low
320	KNX: Channel 1	White brightness, status		1 byte	C	-	W	T	U	percentage (0..100%)	Low

RGBW dimming(2)

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
311	KNX: Channel 1	Switch			1 bit	C	-	-	T	-	switch	Low
312	KNX: Channel 1	Switch, status			1 bit	C	-	W	T	U	switch	Low
313	KNX: Channel 1	Color temperature value			2 bytes	C	-	-	T	-	absolute colour temperature (K)	Low
314	KNX: Channel 1	Brightness value			1 byte	C	-	-	T	-	percentage (0..100%)	Low
317	KNX: Channel 1	Color temperature, status			2 bytes	C	-	W	T	U	absolute colour temperature (K)	Low
318	KNX: Channel 1	Brightness, status			1 byte	C	-	W	T	U	percentage (0..100%)	Low

Color temperature

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
311	KNX: Channel 1	Open/Close			1 bit	C	-	-	T	-	open/close	Low
312	KNX: Channel 1	Stop			1 bit	C	-	-	T	-	step	Low

Curtain step/move

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
311	KNX: Channel 1	Up/Down			1 bit	C	-	-	T	-	up/down	Low
312	KNX: Channel 1	Stop			1 bit	C	-	-	T	-	step	Low

Roller blind step/move

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
311	KNX: Channel 1	Open/Close			1 bit	C	-	-	T	-	open/close	Low
312	KNX: Channel 1	Stop			1 bit	C	-	-	T	-	step	Low
313	KNX: Channel 1	Blind position			1 byte	C	-	-	T	-	percentage (0..100%)	Low
315	KNX: Channel 1	Blind position, status			1 byte	C	-	W	T	U	percentage (0..100%)	Low

Curtain position

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
311	KNX: Channel 1	Up/Down			1 bit	C	-	-	T	-	up/down	Low
312	KNX: Channel 1	Stop			1 bit	C	-	-	T	-	step	Low
313	KNX: Channel 1	Blind position			1 byte	C	-	-	T	-	percentage (0..100%)	Low
315	KNX: Channel 1	Blind position, status			1 byte	C	-	W	T	U	percentage (0..100%)	Low

Roller blind position

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
311	KNX: Channel 1	Up/Down			1 bit	C	-	-	T	-	open/close	Low
312	KNX: Channel 1	Stop/Slat adj.			1 bit	C	-	-	T	-	step	Low
313	KNX: Channel 1	Blind position			1 byte	C	-	-	T	-	percentage (0..100%)	Low
314	KNX: Channel 1	Slat position			1 byte	C	-	-	T	-	percentage (0..100%)	Low
315	KNX: Channel 1	Blind position, status			1 byte	C	-	W	T	U	percentage (0..100%)	Low
316	KNX: Channel 1	Slat position, status			1 byte	C	-	W	T	U	percentage (0..100%)	Low

Venetian blind position and slat

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
311	KNX: Channel 1	Send 1bit value			1 bit	C	-	-	T	-	switch	Low
311	KNX: Channel 1	Send 2bit value			2 bit	C	-	-	T	-	switch control	Low
311	KNX: Channel 1	Send 1byte percent value			1 byte	C	-	-	T	-	percentage (0..100%)	Low
311	KNX: Channel 1	Send 1byte unsigned value			1 byte	C	-	-	T	-	counter pulses (0..255)	Low
311	KNX: Channel 1	Recall scene No.			1 byte	C	-	-	T	-	scene number	Low
311	KNX: Channel 1	Send 2byte float value			2 bytes	C	-	-	T	-	2-byte float value	Low
311	KNX: Channel 1	Send 2byte unsigned value			2 bytes	C	-	-	T	-	pulses	Low

Value sender

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
311	KNX: Channel 1	Trigger scene No.			1 byte	C	-	W	-	-	scene number	Low

Scene switch

Fig.6.2.1 Basic function communication object

NO.	Object Function	Name	Data Type	Flag	DPT
311	**Switch**	**KNX: {{Channel 1}}**	**1bit**	**C,T**	**1.001 switch**
312	**Switch, status**	**KNX: {{Channel 1}}**	**1bit**	**C,W,T,U**	**1.001 switch**

These two communication objects apply to switch, dimming, colour and colour temperature control. Telegrams:

0——Turn off the light

1——Turn on the light

Obj.311: Used for sending On/Off telegrams to the bus, to control the light on/off.

Obj.312: Used for receiving On/Off status responded from other bus devices, such as dimming actuator, switch actuator.

312	**Brightness dimming**	**KNX: {{Channel 1}}**	**1byte**	**C,T**	**5.001 percentage(0..100%)**
313	**Brightness, status**	**KNX: {{Channel 1}}**	**1byte**	**C,W,T,U**	**5.001 percentage(0..100%)**

These two communication objects apply to brightness dimming. Telegrams: 0~100%

Obj.312: Used for sending dimming telegrams to the bus, that is, sending the brightness value.

Obj.313: Used for receiving brightness status responded from dimming actuator.

313	**Red dimming value**	**KNX: {{Channel 1}}**	**1byte**	**C,T**	**5.001 percentage(0..100%)**
317	**Red brightness, status**	**KNX: {{Channel 1}}**	**1byte**	**C,W,T,U**	**5.001 percentage(0..100%)**

These two communication objects are visible when 3x1byte for the RGB object type or 4x1byte for the RGBW object type is selected. Apply to control brightness of multi-colour lamp, and also support colour temperature adjustment. Telegrams: 0...100%

Obj.313 Used for sending brightness value of the control R (red) channel to the bus.

Obj.317: Used for receiving brightness value of the control R (red) channel from bus.

314	**Green dimming value**	**KNX: {{Channel 1}}**	**1byte**	**C,T**	**5.001 percentage(0..100%)**
318	**Green brightness, status**	**KNX: {{Channel 1}}**	**1byte**	**C,W,T,U**	**5.001 percentage(0..100%)**

These two communication objects are visible when 3x1byte for the RGB object type or 4x1byte for the RGBW object type is selected. Apply to control brightness of multi-colour lamp, and also support colour temperature adjustment. Telegrams: 0...100%

Obj.314: Used for sending brightness value of the control G (green) channel to the bus.

Obj.318: Used for receiving brightness value of the control G (green) channel from bus.

315	**Blue dimming value**	**KNX: {{Channel 1}}**	**1byte**	**C,T**	**5.001 percentage(0..100%)**
319	**Blue brightness, status**	**KNX: {{Channel 1}}**	**1byte**	**C,W,T,U**	**5.001 percentage(0..100%)**

These two communication objects are visible when 3x1byte for the RGB object type or 4x1byte for the RGBW object type is selected. Apply to control brightness of multi-colour lamp, and also support colour temperature adjustment. Telegrams: 0...100%

Obj.315: Used for sending brightness value of the control B (blue) channel to the bus.

Obj.319: Used for receiving brightness value of the control B (blue) channel from bus.

316	**White dimming value**	**KNX: {{Channel 1}}**	**1byte**	**C,T**	**5.001 percentage(0..100%)**
320	**White brightness, status**	**KNX: {{Channel 1}}**	**1byte**	**C,W,T,U**	**5.001 percentage(0..100%)**

These two communication objects are visible when 4x1byte for the RGBW object type is selected. Apply to control brightness of multi-colour lamp, and also support colour temperature adjustment. Telegrams: 0...100%

Obj.316: Used for sending brightness value of the control W (white) channel to the bus.

Obj.320: Used for receiving brightness value of the control W (white) channel from bus.

313	**RGB dimming value**	**KNX: {{Channel 1}}**	**3byte**	**C,T**	**232.600 RGB value 3x(0..255)**
317	**RGB brightness, status**	**KNX: {{Channel 1}}**	**3byte**	**C,W,T,U**	**232.600 RGB value 3x(0..255)**

These two communication objects are visible when 1x3byte for the RGB object type is selected. Apply to control brightness of multi-colour lamp, and also support colour temperature adjustment.

Obj.313: Used for sending brightness value of RGB three-colour lamp to the bus.

Obj.317: Used for receiving brightness value of RGB three-colour lamp from bus.

3-Byte Code for RGB Dimming Object Data Type: U8 U8 U8, as follows:

3_{MSB}	2	1_{LSB}
R	G	B
UUUUUUUU	UUUUUUUU	UUUUUUUU

R: red dimming value; G: green dimming value; B: blue dimming value.

313	**RGBW dimming value**	**KNX: {{Channel 1}}**	**6byte**	**C,T**	**251.600 DPT_Colour_RGBW**
317	**RGBW brightness, status**	**KNX: {{Channel 1}}**	**6byte**	**C,W,T,U**	**251.600 DPT_Colour_RGBW**

These two communication objects are visible when 1x6byte for the RGBW object type is selected. Apply to control brightness of multi-colour lamp, and also support colour temperature adjustment.

6_{MSB}	5	4	3	2	1_{LSB}
R	G	B	W	Reserve	r r r r mR mG mB mW
UUUUUUUU	UUUUUUUU	UUUUUUUU	UUUUUUUU	00000000	0000BBBB

R: red dimming value; G: green dimming value; B: blue dimming value; W: white dimming value;

mR: determines whether the red dimming value is valid, 0 = invalid, 1 = valid;

mG: determines whether the green dimming value is valid, 0 = invalid, 1 = valid;

mB: determines whether the blue dimming value is valid, 0 = invalid, 1 = valid;

mW: Determines whether the white dimming value is valid,0 = invalid,1 =valid.

313	**Color temperature value**	**KNX: {{Channel 1}}**	**2byte**	**C,T**	**7.600 absolute color temperature**
317	**Color temperature, status**	**KNX: {{Channel 1}}**	**2byte**	**C,W,T,U**	**7.600 absolute color temperature**

These two communication objects apply to colour temperature adjustment of monochrome lamp. Telegrams: 1000...10000 K

Obj.313: Used for sending the control telegram of the colour temperature to the bus.

Obj.317: Used for receiving the control telegram of the colour temperature from bus.

314	**Brightness value**	**KNX: {{Channel 1}}**	**1byte**	**C,T**	**5.001 percentage(0..100%)**
318	**Brightness, status**	**KNX: {{Channel 1}}**	**1byte**	**C,W,T,U**	**5.001 percentage(0..100%)**

These two communication objects apply to colour temperature adjustment of monochrome lamp. Telegrams: 0…100%

Obj.314: Used for sending the dimming telegram of the colour temperature to the bus, that is, sending the brightness value.

Obj.318: Used for receiving the brightness status responded from the dimming actuator.

311	**Open/Close**	**KNX: {{Channel 1}}**	**1bit**	**C,T**	**1.009 open/close**
312	**Stop**	**KNX: {{Channel 1}}**	**1bit**	**C,T**	**1.007 step**

Curtain step/move: these two communication objects apply to open and close curtain. Support to open, close, stop.

Obj.311: Used for sending the telegram to the bus, to control curtain open/close. Telegrams:

1——Close the curtain

0——Open the curtain

Obj.312: Used for sending the telegram to the bus, to stop curtain movement. Telegrams:

1——Stop

311	**Up/Down**	**KNX: {{Channel 1}}**	**1bit**	**C,T**	**1.008 up/down**
312	**Stop**	**KNX: {{Channel 1}}**	**1bit**	**C,T**	**1.007 step**

Roller blind step/move: these two communication objects apply to roller blind. Support to up, down, stop.

Obj.311: Used for sending the telegram to the bus, to control blind up/down. Telegrams:

1——Move down

0——Move up

Obj.312 is the same as above.

311	**Open/Close**	**KNX: {{Channel 1}}**	**1bit**	**C,T**	**1.009 open/close**
312	**Stop**	**KNX: {{Channel 1}}**	**1bit**	**C,T**	**1.007 step**
313	**Blind position**	**KNX: {{Channel 1}}**	**1byte**	**C,T**	**5.001 percentage(0..100%)**
315	**Blind position, status**	**KNX: {{Channel 1}}**	**1byte**	**C,W,T,U**	**5.001 percentage(0..100%)**

Curtain position: apply to open and close curtain. Support to open, close, stop, position adjustment and position status feedback.

Obj.311: Used for sending the telegram to the bus, to control curtain open/close. Telegrams:

1——Close the curtain

0——Open the curtain

Obj.312: Used for sending the telegram to the bus, to stop curtain movement. Telegrams:

1——Stop

Obj.313: Used for sending a telegram to control the position of the curtain to the bus. Telegrams: 0…100%

Obj.315: Used for receiving a curtain position status in response to the window curtain actuator on the bus. Telegrams: 0…100%

311	**Up/Down**	**KNX: {{Channel 1}}**	**1bit**	**C,T**	**1.008 up/down**
312	**Stop**	**KNX: {{Channel 1}}**	**1bit**	**C,T**	**1.007 step**
313	**Blind position**	**KNX: {{Channel 1}}**	**1byte**	**C,T**	**5.001 percentage(0..100%)**
315	**Blind position, status**	**KNX: {{Channel 1}}**	**1byte**	**C,W,T,U**	**5.001 percentage(0..100%)**

Roller blind position: apply to a roller blind without slat. Support to up, down, stop, position adjustment and position status feedback.

Obj.311: Used for sending the telegram to the bus, to control blind up/down. Telegrams:

1——Move down

0——Move up

Obj.312, Obj.313, Obj.315 are the same as above.

311	**Up/Down**	**KNX: {{Channel 1}}**	**1bit**	**C,T**	**1.008 up/down**
312	**Stop/Slat adj.**	**KNX: {{Channel 1}}**	**1bit**	**C,T**	**1.007 step**
313	**Blind position**	**KNX: {{Channel 1}}**	**1byte**	**C,T**	**5.001 percentage(0..100%)**
314	**Slat position**	**KNX: {{Channel 1}}**	**1byte**	**C,T**	**5.001 percentage(0..100%)**
315	**Blind position, status**	**KNX: {{Channel 1}}**	**1byte**	**C,W,T,U**	**5.001 percentage(0..100%)**
316	**Slat position, status**	**KNX: {{Channel 1}}**	**1byte**	**C,W,T,U**	**5.001 percentage(0..100%)**

Venetian blind position and slat: apply to a blind with slat. Support to up, down, stop, position and slat adjustment, position and slat status feedback.

Obj.311、Obj.313 and Obj.315 are same as above.

Obj.312: Used for sending a telegram to the bus to stop the curtain movement or adjust the slat angle. Telegrams:

1——Stop/Slat adj. Down

0——Stop/Slat adj. Up

Obj.314: Used for sending a telegram to control the position of the blind to the bus. Telegrams: 0…100%

Obj.316: Used for receiving a blind position status in response to the blind actuator on the bus. Telegrams: 0…100%

311	**Send 1bit value** **Send 2bit value** **Send 1byte percent value** **Send 1byte unsigned value** **Recall scene No.** **Send 2byte float value** **Send 2byte unsigned value**	**KNX: {{Channel 1}}**	**1bit on/off** **2bit 0..3** **1byte 0..100%** **1byte 0..255** **1byte 1..64** **2byte -671088.64..670760.96** **2byte 0..65535**	**C, T**	**1.001 switch** **2.001 switch control** **5.001 percentage(0..100%)** **5.010 counter pulses** **17.001 scene number** **9.x float value** **7.001 pulses**
The communication object is used for sending a preset telegram to the bus when trigger the calling command on the APP. Object type and value range are determined by the parameter setting datatype.					
311	**Trigger scene No.**	**KNX: {{Channel 1}}**	**2byte**	**C,W**	**17.001 scene number**
The communication object is used for receiving the scene triggered telegram from KNX system. Telegram: 1..64					

Table 6.2.1 Basic function communication object table

6.2.2. Communication Object of Air condition

	Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
	311	KNX: Channel 1	Power on/off			1 bit	C	-	-	T	-	switch	Low
	312	KNX: Channel 1	Current setpoint adjustment			2 bytes	C	-	-	T	-	temperature (°C)	Low
	313	KNX: Channel 1	Control mode			1 byte	C	-	-	T	-	HVAC control mode	Low
	314	KNX: Channel 1	Fan speed			1 byte	C	-	-	T	-	percentage (0..100%)	Low
	317	KNX: Channel 1	Power on/off, status			1 bit	C	-	W	-	-	switch	Low
	318	KNX: Channel 1	Room temperature sensor			2 bytes	C	-	W	T	U	temperature (°C)	Low
	319	KNX: Channel 1	Current temperature setpoint, status			2 bytes	C	-	W	-	U	temperature (°C)	Low
	320	KNX: Channel 1	Control mode, status			1 byte	C	-	W	-	-	HVAC control mode	Low
	321	KNX: Channel 1	Fan speed, status			1 byte	C	-	W	-	-	percentage (0..100%)	Low

Air conditioner

	Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
	311	KNX: Channel 1	Power on/off			1 bit	C	-	-	T	-	switch	Low
	312	KNX: Channel 1	Current setpoint adjustment			2 bytes	C	-	-	T	-	temperature (°C)	Low
	313	KNX: Channel 1	Control mode			1 byte	C	-	-	T	-	HVAC control mode	Low
	314	KNX: Channel 1	Fan speed			1 byte	C	-	-	T	-	percentage (0..100%)	Low
	315	KNX: Channel 1	Vanes swing (1-swing,0-stop)			1 bit	C	-	-	T	-	start/stop	Low
	317	KNX: Channel 1	Power on/off, status			1 bit	C	-	W	-	-	switch	Low
	318	KNX: Channel 1	Room temperature sensor			2 bytes	C	-	W	T	U	temperature (°C)	Low
	319	KNX: Channel 1	Current temperature setpoint, status			2 bytes	C	-	W	-	U	temperature (°C)	Low
	320	KNX: Channel 1	Control mode, status			1 byte	C	-	W	-	-	HVAC control mode	Low
	321	KNX: Channel 1	Fan speed, status			1 byte	C	-	W	-	-	percentage (0..100%)	Low
	322	KNX: Channel 1	Vanes swing (1-swing,0-stop), status			1 bit	C	-	W	-	-	start/stop	Low

Air conditioner(with swing)

Fig.6.2.2 Air condition function communication object

NO.	Object Function	Name	Data Type	Flag	DPT
311	**Power on/off**	**KNX: {{Channel 1}}**	**1bit**	**C,T**	**1.001 switch**
The communication object is used to control air condition power on/off via the APP, and send the telegram value to the bus. Telegrams: 1——On 0——Off					
312	**Current setpoint adjustment**	**KNX: {{Channel 1}}**	**2byte**	**C,T**	**9.001 temperature**
The communication object is used to adjust setpoint temperature via the APP, and send telegram value to the bus.					
313	**Control mode**	**KNX: {{Channel 1}}**	**1byte**	**C,T**	**20.105 HVAC control mode**
The communication object is used to send control telegram of each air condition mode to the bus. Different telegram means different control mode: 0-Auto, 1- Heating, 3-Cooling, 9-Fan, 14-Dehumidity, other reserved.					
314	**Fan speed**	**KNX: {{Channel 1}}**	**1byte**	**C,T**	**5.001 percentage** **5.100 fan stage**
The communication object is used to send control telegram of each fan speed to the bus. Telegram value is determined by parameter setting datatype.					
315	**Vanes swing (1-swing,0-stop)**	**KNX: {{Channel 1}}**	**1bit**	**C,T**	**1.010 start/stop**
The communication object is visible only when “Air conditioner(with swing)” is selected. Used to send telegram controlling vanes swing to the bus. Telegrams: 1——Swing 0——Stop					
317	**Power on/off, status**	**KNX: {{Channel 1}}**	**1bit**	**C,W**	**1.001 switch**
The communication object is used to receive the power on/off telegram of air condition from the bus, and feedback to the APP display. Telegrams: 1——On 0——Off					
318	**Room temperature sensor**	**KNX: {{Channel 1}}**	**2byte**	**C,W,T,U**	**9.001 temperature**
The communication object is used to receive the room temperature from the bus, and send read request cyclically, and feedback to the APP display.					

319	**Current temperature setpoint, status**	**KNX: {{Channel 1}}**	**2byte**	**C,W,U**	**9.001 temperature**
The communication object is used to receive the current setpoint temperature from the bus, and feedback to the APP display.					
320	**Control mode, status**	**KNX: {{Channel 1}}**	**1byte**	**C,W**	**20.105 HVAC control mode**
The communication object is used to receive the current control mode from the bus, and feedback to the APP display. Different telegram means different control mode: 0-Auto, 1- Heating, 3-Cooling, 9-Fan, 14-Dehumidity, other reserved.					
321	**Fan speed, status**	**KNX: {{Channel 1}}**	**1byte**	**C,W**	**5.001 percentage** **5.100 fan stage**
The communication object is used to receive the current fan speed from the bus, and feedback to the APP display. Telegram value is determined by parameter setting datatype.					
322	**Vanes swing (1-swing,0-stop), status**	**KNX: {{Channel 1}}**	**1bit**	**C,W**	**1.010 start/stop**
The communication object is visible only when “Air conditioner(with swing)” is selected. Used to receive vanes swing status from the bus. Telegrams: 1——Swing 0——Stop					

Table 6.2.2 Air condition function communication object table

6.2.3. Communication Object of Room temperature unit

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
311	KNX: Channel 1	Power on/off			1 bit	C	-	-	T	-	switch	Low
312	KNX: Channel 1	Current setpoint adjustment			2 bytes	C	-	-	T	-	temperature (°C)	Low
313	KNX: Channel 1	Heating/Cooling mode			1 bit	C	-	-	T	-	cooling/heating	Low
325	KNX: Channel 1	Power on/off, status			1 bit	C	-	W	-	-	switch	Low
326	KNX: Channel 1	Room temperature sensor			2 bytes	C	-	W	T	U	temperature (°C)	Low
327	KNX: Channel 1	Current temperature setpoint, status			2 bytes	C	-	W	-	U	temperature (°C)	Low
328	KNX: Channel 1	Heating/Cooling mode, status			1 bit	C	-	W	-	-	cooling/heating	Low

Room temperature unit

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
311	KNX: Channel 1	Power on/off			1 bit	C	-	-	T	-	switch	Low
312	KNX: Channel 1	Current setpoint adjustment			2 bytes	C	-	-	T	-	temperature (°C)	Low
313	KNX: Channel 1	Heating/Cooling mode			1 bit	C	-	-	T	-	cooling/heating	Low
314	KNX: Channel 1	Operation mode			1 byte	C	-	-	T	-	HVAC mode	Low
325	KNX: Channel 1	Power on/off, status			1 bit	C	-	W	-	-	switch	Low
326	KNX: Channel 1	Room temperature sensor			2 bytes	C	-	W	T	U	temperature (°C)	Low
327	KNX: Channel 1	Current temperature setpoint, status			2 bytes	C	-	W	-	U	temperature (°C)	Low
328	KNX: Channel 1	Heating/Cooling mode, status			1 bit	C	-	W	-	-	cooling/heating	Low
329	KNX: Channel 1	Operation mode, status			1 byte	C	-	W	-	-	HVAC mode	Low

Room temperature unit(with operation mode)

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
311	KNX: Channel 1	Power on/off			1 bit	C	-	-	T	-	switch	Low
312	KNX: Channel 1	Current setpoint adjustment			2 bytes	C	-	-	T	-	temperature (°C)	Low
313	KNX: Channel 1	Heating/Cooling mode			1 bit	C	-	-	T	-	cooling/heating	Low
314	KNX: Channel 1	Operation mode			1 byte	C	-	-	T	-	HVAC mode	Low
319	KNX: Channel 1	Fan speed			1 byte	C	-	-	T	-	percentage (0..100%)	Low
320	KNX: Channel 1	Fan speed low			1 bit	C	-	-	T	-	switch	Low
321	KNX: Channel 1	Fan speed medium			1 bit	C	-	-	T	-	switch	Low
322	KNX: Channel 1	Fan speed high			1 bit	C	-	-	T	-	switch	Low
324	KNX: Channel 1	Fan automatic operation			1 bit	C	-	-	T	-	enable	Low
325	KNX: Channel 1	Power on/off, status			1 bit	C	-	W	-	-	switch	Low
326	KNX: Channel 1	Room temperature sensor			2 bytes	C	-	W	T	U	temperature (°C)	Low
327	KNX: Channel 1	Current temperature setpoint, status			2 bytes	C	-	W	-	U	temperature (°C)	Low
328	KNX: Channel 1	Heating/Cooling mode, status			1 bit	C	-	W	-	-	cooling/heating	Low
329	KNX: Channel 1	Operation mode, status			1 byte	C	-	W	-	-	HVAC mode	Low
334	KNX: Channel 1	Fan speed, status			1 byte	C	-	W	-	-	percentage (0..100%)	Low
335	KNX: Channel 1	Fan speed low, status			1 bit	C	-	W	-	-	switch	Low
336	KNX: Channel 1	Fan speed medium, status			1 bit	C	-	W	-	-	switch	Low
337	KNX: Channel 1	Fan speed high, status			1 bit	C	-	W	-	-	switch	Low
339	KNX: Channel 1	Fan automatic operation, status			1 bit	C	-	W	-	-	enable	Low

Room temperature unit(with operation mode & fan speed)

Fig.6.2.3 Room temperature unit function communication Object

NO.	Object Function	Name	Data Type	Flag	DPT
311	**Power on/off**	**KNX: {{Channel 1}}**	**1bit**	**C,T**	**1.001 switch**
The communication object is used to control RTC power on/off via the APP, and send the telegram value to the bus. Telegrams: 1——On 0——Off					
312	**Current setpoint adjustment**	**KNX: {{Channel 1}}**	**2byte**	**C,T**	**9.001 temperature**
The communication object is used to adjust setpoint temperature via the APP, and send telegram value to the bus.					
313	**Heating/Cooling mode**	**KNX: {{Channel 1}}**	**1bit**	**C,T**	**1.100 cooling/heating**
The communication object is used to send telegram for switching cooling and heating functions to the bus. Telegrams: 1——Heating 0——Cooling					
314	**Operation mode**	**KNX: {{Channel 1}}**	**1byte**	**C,T**	**20.102 HVAC mode**
The communication object is used to send the telegram of the room operation mode to the bus. Different telegram means different control mode: 1-Comfort, 2-Standby, 3-Economy, 4-Protection, other reserved.					

319	**Fan speed**	**KNX: {{Channel 1}}**	**1byte**	**C,T**	**5.001 percentage** **5.100 fan stage**
320	**Fan speed low**	**KNX: {{Channel 1}}**	**1bit**	**C,T**	**1.001 switch**
321	**Fan speed medium**	**KNX: {{Channel 1}}**	**1bit**	**C,T**	**1.001 switch**
322	**Fan speed high**	**KNX: {{Channel 1}}**	**1bit**	**C,T**	**1.001 switch**
323	**Fan speed off**	**KNX: {{Channel 1}}**	**1bit**	**C,T**	**1.001 switch**

These communication objects are used to control fan speed via the APP, and send telegrams to the bus.

1bit object is visible according to the parameter setting :

Object 320——Low fan speed

Object 321——Medium fan speed

Object 322——High fan speed

Object 323——Fan speed off

Only the corresponding object sends telegram “1” when switch to a certain fan speed. When 1bit-off object is not enable, all objects send telegrams “0” when switch to fan speed off (The situation apply to connect with fan actuator of GVS);

When 1bit-off object is enable, only 1bit-off object send telegram “1” (The situation apply to connect with fan actuator of other manufacturers).

1byte: the corresponding telegram value of each fan speed is defined by the parameter. Activate the corresponding fan speed on the APP, and object 319 sends the corresponding telegram value of the fan speed to the bus.

324	**Fan automatic operation**	**KNX: {{Channel 1}}**	**1bit**	**C,T**	**1.003 enable**

The communication object is used to send the telegram of fan automatic control to the bus. Telegrams:

1——Auto

0——Exit auto

325	**Power on/off, status**	**KNX: {{Channel 1}}**	**1bit**	**C,W**	**1.001 switch**

The communication object is used to receive the power on/off telegram of RTC from the bus, and feedback to the APP display. Telegrams:

1——On

0——Off

326	**Room temperature sensor**	**KNX: {{Channel 1}}**	**2byte**	**C,W,T,U**	**9.001 temperature**
The communication object is used to receive the room temperature from the bus, and send read request cyclically, and feedback to the APP display.					
327	**Current temperature setpoint, status**	**KNX: {{Channel 1}}**	**2byte**	**C,W,U**	**9.001 temperature**
The communication object is used to receive the current setpoint temperature from the bus, and feedback to the APP display.					
328	**Heating/Cooling mode, status**	**KNX: {{Channel 1}}**	**1bit**	**C,W**	**1.100 cooling/heating**
The communication object is used for receiving the status of heating and cooling from the bus, and feedback to the APP display. Telegrams: 1 ——Heating 0 ——Cooling					
329	**Operation mode, status**	**KNX: {{Channel 1}}**	**1byte**	**C,W**	**20.102 HVAC mode**
The communication object is used for receiving the telegram of RTC operation mode from the bus. Different telegram means different control mode: 1-Comfort, 2-Standby, 3-Economy, 4-Protection, other reserved.					
334	**Fan speed, status**	**KNX: {{Channel 1}}**	**1byte**	**C,W**	**5.001 percentage** **5.100 fan stage**
335	**Fan speed low, status**	**KNX: {{Channel 1}}**	**1bit**	**C,W**	**1.001 switch**
336	**Fan speed medium, status**	**KNX: {{Channel 1}}**	**1bit**	**C,W**	**1.001 switch**
337	**Fan speed high, status**	**KNX: {{Channel 1}}**	**1bit**	**C,W**	**1.001 switch**
338	**Fan speed off, status**	**KNX: {{Channel 1}}**	**1bit**	**C,W**	**1.001 switch**
These communication objects are used to receive status feedback to control fan speed via four 1bit objects or a 1byte object. 1bit : Object 335——Low fan speed Object 336——Medium fan speed Object 337——High fan speed Object 338——Fan speed off The object receives telegram “1”, activate corresponding fan speed, and and feedback to the APP display. When 1bit-off object is not enable, fan speed off when telegram values of low/medium/high fan speed are “0”;					

When 1bit-off object is enable, fan speed off when 1bit-off object receives telegram “1”, and fan speed also off when telegram values of low/medium/high fan speed are “0” 1byte: the corresponding telegram value of each fan speed is defined by the parameter. When object 334 receives the corresponding value, update the fan speed status on the APP display.					
339	**Fan automatic operation, status**	**KNX: {{Channel 1}}**	**1bit**	**C,W**	**1.003 enable**
The communication object is used to receive feedback status of fan automatic control. Telegrams: 1——Automatic 0——Cancel automatic					

Table 6.2.3 Room temperature unit function communication object table

6.2.4. Communication Object of Ventilation system

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
311	KNX: Channel 1	Power on/off			1 bit	C	-	-	T	-	switch	Low
312	KNX: Channel 1	Fan speed			1 byte	C	-	-	T	-	percentage (0..100%)	Low
313	KNX: Channel 1	Fan speed low			1 bit	C	-	-	T	-	switch	Low
314	KNX: Channel 1	Fan speed medium			1 bit	C	-	-	T	-	switch	Low
315	KNX: Channel 1	Fan speed high			1 bit	C	-	-	T	-	switch	Low
316	KNX: Channel 1	Fan automatic operation			1 bit	C	-	-	T	-	enable	Low
317	KNX: Channel 1	Heat recovery			1 bit	C	-	-	T	-	enable	Low
318	KNX: Channel 1	Power on/off, status			1 bit	C	-	W	-	-	switch	Low
319	KNX: Channel 1	Fan speed, status			1 byte	C	-	W	-	-	percentage (0..100%)	Low
320	KNX: Channel 1	Fan speed low, status			1 bit	C	-	W	-	-	switch	Low
321	KNX: Channel 1	Fan speed medium, status			1 bit	C	-	W	-	-	switch	Low
322	KNX: Channel 1	Fan speed high, status			1 bit	C	-	W	-	-	switch	Low
323	KNX: Channel 1	Fan automatic operation, status			1 bit	C	-	W	-	-	enable	Low
324	KNX: Channel 1	Heat recovery, status			1 bit	C	-	W	-	-	enable	Low
325	KNX: Channel 1	Filter alarm, status			1 bit	C	-	W	-	-	alarm	Low

Ventilation system

Fig.6.2.4 Ventilation system function communication object

NO.	Object Function	Name	Data Type	Flag	DPT
311	**Power on/off**	**KNX: {{Channel 1}}**	**1bit**	**C,T**	**1.001 switch**
The communication object is used to control ventilation power on/off via the APP, and send the telegram value to the bus. Telegrams: 1——On 0——Off					

312	**Fan speed**	**KNX: {{Channel 1}}**	**1byte**	**C,T**	**5.001 percentage** **5.100 fan stage**
313	**Fan speed low**	**KNX: {{Channel 1}}**	**1bit**	**C,T**	**1.001 switch**
314	**Fan speed medium**	**KNX: {{Channel 1}}**	**1bit**	**C,T**	**1.001 switch**
315	**Fan speed high**	**KNX: {{Channel 1}}**	**1bit**	**C,T**	**1.001 switch**

These communication objects are used to send the control telegrams of fan speed to the bus.

1byte: the corresponding telegram value of each fan speed is defined by the parameter. Activate the corresponding fan speed on the APP, and object 312 sends the corresponding telegram value of the fan speed to the bus.

1bit:

Object 313——Low fan speed

Object 314——Medium fan speed

Object 315——High fan speed

The corresponding object sends telegram “1” to activate corresponding fan speed on the APP. Turn off fan speed when send telegram “0”.

316	**Fan automatic operation**	**KNX: {{Channel 1}}**	**1bit**	**C,T**	**1.003 enable**

The communication object is used to send the telegram of fan automatic control to the bus. Telegrams:

1——Auto

0——Exit auto

317	**Heat recovery**	**KNX: {{Channel 1}}**	**1bit**	**C,T**	**1.003 enable**

The communication object is used to control heat recovery mode via the APP, and send telegram to the bus. Telegrams:

1——Active

0——Inactive

318	**Power on/off, status**	**KNX: {{Channel 1}}**	**1bit**	**C,W**	**1.001 switch**

The communication object is used to receive the power on/off telegram of ventilation from the bus, and feedback to the APP display. Telegrams:

1——On

0——Off

319	**Fan speed, status**	**KNX: {{Channel 1}}**	**1byte**	**C,W**	**5.001 percentage** **5.100 fan stage**
320	**Fan speed low, status**	**KNX: {{Channel 1}}**	**1bit**	**C,W**	**1.001 switch**
321	**Fan speed medium, status**	**KNX: {{Channel 1}}**	**1bit**	**C,W**	**1.001 switch**
322	**Fan speed high, status**	**KNX: {{Channel 1}}**	**1bit**	**C,W**	**1.001 switch**
These communication objects are used to receive status feedback to control fan speed via three 1bit objects or a 1byte object. 1byte: the corresponding telegram value of each fan speed is defined by the parameter. When object 319 receives the corresponding value, update the fan speed status on the APP display. 1bit : Object 320——Low fan speed Object 321——Medium fan speed Object 322——High fan speed The object receives telegram "1", activate corresponding fan speed, and and feedback to the APP display. All telegram values of fan speed should be 0 when turn off fan speed;					
323	**Fan automatic operation, status**	**KNX: {{Channel 1}}**	**1bit**	**C,W**	**1.003 enable**
The communication object is used to receive feedback status of fan automatic control. Telegrams: 1——Automatic 0——Cancel automatic					
324	**Heat recovery, status**	**KNX: {{Channel 1}}**	**1bit**	**C,W**	**1.003 enable**
The communication object is used to receive status of heat recovery mode, and feedback to the APP display. Telegrams: 1——Active 0——Inactive					
325	**Filter alarm, status**	**KNX: {{Channel 1}}**	**1bit**	**C,W**	**1.005 alarm**
The communication object is used to receive filter status from the bus, and feedback to the APP display. Telegrams: 1——In normal use 0——Replace alarm					

Table 6.2.4 Ventilation system function communication object table

6.2.5. 6.6.5.Communication Object of Audio control

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
312	KNX: Channel 1	Play=1/Pause=0			1 bit	C	-	-	T	-	start/stop	Low
313	KNX: Channel 1	Next track=1/Previous track=0			1 bit	C	-	-	T	-	step	Low
314	KNX: Channel 1	Volume+=1/Volume-=0			1 bit	C	-	-	T	-	step	Low
315	KNX: Channel 1	Absolute volume			1 byte	C	-	-	T	-	percentage (0..100%)	Low
316	KNX: Channel 1	Mute			1 bit	C	-	-	T	-	enable	Low
317	KNX: Channel 1	Play mode			1 byte	C	-	-	T	-	counter pulses (0..255)	Low
319	KNX: Channel 1	Play=1/Pause=0, status			1 bit	C	-	W	T	U	start/stop	Low
320	KNX: Channel 1	Absolute volume, status			1 byte	C	-	W	T	U	percentage (0..100%)	Low
321	KNX: Channel 1	Mute, status			1 bit	C	-	W	T	U	enable	Low
322	KNX: Channel 1	Play mode, status			1 byte	C	-	W	T	U	counter pulses (0..255)	Low

Audio control

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
311	KNX: Channel 1	Power on/off			1 bit	C	-	-	T	-	switch	Low
312	KNX: Channel 1	Play=1/Pause=0			1 bit	C	-	-	T	-	start/stop	Low
313	KNX: Channel 1	Next track=1/Previous track=0			1 bit	C	-	-	T	-	step	Low
314	KNX: Channel 1	Volume+=1/Volume-=0			1 bit	C	-	-	T	-	step	Low
317	KNX: Channel 1	Play mode			1 byte	C	-	-	T	-	counter pulses (0..255)	Low
318	KNX: Channel 1	Power on/off, status			1 bit	C	-	W	-	-	switch	Low
319	KNX: Channel 1	Play=1/Pause=0, status			1 bit	C	-	W	-	-	start/stop	Low
322	KNX: Channel 1	Play mode, status			1 byte	C	-	W	-	-	counter pulses (0..255)	Low

Audio control(with on/off)

Fig.6.2.5 Audio control function communication object

Note: If "Audio control(with on/off)" is selected, default as power off when voltage recovery, and the object no need to send read request telegram. If "Audio control" is selected, the object needs to send read request telegram when voltage recovery.

NO.	Object Function	Name	Data Type	Flag	DPT
311	**Power on/off**	**KNX: {{Channel 1}}**	**1bit**	**C,T**	**1.001 switch**
The communication object only applies to the audio control with power on/off. Used to to control the audio power on/off via the APP, and send the telegram to the bus. Telegrams: 1——On 0——Off					
312	**Play=1/Pause=0**	**KNX: {{Channel 1}}**	**1bit**	**C,T**	**1.010 start/stop**
The communication object is used to play/stop the music in the audio module via the APP. Telegrams: 1——Play music 0——Pause playing music					
313	**Next track=1/Previous track=0**	**KNX: {{Channel 1}}**	**1bit**	**C,T**	**1.007 step**
The communication object is used to switch the playing song of the audio module via the APP, to switch the previous song/the next song. Telegrams: 1——Play the next song 0——Play the previous song					

314	**Volume+=1/Volume-=0**	**KNX: {{Channel 1}}**	**1bit**	**C,T**	**1.007 step**
315	**Absolute volume**	**KNX: {{Channel 1}}**	**1byte**	**C,T**	**5.001 percentage** **5.004 percentage**
The communication object is used to adjust volume of the audio module via the APP. Telegram value is determined by different object datatype. 1 bit object, telegrams: 1——Increase volume 0——Decrease volume 1 byte object only applies to the audio control function without power on/off. Telegram value is according to object type: 0..100 / 0..255					
316	**Mute**	**KNX: {{Channel 1}}**	**1bit**	**C,T**	**1.003 enable**
The communication object only applies to the audio control function without power on/off, used to control mute of the audio module via the APP. Telegrams: 1——Mute 0——Cancel mute					
317	**Play mode**	**KNX: {{Channel 1}}**	**1byte**	**C,T**	**5.010 counter pulses**
The communication object is used to send control telegram of the audio module play mode, different mode telegrams are preset by parameters.					
318	**Power on/off, status**	**KNX: {{Channel 1}}**	**1bit**	**C,W**	**1.001 switch**
The communication object only applies to the audio control function with power on/off. Used to receive the power on/off telegram of the audio module from the bus.					
319	**Play=1/Pause=0, status**	**KNX: {{Channel 1}}**	**1bit**	**C,W** **C,W,T,U**	**1.010 start/stop**
The communication object is used to receive the music play/stop status from the bus. When the audio control function with power on/off, the object flag is C,W; And without power on/off, the object flag is C,W,T,U.					
320	**Absolute volume, status**	**KNX: {{Channel 1}}**	**1byte**	**C,W,T,U**	**5.001 percentage** **5.004 percentage**
Volume adjustment function, 1byte object only applies to the audio control function without power on/off. Used to receive the volume status of audio control. Telegram value is according to object type: 0..100 / 0..255					

321	**Mute, status**	**KNX: {{Channel 1}}**	**1bit**	**C,W,T,U**	**1.003 enable**
The communication object only applies to the audio control function without power on/off. Used to receive the mute status of audio control.					
322	**Play mode, status**	**KNX: {{Channel 1}}**	**1byte**	**C,W** **C,W,T,U**	**5.010 counter pulses**
The communication object is used to receive the play mode status of the audio module from the bus, different mode telegrams are preset by parameters. When the audio control function with power on/off, the object flag is C,W; And without power on/off, the object flag is C,W,T,U.					

Table 6.2.5 Audio control function communication object table

6.2.6. Communication Object of sensor function

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
311	KNX: Channel 1	PM2.5 value			2 bytes	C	-	W	T	U	pulses	Low
312	KNX: Channel 1	PM10 value			2 bytes	C	-	W	T	U	pulses	Low
313	KNX: Channel 1	VOC value			2 bytes	C	-	W	T	U	pulses	Low
314	KNX: Channel 1	CO2 value			2 bytes	C	-	W	T	U	pulses	Low
311	KNX: Channel 1	PM2.5 value			2 bytes	C	-	W	T	U	concentration (µg/m³)	Low
312	KNX: Channel 1	PM10 value			2 bytes	C	-	W	T	U	concentration (µg/m³)	Low
313	KNX: Channel 1	VOC value			2 bytes	C	-	W	T	U	concentration (µg/m³)	Low
314	KNX: Channel 1	CO2 value			2 bytes	C	-	W	T	U	parts/million (ppm)	Low
315	KNX: Channel 1	AQI value			2 bytes	C	-	W	T	U	pulses	Low
316	KNX: Channel 1	Temperature value			2 bytes	C	-	W	T	U	temperature (°C)	Low
317	KNX: Channel 1	Humidity value			2 bytes	C	-	W	T	U	humidity (%)	Low

Air quality sensor

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
314	KNX: Channel 1	CO2 value			2 bytes	C	-	W	T	U	pulses	Low
314	KNX: Channel 1	CO2 value			2 bytes	C	-	W	T	U	parts/million (ppm)	Low
316	KNX: Channel 1	Temperature value			2 bytes	C	-	W	T	U	temperature (°C)	Low
317	KNX: Channel 1	Humidity value			2 bytes	C	-	W	T	U	humidity (%)	Low

CO2 sensor

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
311	KNX: Channel 1	PM2.5 value			2 bytes	C	-	W	T	U	pulses	Low
311	KNX: Channel 1	PM2.5 value			2 bytes	C	-	W	T	U	concentration (µg/m³)	Low
316	KNX: Channel 1	Temperature value			2 bytes	C	-	W	T	U	temperature (°C)	Low
317	KNX: Channel 1	Humidity value			2 bytes	C	-	W	T	U	humidity (%)	Low

PM2.5 sensor

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
313	KNX: Channel 1	VOC value			2 bytes	C	-	W	T	U	pulses	Low
313	KNX: Channel 1	VOC value			2 bytes	C	-	W	T	U	concentration (µg/m³)	Low
316	KNX: Channel 1	Temperature value			2 bytes	C	-	W	T	U	temperature (°C)	Low
317	KNX: Channel 1	Humidity value			2 bytes	C	-	W	T	U	humidity (%)	Low

VOC sensor

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
311	KNX: Channel 1	Presence detector			1 bit	C	-	W	T	U	occupancy	Low
312	KNX: Channel 1	Brightness value(lux)			2 bytes	C	-	W	T	U	lux (Lux)	Low
312	KNX: Channel 1	Brightness value(lux)			2 bytes	C	-	W	T	U	brightness (lux)	Low

Presence sensor

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
311	KNX: Channel 1	Motion detector			1 bit	C	-	W	T	U	occupancy	Low

Motion sensor

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
312	KNX: Channel 1	Brightness value(lux)			2 bytes	C	-	W	T	U	lux (Lux)	Low
312	KNX: Channel 1	Brightness value(lux)			2 bytes	C	-	W	T	U	brightness (lux)	Low

Brightness sensor

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
311	KNX: Channel 1	I/O signal value			1 bit	C	-	W	T	U	switch	Low

I/O signal

Fig.6.2.6 Sensor function communication object

NO.	Object Function	Name	Data Type	Flag	DPT
311	**PM2.5 value**	**KNX: {{Channel 1}}**	**2byte**	**C,W,T,U**	**7.001 pulse** **9.030 concentration(ug/m3)**
The communication object is used to receive the input of the PM2.5 value and get the corresponding value from the bus to be updated to the APP display in ug/m^3. Range: 0~999ug/m^3, object datatype is determined by the parameter setting.					
312	**PM10 value**	**KNX: {{Channel 1}}**	**2byte**	**C,W,T,U**	**7.001 pulse** **9.030 concentration(ug/m3)**
The communication object is used to receive the input of the PM10 value and get the corresponding value from the bus to be updated to the APP display in ug/m^3. Range: 0~999ug/m^3, object datatype is determined by the parameter setting.					
313	**VOC value**	**KNX: {{Channel 1}}**	**2byte**	**C,W,T,U**	**7.001 pulse** **9.030 concentration(ug/m3)**
The communication object is used to receive the input of the VOC value and get the corresponding value from the bus to be updated to the APP display in ug/m^3. Range: 0~999ug/m^3, object datatype is determined by the parameter setting. When the object datatype is selected 7.001pulses, thousand ratio reduction based on the datapoint type 7.001 pulses, for example, received value is 5000(ug/m^3), actually it displays 5.000mg/m^3					
314	**CO2 value**	**KNX: {{Channel 1}}**	**2byte**	**C,W,T,U**	**7.001 pulse** **9.008 parts/million(ppm)**
The communication object is used to receive the input of the CO2 value and get the corresponding value from the bus to be updated to the APP display in ppm. Range: 0...4000ppm					
316	**AQI value**	**KNX: {{Channel 1}}**	**2byte**	**C,W,T,U**	**7.001 pulses**
The communication object is used to receive the input of the AQI value and get the corresponding value from the bus to be updated to the APP display. Range: 0...500					

317	**Temperature value**	**KNX: {{Channel 1}}**	**2byte**	**C,W,T,U**	**9.001 temperature**
The communication object is used to receive the input of the temperature value and get the corresponding value from the temperature sensor on bus to be updated to the APP display. Range: -40...40℃					
311	**Humidity value**	**KNX: {{Channel 1}}**	**2byte**	**C,W,T,U**	**9.007 humidity**
The communication object is used to receive the input of the humidity value and get the corresponding value from the humidity sensor on bus to be updated to the APP display. Range: 0...100%					
312	**Brightness value(lux)**	**KNX: {{Channel 1}}**	**2byte**	**C,W,T,U**	**7.013 brightness(lux)** **9.004 lux(lux)**
The communication object is used to receive the input of the brightness value and get the corresponding value from the bus to be updated to the APP display in lux. Range: 0...65535lux, object datatype is determined by the parameter setting.					
311	**Motion detector**	**KNX: {{Channel 1}}**	**1bit**	**C,W,T,U**	**1.018 Occupancy**
The communication object is used to receive the input of the 1bit presence signal. Telegram: 0——No one 1——Some one					
311	**I/O signal value**	**KNX: {{Channel 1}}**	**1bit**	**C,W,T,U**	**1.001 switch**
The communication object is used to receive the input of the 1bit I/O signal. Telegram: 0——Open 1——Close					

Table 6.2.6 Sensor function communication object table

6.2.7. Communication Object of current/energy metering

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
311	KNX: Channel 1	Current value			2 bytes	C	-	W	T	U	current (mA)	Low
313	KNX: Channel 1	Power value			2 bytes	C	-	W	T	U	power (kW)	Low
311	KNX: Channel 1	Current value			4 bytes	C	-	W	T	U	electric current (A)	Low
313	KNX: Channel 1	Power value			4 bytes	C	-	W	T	U	power (W)	Low

Current metering

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
311	KNX: Channel 1	Current value			2 bytes	C	-	W	T	U	current (mA)	Low
312	KNX: Channel 1	Voltage value			2 bytes	C	-	W	T	U	voltage (mV)	Low
313	KNX: Channel 1	Power value			2 bytes	C	-	W	T	U	power (kW)	Low
314	KNX: Channel 1	Energy value			4 bytes	C	-	W	T	U	active energy (Wh)	Low
311	KNX: Channel 1	Current value			4 bytes	C	-	W	T	U	electric current (A)	Low
312	KNX: Channel 1	Voltage value			4 bytes	C	-	W	T	U	electric potential (V)	Low
313	KNX: Channel 1	Power value			4 bytes	C	-	W	T	U	power (W)	Low
314	KNX: Channel 1	Energy value			4 bytes	C	-	W	T	U	active energy (kWh)	Low

Energy metering

Fig.6.2.7 Current/Energy metering function communication object

NO.	Object Function	Name	Data Type	Flag	DPT
311	**Current value**	**KNX: {{Channel 1}}**	**2byte** **4byte**	**C,W,T,U**	**7.012 current(mA)** **9.021 current(mA)** **14.019 electric current(A)**
The communication object is used to receive the current value from the bus, updated to the APP display. Object datatype is determined by the parameter setting. When 2byte int is selected, display range: 0...65535mA, step value is 1mA. When 2byte float is selected, display range: -670760...670760mA, step value is 0.01mA. When 4byte is selected, display range: -99999999.9...99999999.9A, step value is 0.1A.					
312	**Voltage value**	**KNX: {{Channel 1}}**	**2byte** **4byte**	**C,W,T,U**	**9.020 voltage(mV)** **14.027 electric potential(V)**
The communication object is used to receive the voltage value from the bus, updated to the APP display. Object datatype is determined by the parameter setting. When 2byte is selected, display range: -670760mV...670760mV, step value is 0.01mV. When 4byte is selected, display range: -99999999.9...99999999.9V, step value is 0.1V.					
313	**Power value**	**KNX: {{Channel 1}}**	**2byte** **4byte**	**C,W,T,U**	**9.024 power(kW)** **14.056 power(W)**
The communication object is used to receive the power value from the bus, updated to the APP display. Object datatype is determined by the parameter setting. When 2byte is selected, display range: -670760kW...670760kW, step value is 0.01kW. When 4byte is selected, display range: -99999999.9...99999999.9W, step value is 0.1W.					
314	**Energy value**	**KNX: {{Channel 1}}**	**4byte**	**C,W,T,U**	**13.010 active energy(Wh)** **13.013 active energy(kWh)**
The communication object is used to receive the energy value from the bus, updated to the APP display. Object datatype is determined by the parameter setting. Display range: -2147483648...2147483647Wh, step value is 1Wh. Display range: -2147483648...2147483647kWh, step value is 1kWh.					

Table 6.2.7 Current/Energy metering function communication object table

6.3. “Room temperature controller” Communication Object

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
74	RTC 1	Power on/off			1 bit	C	-	W	-	-	switch	Low
75	RTC 1	Room temperature sensor			2 bytes	C	-	W	T	U	temperature (°C)	Low
76	RTC 1	Current setpoint adjustment			2 bytes	C	-	W	-	-	temperature (°C)	Low
77	RTC 1	Heating/Cooling mode			1 bit	C	-	W	-	-	cooling/heating	Low
78	RTC 1	Operation mode			1 byte	C	-	W	-	-	HVAC mode	Low
79	RTC 1	Comfort mode			1 bit	C	-	W	-	-	enable	Low
80	RTC 1	Economy mode			1 bit	C	-	W	-	-	enable	Low
81	RTC 1	Frost/Heat protection mode			1 bit	C	-	W	-	-	enable	Low
82	RTC 1	Standby mode			1 bit	C	-	W	-	-	enable	Low
83	RTC 1	Fan automatic operation			1 bit	C	-	W	-	-	enable	Low
84	RTC 1	Window contact			1 bit	C	-	W	-	U	window/door	Low
85	RTC 1	Presence detector			1 bit	C	-	W	-	U	occupancy	Low
86	RTC 1	Current temperature setpoint, status			2 bytes	C	R	-	T	-	temperature (°C)	Low
87	RTC 1	Heating/Cooling mode, status			1 bit	C	R	-	T	-	cooling/heating	Low
88	RTC 1	Operation mode, status			1 byte	C	R	-	T	-	HVAC mode	Low
89	RTC 1	Comfort mode, status			1 bit	C	R	-	T	-	enable	Low
90	RTC 1	Economy mode, status			1 bit	C	R	-	T	-	enable	Low
91	RTC 1	Frost/Heat protection mode, status			1 bit	C	R	-	T	-	enable	Low
92	RTC 1	Standby mode, status			1 bit	C	R	-	T	-	enable	Low
93	RTC 1	Heating control value, status			1 bit	C	R	-	T	-	switch	Low
94	RTC 1	Cooling control value, status			1 bit	C	R	-	T	-	switch	Low
95	RTC 1	Fan speed, status			1 byte	C	-	-	T	-	percentage (0..100%)	Low

Fig.6.3 “Room temperature controller” communication object

NO.	Object Function	Name	Data Type	Flag	DPT
74	**Power on/off**	**RTC 1**	**1bit**	**C,W**	**1.001 switch**
The communication object is used to receive the telegram value of control RTC power on/off from the bus. Telegrams: 1——On 0——Off					
75	**Room temperature sensor**	**RTC 1**	**2byte**	**C,W,T,U**	**9.001 temperature**
The communication object is used to receive the room temperature from the bus, and send read request cyclically, and also receive feedback.					
76	**Current setpoint adjustment**	**RTC 1**	**2byte**	**C,W**	**9.001 temperature**
The communication object is used to modify the current setpoint value of temperature. Telegrams: 0..65535					
77	**Heating/Cooling mode**	**RTC 1**	**1bit**	**C,W**	**1.100 cooling/heating**
The communication object is used to switch heating/cooling mode via the bus. Telegrams: 1——Heating 0——Cooling					

78	**Operation mode**	**RTC 1**	**1byte**	**C,W**	**20.102 HVAC mode**
79	**Comfort mode**	**RTC 1**	**1bit**	**C,W**	**1.003 enable**
80	**Economy mode**	**RTC 1**	**1bit**	**C,W**	**1.003 enable**
81	**Frost/Heat protection mode**	**RTC 1**	**1bit**	**C,W**	**1.003 enable**
82	**Standby mode**	**RTC 1**	**1bit**	**C,W**	**1.003 enable**

These communication objects are used to control RTC operation mode via the bus.

1byte: object 78 is visible. Telegrams: 1-comfort, 2-standby, 3-economy, 4-protection, other reserved.

1bit:

Object 79——Comfort mode

Object 80——Economy mode

Object 81——Protection mode

Object 82——Standby mode

Activate corresponding mode when receive a telegram “1”. When 1bit standby mode is not enable, and objects value of comfort, economy and protection are all 0, it is standby mode. When 1bit standby mode is enable, the 1bit standby object receives 1 is standby mode, and telegram “0” will be ignored.

83	**Fan automatic operation**	**RTC 1**	**1bit**	**C,W**	**1.003 enable**

The communication object is used to activate fan automatic control via the bus. Telegrams:

1——Auto

0——Exit auto

84	**Window contact**	**RTC 1**	**1bit**	**C,W,U**	**1.019 Window/door**

The communication object is used to receive the switch status of window contact. Telegrams:

1——Open window

0——Close window

85	**Presence detector**	**RTC 1**	**1bit**	**C,W,U**	**1.018 occupancy**

The communication object is used to receive the room occupancy status from presence detector. Telegrams:

1——Some one

0——No one

86	**Current temperature setpoint, status**	**RTC 1**	**2byte**	**C,R,T**	**9.001 temperature**

The communication object is used to send the current setpoint temperature of RTC to the bus.

Telegrams: 0..65535

87	**Heating/Cooling mode, status**	**RTC 1**	**1bit**	**C,R,T**	**1.100 cooling/heating**
The communication object is used to feedback the telegram to the bus for switching heating and cooling function.					
88	**Operation mode, status**	**RTC 1**	**1byte**	**C,R,T**	**20.102 HVAC mode**
89	**Comfort mode, status**	**RTC 1**	**1bit**	**C,R,T**	**1.003 enable**
90	**Economy mode, status**	**RTC 1**	**1bit**	**C,R,T**	**1.003 enable**
91	**Frost/Heat protection mode, status**	**RTC 1**	**1bit**	**C,R,T**	**1.003 enable**
92	**Standby mode, status**	**RTC 1**	**1bit**	**C,R,T**	**1.003 enable**
These communication objects are used to send RTC operation mode status to the bus. When 1 byte: object 88 is visible, telegrams: 1-comfort, 2-standby, 3-economy, 4-protection, other reserved. When 1bit: Object 89—— Comfort mode Object 90—— Economy mode Object 91—— Protection mode Object 92—— Standby mode When a mode is activated, the corresponding object only sends telegram “1”. When 1 bit standby object is not enable, activate standby mode when comfort, economy, protection objects send telegram 0 together. When 1 bit standby object is enable, activate standby mode only when standby object send 1. **Note: no requirement to send mode status to the bus when switchover via bus. The same is fan speed and other operation.**					
93	**Heating control value, status**	**RTC 1**	**1bit** **1byte**	**C,R,T**	**1.001 switch** **5.001 percentage**
94	**Cooling control value, status**	**RTC 1**	**1bit** **1byte**	**C,R,T**	**1.001 switch** **5.001 percentage**
These communication objects are used to send control status of heating or cooling function to the bus. Object datatype is according to parameter setting.					
94	**Fan speed, status**	**RTC 1**	**1byte**	**C,T**	**5.001 percentage** **5.100 fan stage**
The communication object is used to send the status of fan speed to the bus. The corresponding telegram value of each fan speed is defined by the parameter.					

Table 6.4 “Room temperature controller” communication object table

6.4. “Ventilation controller” Communication Object

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
294	Ventilation controller	Fan automatic operation			1 bit	C	-	W	-	-	enable	Low
295	Ventilation controller	PM 2.5 value			2 bytes	C	-	W	T	U	pulses	Low
296	Ventilation controller	Fan speed, status			1 byte	C	-	-	T	-	percentage (0..100%)	Low
297	Ventilation controller	Fan speed low, status			1 bit	C	-	-	T	-	switch	Low
298	Ventilation controller	Fan speed medium, status			1 bit	C	-	-	T	-	switch	Low
299	Ventilation controller	Fan speed high, status			1 bit	C	-	-	T	-	switch	Low

Fig.6.4 “Ventilation controller” communication object

NO.	Object Function	Name	Data Type	Flag	DPT
294	**Fan automatic operation**	**Ventilation controller**	**1bit**	**C,W**	**1.003 enable**
The communication object is used to activate fan automatic control via the bus. Telegrams: 1——Auto 0——Exit auto					
295	**PM 2.5 value** **VOC value** **CO2 value**	**Ventilation controller**	**2byte**	**C,W,T,U**	**7.001 pulse** **9.030 concentration(ug/m3)** **9.008 parts/million(ppm)**
The communication object is used to receive the input value of PM2.5/VOC/CO2, and get the corresponding value from the bus and update it to the display, the corresponding object datatype is defined by the parameter. Unit is ug/m^3 or ppm. Range: 0~999ug/m^3 or 0...4000ppm For example, Control value of automatic operation is PM2.5, that is, under the automation, the ventilation system can adjust the fan speed automatically according to PM2.5 concentration.					
296	**Fan speed, status**	**Ventilation controller**	**1byte**	**C,T**	**5.001 percentage** **5.100 fan stage**
297	**Fan speed low, status**	**Ventilation controller**	**1bit**	**C,T**	**1.001 switch**
298	**Fan speed medium, status**	**Ventilation controller**	**1bit**	**C,T**	**1.001 switch**
299	**Fan speed high, status**	**Ventilation controller**	**1bit**	**C,T**	**1.001 switch**
These communication objects are used to send control telegrams of the fan speed to the bus. 1byte: the corresponding telegram value of each fan speed is defined by the parameter. Object 296 sends the corresponding telegram value of the fan speed to the bus. 1bit: Object 297——Low fan speed Object 298——Medium fan speed Object 299——High fan speed The corresponding object sends telegram “1” to the bus, and sends telegram “0” is turn off fan speed.					

Table 6.4 “Ventilation controller” communication object table

6.5. “Logic” Communication Object

6.5.1. “AND/OR/XOR” Communication Object

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
2	1st Logic	Input a			1 bit	C	-	W	T	U	boolean	Low
3	1st Logic	Input b			1 bit	C	-	W	T	U	boolean	Low
4	1st Logic	Input c			1 bit	C	-	W	T	U	boolean	Low
5	1st Logic	Input d			1 bit	C	-	W	T	U	boolean	Low
6	1st Logic	Input e			1 bit	C	-	W	T	U	boolean	Low
7	1st Logic	Input f			1 bit	C	-	W	T	U	boolean	Low
8	1st Logic	Input g			1 bit	C	-	W	T	U	boolean	Low
9	1st Logic	Input h			1 bit	C	-	W	T	U	boolean	Low
10	1st Logic	Logic result			1 bit	C	-	-	T	-	boolean	Low

Fig.6.5.1 “AND/OR/XOR” Communication Object

NO.	Object Function	Name	Data Type	Flag	DPT
2/...	**Input x**	**1st /.../8th Logic**	**1bit**	**C,W,T,U**	**1.002 boolean**
The communication object is used to receive the value of logical input Input x.					
10	**Logic result**	**1st /.../8th Logic**	**1bit**	**C,T**	**1.002 boolean**
The communication object is used to send the results of logical operation.					

Table 6.5.1 “AND/OR/XOR” communication object table

6.5.2. “Gate forwarding” Communication Object

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
2	1st Logic	Gate value select			1 byte	C	-	W	-	-	scene number	Low
3	1st Logic	Input A			1 bit	C	-	W	-	-	switch	Low
4	1st Logic	Input B			1 bit	C	-	W	-	-	switch	Low
5	1st Logic	Input C			1 bit	C	-	W	-	-	switch	Low
6	1st Logic	Input D			1 bit	C	-	W	-	-	switch	Low
7	1st Logic	Output A			1 bit	C	-	-	T	-	switch	Low
8	1st Logic	Output B			1 bit	C	-	-	T	-	switch	Low
9	1st Logic	Output C			1 bit	C	-	-	T	-	switch	Low
10	1st Logic	Output D			1 bit	C	-	-	T	-	switch	Low

Fig.6.5.2 “Gate forwarding” Communication Object

NO.	Object Function	Name	Data Type	Flag	DPT
2	**Gate value select**	**1st /.../8th Logic**	**1byte**	**C,W**	**17.001 scene number**
The communication object is used to select the scene of logical gate forwarding.					
3/.../6	**Input x**	**1st /.../8th Logic**	**1bit** **4bit** **1byte**	**C,W**	**1.001 switch** **3.007 dimming control** **5.010 counter pulses(0..255)**
The communication object is used to receive the value of the logic gate input Input x.					

7/../10	**Output x**	**1st /.../8th Logic**	**1bit** **4bit** **1byte**	**C,T**	**1.001 switch** **3.007 dimming control** **5.010 counter pulses(0..255)**
The communication object is used to output the value forwarded by the logic gate. The output value is the same as the input value, but one input can be forwarded into one or more outputs, set by parameters.					

Table 6.5.2 “Gate forwarding” communication object table

6.5.3. “Threshold comparator” Communication Object

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
2	1st Logic	Threshold value input			4 bit	C	-	W	-	U	dimming control	Low
2	1st Logic	Threshold value input			1 byte	C	-	W	-	U	counter pulses (0..255)	Low
2	1st Logic	Threshold value input			2 bytes	C	-	W	-	U	pulses	Low
2	1st Logic	Threshold value input			4 bytes	C	-	W	-	U	counter pulses (unsigned)	Low
10	1st Logic	Logic result			1 bit	C	-	-	T	-	boolean	Low

Fig.6.5.3 “Threshold comparator” Communication Object

NO.	Object Function	Name	Data Type	Flag	DPT
2	**Threshold value input**	**1st /.../8th Logic**	**4bit** **1byte** **2byte** **4byte**	**C,W, U**	**3.007 dimming control** **5.010 counter pulses(0..255)** **7.001 pulses** **12.001 counter pulses**
The communication object is used to input threshold value.					
10	**Logic result**	**1st /.../8th Logic**	**1bit**	**C,T**	**1.002 boolean**
The communication object is used to send the results of logical operation. That is, the value that should be sent after the object input threshold is compared with the setting threshold value.					

Table 6.5.3 “Threshold comparator” communication object table

6.5.4. “Format convert” Communication Object

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
2	1st Logic	Input 1bit-bit0			1 bit	C	-	W	-	U	boolean	Low
3	1st Logic	Input 1bit-bit1			1 bit	C	-	W	-	U	boolean	Low
10	1st Logic	Output 2bit			2 bit	C	-	-	T	-	switch control	Low

“2x1bit --> 1x2bit”function: converts two 1bit values to a 2bit value, such as Input bit1=1, bit0=0--> Output 2bit=2

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
2	1st Logic	Input 1bit-bit0			1 bit	C	-	W	-	U	boolean	Low
3	1st Logic	Input 1bit-bit1			1 bit	C	-	W	-	U	boolean	Low
4	1st Logic	Input 1bit-bit2			1 bit	C	-	W	-	U	boolean	Low
5	1st Logic	Input 1bit-bit3			1 bit	C	-	W	-	U	boolean	Low
6	1st Logic	Input 1bit-bit4			1 bit	C	-	W	-	U	boolean	Low
7	1st Logic	Input 1bit-bit5			1 bit	C	-	W	-	U	boolean	Low
8	1st Logic	Input 1bit-bit6			1 bit	C	-	W	-	U	boolean	Low
9	1st Logic	Input 1bit-bit7			1 bit	C	-	W	-	U	boolean	Low
10	1st Logic	Output 1byte			1 byte	C	-	-	T	-	counter pulses (0..255)	Low

“8x1bit --> 1x1byte”function: converts eight 1bit values to a 1byte value, such as Input bit2=1, bit1=1, bit0=1,other bits are 0--> Output 1byte=7

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
2	1st Logic	Input 1byte			1 byte	C	-	W	-	U	counter pulses (0..255)	Low
10	1st Logic	Output 2byte			2 bytes	C	-	-	T	-	pulses	Low

“1x1byte --> 1x2byte”function: converts one 1byte values to a 2byte value, such as Input 1byte=125--> Output 2byte=125.Although the value remains the same, the data type of the value is different.

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
2	1st Logic	Input 1byte-low			1 byte	C	-	W	-	U	counter pulses (0..255)	Low
3	1st Logic	Input 1byte-high			1 byte	C	-	W	-	U	counter pulses (0..255)	Low
10	1st Logic	Output 2byte			2 bytes	C	-	-	T	-	pulses	Low

“2x1byte --> 1x2byte”function: converts two 1byte values to a 2byte value, such as Input 1byte-low = 255 ($FF), Input 1byte-high = 100 ($64) --> Output 2byte = 25855 ($64 FF)

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
2	1st Logic	Input 2byte-low			2 bytes	C	-	W	-	U	pulses	Low
3	1st Logic	Input 2byte-high			2 bytes	C	-	W	-	U	pulses	Low
10	1st Logic	Output 4byte			4 bytes	C	-	-	T	-	counter pulses (unsigned)	Low

“2x2byte --> 1x4byte”function: converts two 2 byte values to a 4byte value, such as Input 2byte-low = 65530 ($FF FA), Input 2byte-high = 32768 ($80 00)--> Output 2byte = 2147549178 ($80 00 FF FA)

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
2	1st Logic	Input 1byte			1 byte	C	-	W	-	U	counter pulses (0..255)	Low
3	1st Logic	Output 1bit-bit0			1 bit	C	-	-	T	-	boolean	Low
4	1st Logic	Output 1bit-bit1			1 bit	C	-	-	T	-	boolean	Low
5	1st Logic	Output 1bit-bit2			1 bit	C	-	-	T	-	boolean	Low
6	1st Logic	Output 1bit-bit3			1 bit	C	-	-	T	-	boolean	Low
7	1st Logic	Output 1bit-bit4			1 bit	C	-	-	T	-	boolean	Low
8	1st Logic	Output 1bit-bit5			1 bit	C	-	-	T	-	boolean	Low
9	1st Logic	Output 1bit-bit6			1 bit	C	-	-	T	-	boolean	Low
10	1st Logic	Output 1bit-bit7			1 bit	C	-	-	T	-	boolean	Low

“1x1byte --> 8x1bit” function: converts one 1byte values to eight 1but value, such as Input 1byte=200 --> Output bit0=0, bit1=0, bit2=0, bit3=1, bit4=0, bit5=0, bit6=1, bit7=1

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
2	1st Logic	Input 2byte			2 bytes	C	-	W	-	U	pulses	Low
9	1st Logic	Output 1byte-low			1 byte	C	-	-	T	-	counter pulses (0..255)	Low
10	1st Logic	Output 1byte-high			1 byte	C	-	-	T	-	counter pulses (0..255)	Low

“1x2byte --> 2x1byte”function: converts one 2byte values to two 2byte value, such as Input 2byte = 55500 ($D8 CC) --> Output 1byte-low = 204 ($CC), Output 1byte-high =216 ($D8)

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
2	1st Logic	Input 4byte			4 bytes	C	-	W	-	U	counter pulses (unsigned)	Low
9	1st Logic	Output 2byte-low			2 bytes	C	-	-	T	-	pulses	Low
10	1st Logic	Output 2byte-high			2 bytes	C	-	-	T	-	pulses	Low

"1x4byte --> 2x2byte"function: converts one 4byte values to two 2byte value, such asInput 4byte = 78009500 ($04 A6 54 9C) --> Output 2byte-low = 21660 ($54 9C), Output 2byte-high =1190 ($04 A6)

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
2	1st Logic	Input 3byte			3 bytes	C	-	W	-	U	RGB value 3x(0..255)	Low
8	1st Logic	Output 1byte-low			1 byte	C	-	-	T	-	counter pulses (0..255)	Low
9	1st Logic	Output 1byte-middle			1 byte	C	-	-	T	-	counter pulses (0..255)	Low
10	1st Logic	Output 1byte-high			1 byte	C	-	-	T	-	counter pulses (0..255)	Low

"1x3byte --> 3x1byte"function: converts one 3byte values to three 1byte value, such as Input 3byte = $78 64 C8--> Output 1byte-low = 200 ($C8) , Output 1byte-middle = 100 ($64) , Output 1byte-high =120 ($78)

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
2	1st Logic	Input 1byte-low			1 byte	C	-	W	-	U	counter pulses (0..255)	Low
3	1st Logic	Input 1byte-middle			1 byte	C	-	W	-	U	counter pulses (0..255)	Low
4	1st Logic	Input 1byte-high			1 byte	C	-	W	-	U	counter pulses (0..255)	Low
10	1st Logic	Output 3byte			3 bytes	C	-	-	T	-	RGB value 3x(0..255)	Low

"3x1byte --> 1x3byte"function: converts three 1byte values to a 3byte value, such as Input 1byte-low = 150 ($96), Input 1byte-middle = 100 ($64), Input 1byte-high = 50 ($32)--> Output 3byte = $32 64 96

Fig.6.5.4 "Format convert" Communication Object

NO.	Object Function	Name	Data Type	Flag	DPT
2	**Input ...**	**1st /.../8th Logic**	**1bit** **1byte** **2byte** **3byte** **4byte**	**C,W,U**	**1.001 switch** **5.010 counter pulses(0..255)** **7.001 pulses** **232.600 RGB value 3x(0..255)** **12.001 counter pulses**
The communication object is used to input a value that needs to be converted.					
10	**Output ...**	**1st /.../8th Logic**	**1bit** **2bit** **1byte** **2byte** **3byte** **4byte**	**C,T**	**1.001 switch** **2.001 switch control** **5.010 counter pulses(0..255)** **7.001 pulses** **232.600 RGB value 3x(0..255)** **12.001 counter pulses**
The communication object is used to output the converted value.					

Table 6.5.4 "Format convert" communication object table

6.5.5. “Gate function” Communication Object

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
2	1st Logic	Input			1 bit	C	-	W	-	-	switch	Low
3	1st Logic	Gate input			1 bit	C	-	W	-	-	boolean	Low
10	1st Logic	Output			1 bit	C	-	-	T	-	switch	Low

Input/Output - 1bit[On/Off]

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
2	1st Logic	Input			1 byte	C	-	W	-	-	percentage (0..100%)	Low
3	1st Logic	Gate input			1 bit	C	-	W	-	-	boolean	Low
10	1st Logic	Output			1 byte	C	-	-	T	-	percentage (0..100%)	Low

Input/Output - 1byte[0..100%]

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
2	1st Logic	Input			1 byte	C	-	W	-	-	counter pulses (0..255)	Low
3	1st Logic	Gate input			1 bit	C	-	W	-	-	boolean	Low
10	1st Logic	Output			1 byte	C	-	-	T	-	counter pulses (0..255)	Low

Input/Output - 1byte[0..255]

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
2	1st Logic	Input			2 bytes	C	-	W	-	-	temperature (°C)	Low
3	1st Logic	Gate input			1 bit	C	-	W	-	-	boolean	Low
10	1st Logic	Output			2 bytes	C	-	-	T	-	temperature (°C)	Low

Input/Output - 2byte[Float]

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
2	1st Logic	Input			2 bytes	C	-	W	-	-	pulses	Low
3	1st Logic	Gate input			1 bit	C	-	W	-	-	boolean	Low
10	1st Logic	Output			2 bytes	C	-	-	T	-	pulses	Low

Input/Output - 2byte[0..65535]

Fig.6.5.5 “Gate function” Communication Object

NO.	Object Function	Name	Data Type	Flag	DPT
2	**Input**	**1st /.../8th Logic**	**1bit** **1byte** **2byte**	**C,W**	**1.001 switch** **5.001 percentage** **5.010 counter pulses** **9.001 temperature** **7.001 pulses**
The communication object is used to input a value that needs to gate filter.					
3	**Gate input**	**1st /.../8th Logic**	**1bit**	**C,W**	**1.002 boolean**
The communication object is used to control the switch status of gate input. Input signal is allowed to pass when gate open, then output, and the current input status is still sent if there is a change; Can not pass when gate close.					
10	**Output**	**1st /.../8th Logic**	**bit** **1byte** **2byte**	**C,T**	**1.001 switch** **5.001 percentage** **5.010 counter pulses** **9.001 temperature** **7.001 pulses**
The communication object is used to output the value after gate filtering. Only when gate input status is open, output is available, defined by the object “Gate input”.					

Table 6.5.5 “Gate function” communication object table

6.5.6. “Delay function” Communication Object

	Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
	2	1st Logic	Input			1 bit	C	-	W	-	-	switch	Low
	10	1st Logic	Output			1 bit	C	-	-	T	-	switch	Low

Input/Output - 1bit[On/Off]

	Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
	2	1st Logic	Input			1 byte	C	-	W	-	-	percentage (0..100%)	Low
	10	1st Logic	Output			1 byte	C	-	-	T	-	percentage (0..100%)	Low

Input/Output - 1byte[0..100%]

	Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
	2	1st Logic	Input			1 byte	C	-	W	-	-	counter pulses (0..255)	Low
	10	1st Logic	Output			1 byte	C	-	-	T	-	counter pulses (0..255)	Low

Input/Output - 1byte[0..255]

	Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
	2	1st Logic	Input			2 bytes	C	-	W	-	-	temperature (°C)	Low
	10	1st Logic	Output			2 bytes	C	-	-	T	-	temperature (°C)	Low

Input/Output - 2byte[Float]

	Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
	2	1st Logic	Input			2 bytes	C	-	W	-	-	pulses	Low
	10	1st Logic	Output			2 bytes	C	-	-	T	-	pulses	Low

Input/Output - 2byte[0..65535]

Fig.6.5.6 “Delay function” Communication Object

NO.	Object Function	Name	Data Type	Flag	DPT
2	**Input**	**1st /.../8th Logic**	**1bit** **1byte** **2byte**	**C,W**	**1.001 switch** **5.001 percentage** **5.010 counter pulses** **9.001 temperature** **7.001 pulses**
The communication object is used to input a value that needs to delay.					
10	**Output**	**1st /.../8th Logic**	**1bit** **1byte** **2byte**	**C,T**	**1.001 switch** **5.001 percentage** **5.010 counter pulses** **9.001 temperature** **7.001 pulses**
The communication object is used to output that needs to delay converted value, delay time is defined by the parameter.					

Table 6.5.6 “Delay function” communication object table

6.5.7. “Staircase lighting” Communication Object

Numb	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
2	1st Logic	Trigger value			1 bit	C	-	W	-	-	trigger	Low
3	1st Logic	Light-on duration time			2 bytes	C	-	W	-	-	time (s)	Low
10	1st Logic	Output			1 bit	C	-	-	T	-	switch	Low
10	1st Logic	Output			1 byte	C	-	-	T	-	counter pulses (0..255)	Low

Fig.6.5.7 “Staircase lighting” Communication Object

NO.	Object Function	Name	Data Type	Flag	DPT
2	**Trigger value**	**1st /.../8th Logic**	**1bit**	**C,W**	**1.017 trigger**
The communication object is used to receive the value to trigger staircase lighting.					
3	**Light-on duration time**	**1st /.../8th Logic**	**2byte**	**C,W**	**7.005 time(s)**
The communication object is used to modify the staircase light-on duration time, the modified range is referenced from the range defined by the parameter, take the limit value if exceeded.					
10	**Output**	**1st /.../8th Logic**	**1bit** **1byte**	**C,T**	**1.001 switch** **5.010 counter pulses**
The communication object is used to output value 1 when trigger, and send value 2 after duration time. Telegram value is determined by the parameter setting datatype.					

Table 6.5.7 “Staircase lighting” communication object table

Chapter 7 UI Description for Tuya Smart APP

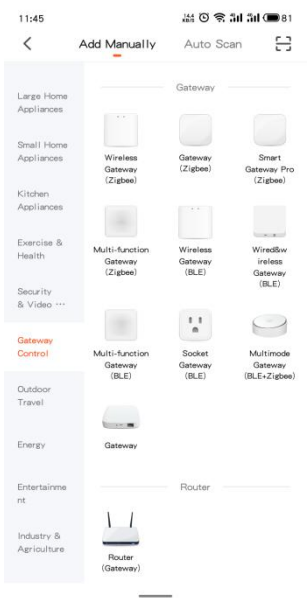
This chapter detail UI use for Tuya Smart APP in functional blocks. Please make sure the database is downloaded to KNX Gateway for Tuya ZigBee before operating APP. Once the database is downloaded, you can add gateways and other devices on the APP.

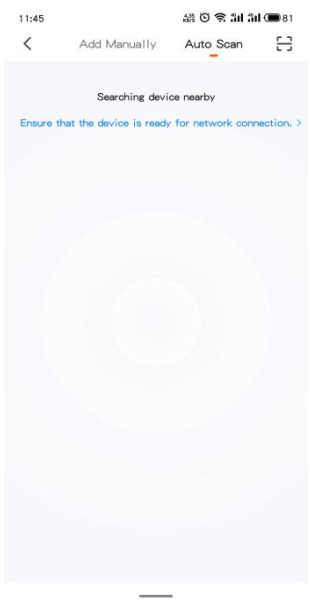
7.1. Add device

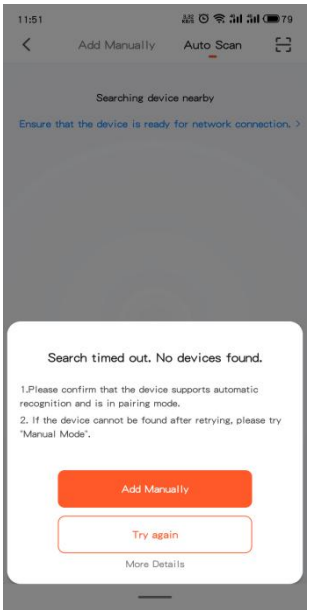
Click icon on the mobile phone desk to start up Tuya Smart APP, register or login the account. Click plus icon in the upper right corner of homepage to add device after login successfully. Take KNX Gateway for Tuya ZigBee as an example to explain the operation of adding a device.

Other devices are added in similar ways to the gateway, and no longer repeat in this chapter.

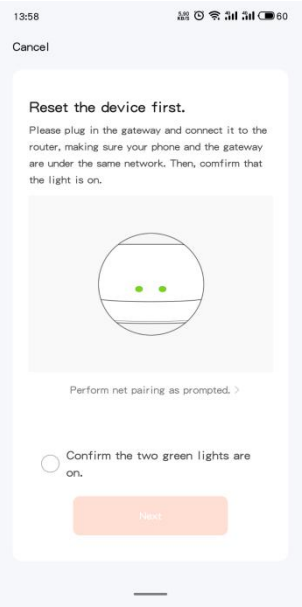
1. Select corresponding device type, KNX Gateway for Tuya ZigBee selects the Gateway in Gateway Control. You can select to add it manually(Step 2~4), or auto scan, please ensure that the device is ready for network connection, click blue word in the follow picture to go to the details of reset and network configuration. If failed to auto add device, you can select to Try again or Add Manually.







2. Click device icon, you can see the device connection steps. If the device is not reset, please perform net pairing as prompted. If it has been reset, ignore step 3. Please note the LED indication status on the gateway.



3. Reset device. Long press ZigBee connect button on the gateway for 5s, to remove the gateway from the cloud, triggered to flash after manually operating the connect button, and flashing 5 times to enter the waiting connection status. LED indication effect of gateway as follow:

LAN LED—Green LED:

——Always on after gateway network is connected OK, flashing when there is a data communication.

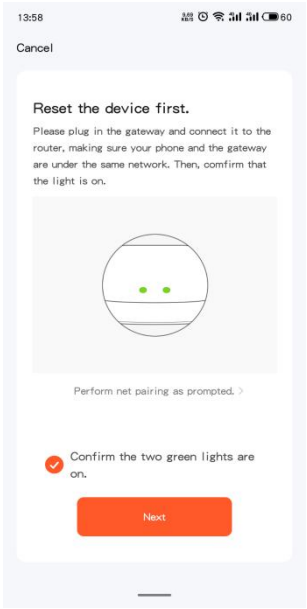
Connect status LED—Green LED:

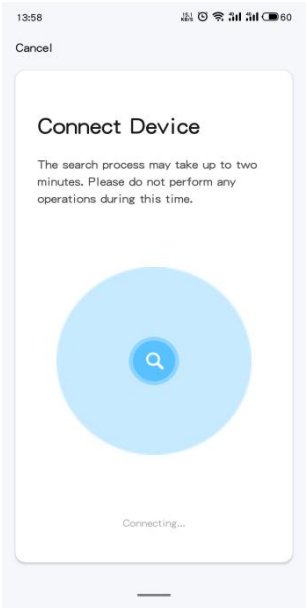
——Off when gateway is connected to cloud;

——Always on when connection is waiting to connect;

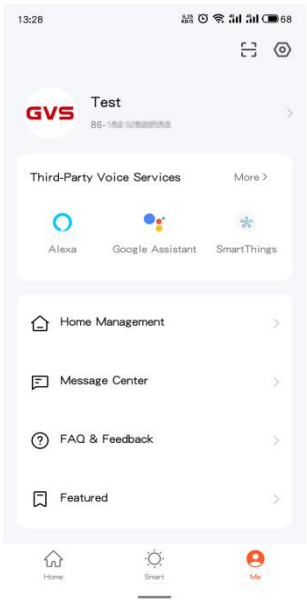
——Always flashing when Tuya authorization code is abnormal(Unprogram).

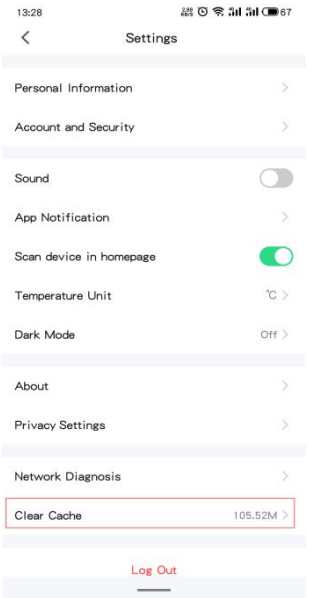
4. Make sure the device reset is successful, and your phone and the gateway are under the same network. Then make sure the device LED is indicated correctly. The gateway confirms that the two green lights(Connect status LED and LAN LED) are always on, then click Next to enter the status for connecting device.





Note: For Android systems, if the gateway fails to connect several times, it need to clear cache. Click icon in the upper right corner of【Me】interface to enter the settings page, select【Clear Cache】.





7.2. Switch interface

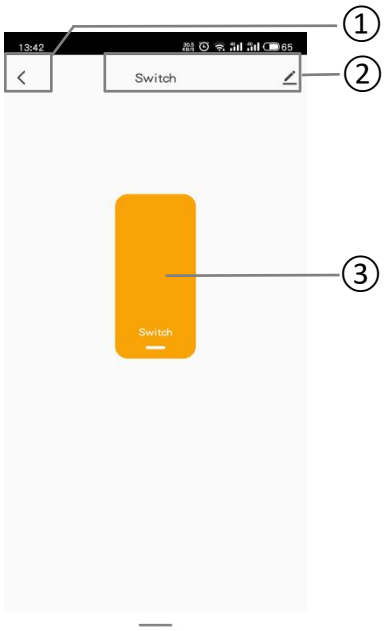


Fig.7.2(1) Switch interface

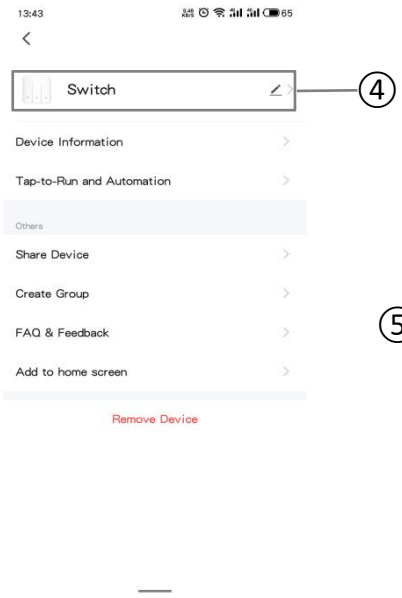


Fig.7.2(2) Device basic settings

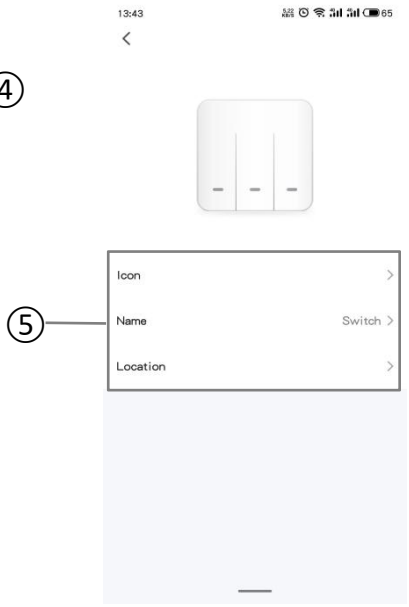


Fig.7.2(3) Info editing interface

①Return to the previous interface.

②Device name, shown it above the interface. Click icon to go to figure7.2(2), you can view, change device settings, and remove devices.

Device basic settings as shown in figure 7.2 (2), including Device Information, Tap-to-Run and Automation, Other settings, and Remove Device. Set the Tap-to-Run and Automation via KNX scene switch, details in chapter 7.7.

Click ④ to go to figure7.2(3), you can edit current device name, icon and location.

③Switch button, gray indicates status off, bright color indicates status on. Used for switching of switch status.

7.3. Switch/Dimming interface

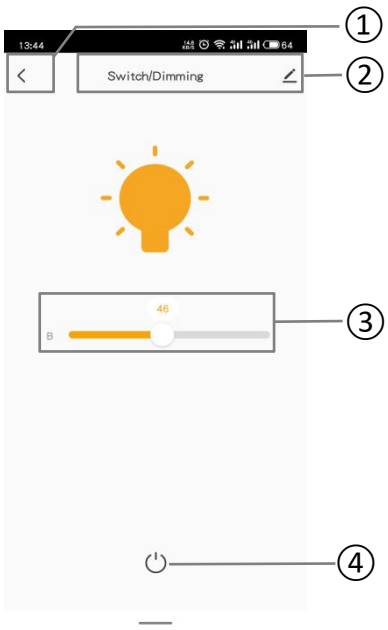


Fig.7.3(1) Switch/Dimming interface

①Return to the previous interface.

②Device name, shown it above the interface. Click icon , you can view, change device settings, and remove devices.

③B: brightness adjustment. Adjust brightness by sliding this slider, range: 0..100%

④LED switch button, gray indicates status off, bright color indicates status on. Used for switching of switch status. When brightness value is not 0, touch this button to close brightness, and value is 0.

7.4. RGB Dimming interface

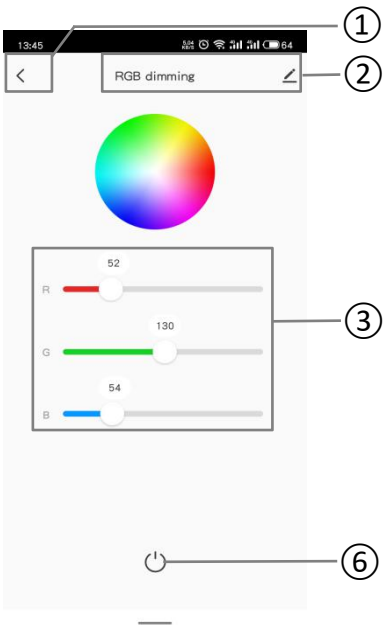


Fig.7.4(1) RGB dimming interface

Fig.7.4(2)RGBW dimming interface

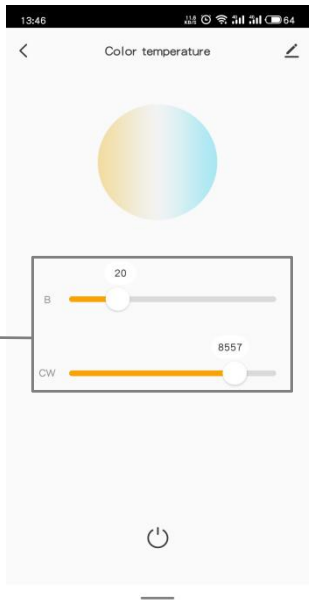


Fig.7.4(3)Color temperature interface

①Return to the previous interface.

②Device name, shown it above the interface. Click icon , you can view, change device settings, and remove devices.

There are 3 control types of RGB dimming function: RGB, RGBW, Color Temperature. Among them, RGB is suitable for adjusting RGB lamp; RGBW is suitable for controlling RGBW strip; Color Temperature is suitable for color temperature control. Shown as figure7.4(1)(2)(3).

③RGB value adjustment. Adjust brightness by sliding R/G/B slider, range: 0..255

④RGBW value adjustment. Adjust brightness by sliding R/G/B/W slider, range: 0..255

⑤B: brightness adjustment. Adjust brightness by sliding this slider, range: 0..100%

CW: color temperature adjustment. Adjust color temperature by sliding this slider, range: 1000...10000K

⑥LED switch button, gray indicates status off, bright color indicates status on. When current value is not 0, touch this button to close R/G/B/W or B/CW, and value is 0.

7.5. Curtain control interface

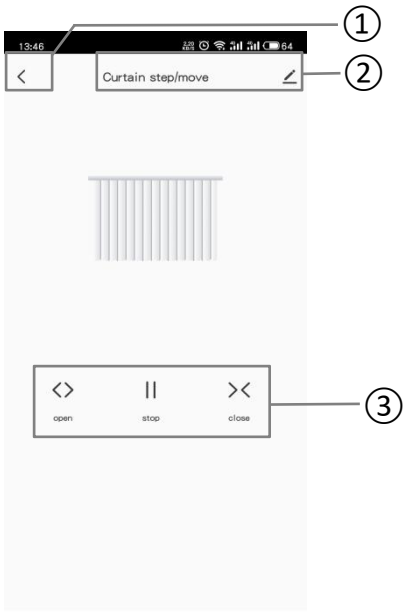


Fig.7.5(1)Curtain step/move

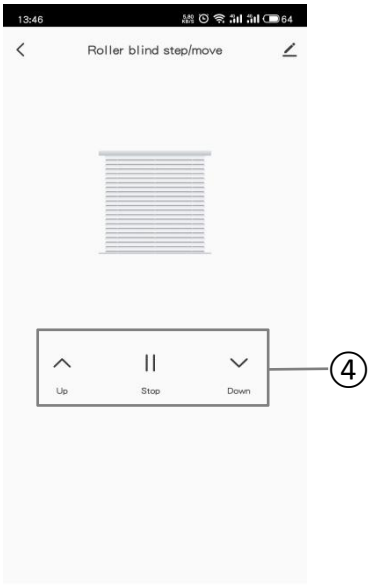


Fig.7.5(2)Roller blind step/move

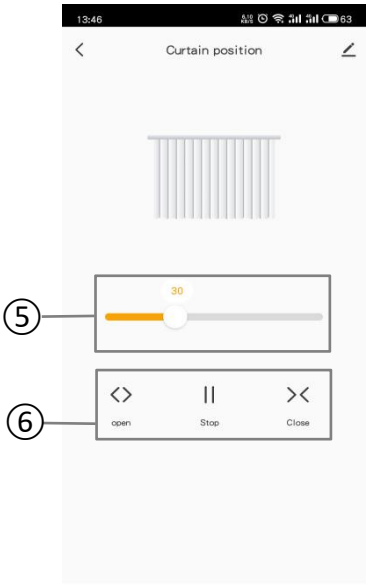


Fig.7.5(3)Curtain position

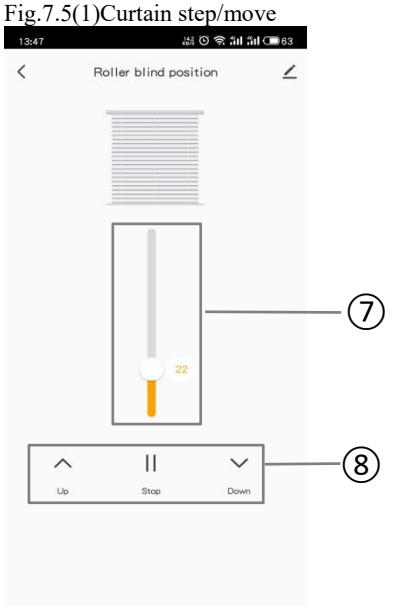


Fig.7.5(4)Roller blind position

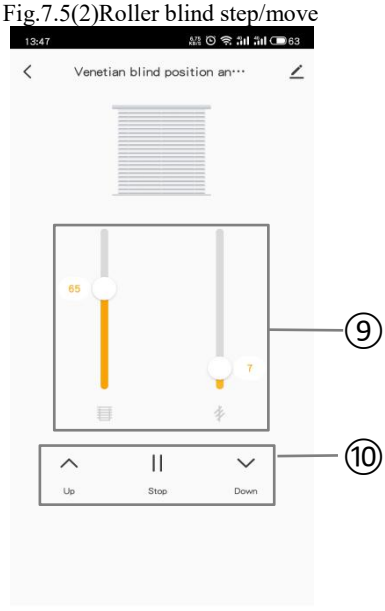


Fig.7.5(5)Venetian blind position and slat

①Return to the previous interface.

②Device name, shown it above the interface. Click icon , you can view, change device settings, and remove devices.

③Curtain step/move:

1)Touch icon <>, open curtain. Device sends out a telegram with the object(Open/Close) value of 0;

2)Touch icon ||, stop curtain. Device sends out a telegram with the object(Stop) value of 1;

3)Touch icon ><, close curtain. Device sends out a telegram with the object(Open/Close) value of 1;

④Roller blind step/move:

1)Touch icon ^, move up blind. Device sends out a telegram with the object(Up/Down) value of 0;

2)Touch icon , stop blind. Device sends out a telegram with the object(Stop) value of 1;

3)Touch icon , move down blind. Device sends out a telegram with the object(Up/Down) value of 1;

⑤⑥Curtain position:

1)By sliding the slider⑤ to adjust the position of curtain;

2) Operation of Open/Close/Stop are the same as above.

⑦⑧Roller blind position:

1)By sliding the slider⑦ to adjust the position of blind;

2) Operation of Up/Down/Stop are the same as above.

⑨⑩Venetian blind position and slat:

1)By sliding the slider⑨ to adjust the position and louver angle of venetian blind;

2) Operation of Up/Down/Stop are the same as above.

7.6. Value sender interface

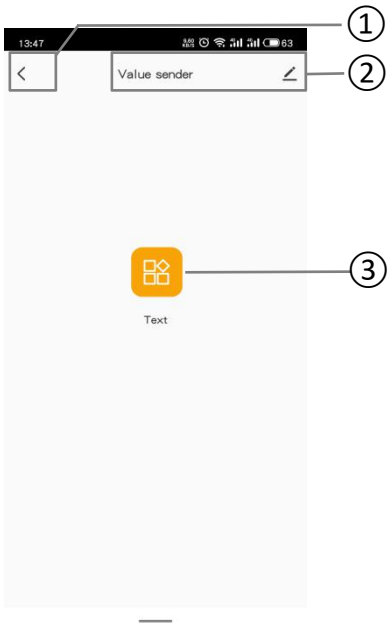


Fig.7.6Value sender

①Return to the previous interface.

②Device name, shown it above the interface. Click icon , you can view, change device settings, and remove devices.

③Touch icon, send telegram to KNX bus. Only used to trigger calling command on the APP, with only button operation.

There are 8 value types sent: 1bit[On/Off], 2bit[0..3], 1byte[0..255], 1byte[0..100%], 1byte[scene control], 2byte[-671 088.64..670 760.96], 2byte[0..65535].

7.7. KNX Scene switch interface

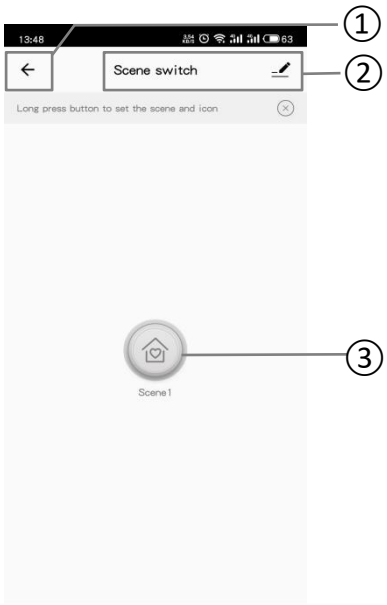


Fig.7.7(1) Scene switch interface

①Return to the previous interface.

②Device name, shown it above the interface. Click icon , you can view, change device settings, and remove devices.

③Long press to enter the scene setting, receive KNX scene calling command, or short press on the APP to perform action set in scene setting, link the KNX device functions to the ZigBee on the APP. Configure specific associated devices and trigger responses for scene on the APP before operation.

Add scene setting:

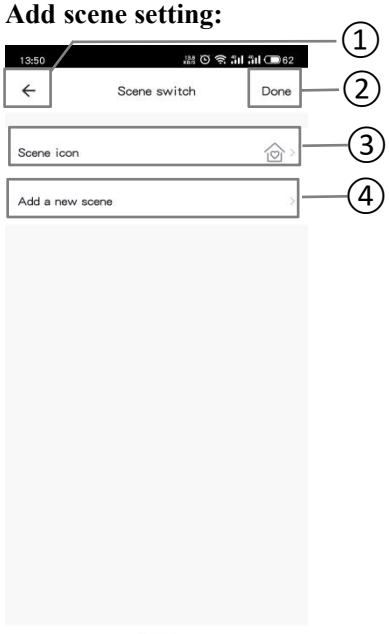


Fig.7.7(2) Scene switch setting

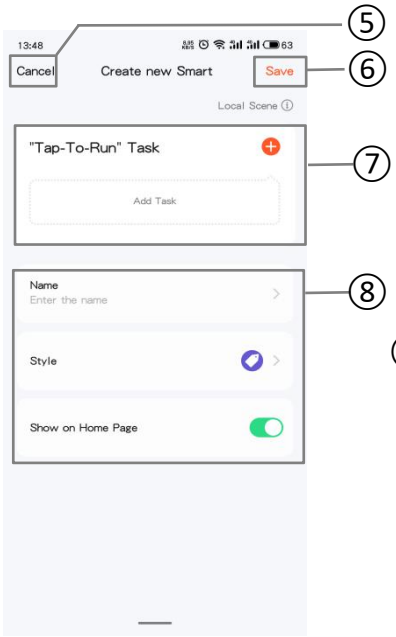


Fig.7.7(3) Create Smart

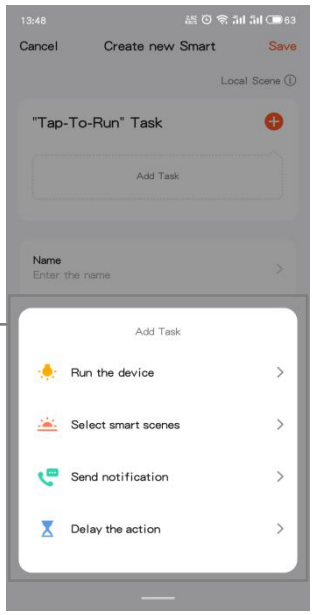


Fig.7.7(4) Add task

①⑤Return to the previous interface.

②Click to save the setting and return to the previous interface after scene switch setting.

③Click to modify scene icon.

④Click to go to the interface Create Smart, shown as figure7.7(3).

⑥Click to save the setting and return to the previous interface after creating smart.

⑦Click plus icon to create"Tap-To-Run" task, the window⑨ is pop up below, you can select required setting, including: all devices, smart scenes, notification and the action delay. Task is shown in figure 7.7(3) after creating successfully.

⑧Set the task name, display style, and whether to show on home page.

Manage scene setting:

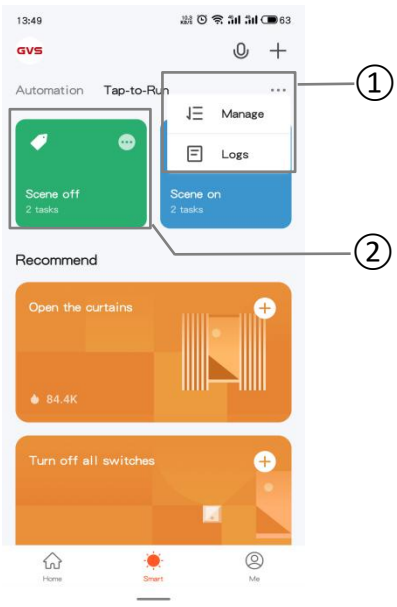


Fig.7.7(5) Smart interface

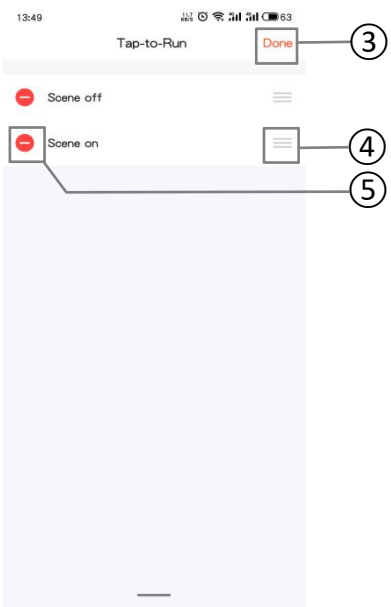


Fig.7.7(6) Manage interface

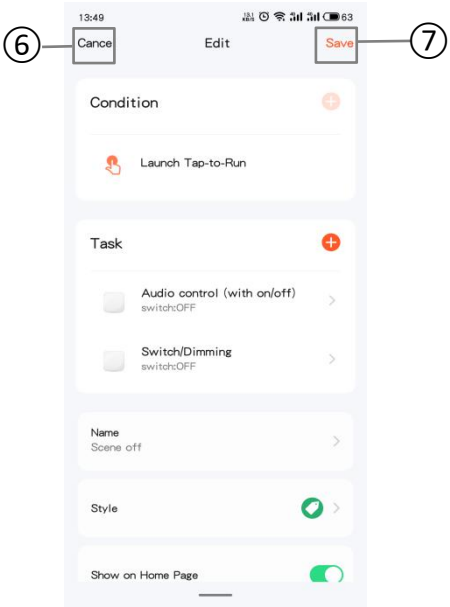


Fig.7.7(7) Edit interface

①Click three-point icon, display drop-down window, you can select 【Manage】 or 【Logs】, shown as figure 7.7(5). Click【Manage】to go to manage interface, shown as figure 7.7(6). Long press icon④ to slide, to adjust the order of scene tasks; click icon⑤ to delete the scene task.

②Click three-point icon, go to the scene edit interface, shown as 7.7(7), operation in the interface: add task, change the task name and display style, whether to show on home page, and delete the scene. Click on the rest of the control to perform the scene immediately.

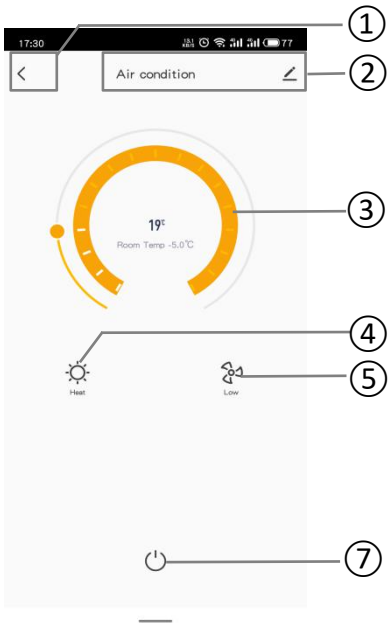
Operate the edit interface also can be via the single device, such as figure 7.2 (2) device basic setting, to click Tap-to-Run and Automation.

③Click to save and return to previous interface after task management.

⑥Click to cancel edit, and return to previous interface.

⑦Click to save and return to previous interface after editing.

7.8. Air condition interface



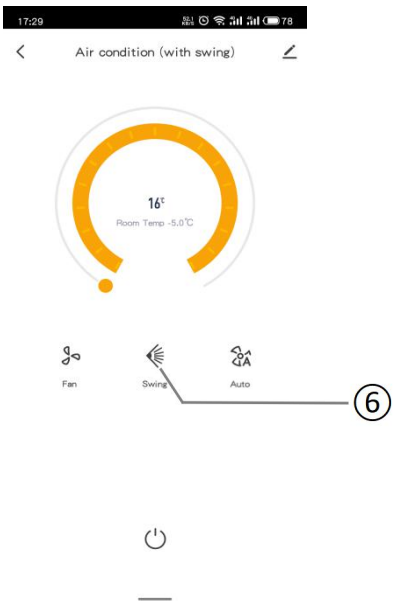


Fig.7.8(1) Air condition interface Fig.7.8(2) Air condition (with swing) interface

①Return to the previous interface.

②Device name, shown it above the interface. Click icon , you can view, change device settings, and remove devices.

③Ring slider is used to adjust the setpoint temperature, the performed adjustment step is 1℃. When temperature unit is set to degrees Celsius (℃), adjustment range of setpoint temperature is default as 16～32℃; Current temperature display according the parameter configuration, you can configure to show detected temperature of internal senor or external sensor.

④Click this icon to switch Air condition control mode, update the mode status via the object: Heat , Cool , Dehumidification , Fan , Auto .

⑤Click this icon to switch fan speed, the current fan speed is cyclically switched in this order--Low speed...AUTO...Low speed: Low speed , Medium speed , High speed , Auto .

⑥Click this icon switch swing control: swing, stop.

⑦Air condition switch button, gray indicates status off, bright color indicates status on. When power off, except for the status of power on/off, the other icon status can not be updated via the bus or manually controlled. Click this icon, device sends a telegram of value 1/0 (On/Off) via the object Power on/off.

7.9. Room temperature unit interface

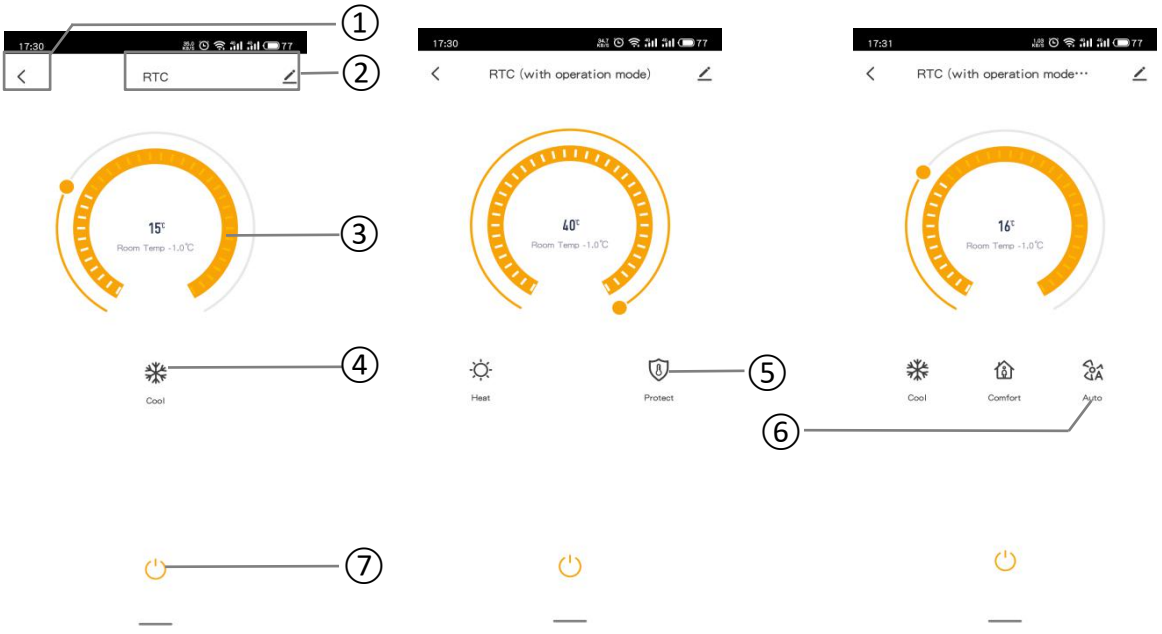


Fig.7.9(1) RTC interface Fig.7.8(2) RTC (with operation mode) interface Fig.7.8(3) RTC (with operation mode&fan speed) interface

①Return to the previous interface.

②Device name, shown it above the interface. Click icon, you can view, change device settings, and remove devices.

③Ring slider is used to adjust the setpoint temperature, the performed adjustment step is 1°C. When temperature unit is set to degrees Celsius (°C), adjustment range of setpoint temperature is default as 5～40°C; Current temperature display according the parameter configuration, you can configure to show detected temperature of internal senor or external sensor.

④Click this icon to switch control mode, update the mode status via the object: Heating, cooling.

You can switch the mode via touch or telegram when the database sets the RTC control mode to “Heating and Cooling”.

⑤Click this icon to switch operation mode: comfort, standby, economy, protection

⑥Click this icon to switch fan speed, the current fan speed is cyclically switched in this order--Low speed...AUTO...Low speed: Low speed, Medium speed, High speed, Auto.

⑦RTC switch button, gray indicates status off, bright color indicates status on. When power off, except for the status of power on/off, the other icon status can not be updated via the bus or manually controlled. Click this icon, device sends a telegram of value 1/0 (On/Off) via the object Power on/off.

7.10. Ventilation system interface

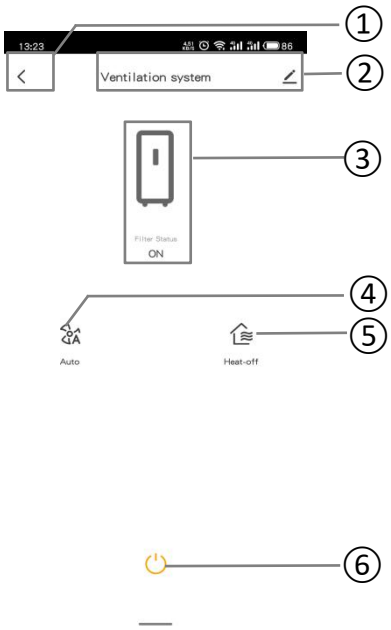


Fig. 7.10 Ventilation system control interface

①Return to the previous interface.

②Device name, shown it above the interface. Click icon, you can view, change device settings, and remove devices.

③It is used to display the switch status of filter.

④Click this icon to switch fan speed, the current fan speed is cyclically switched in this order--Low speed...AUTO...Low speed: Low speed, Medium speed, High speed, Auto.

⑤Click the icon to turn on or off heat recovery function.

⑥Ventilation switch button, gray indicates status off, bright color indicates status on. When power off, except for

the status of power on/off, the other icon status can not be updated via the bus or manually controlled. Click this icon, device sends a telegram of value 1/0 (On/Off) via the object Power on/off.

7.11. Audio control interface

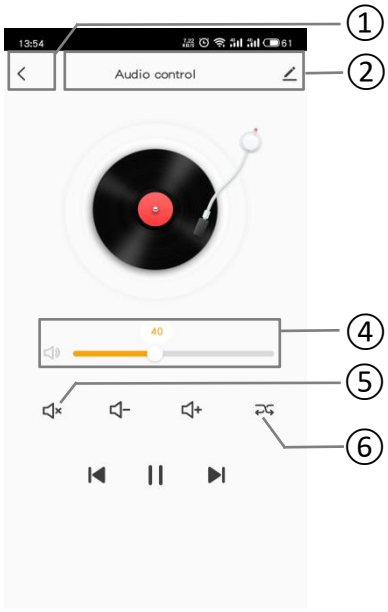


Fig.7.11(1) Audio control interface

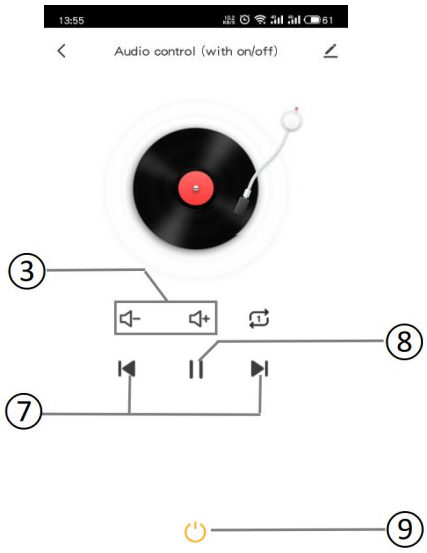


Fig.7.11(2) Audio control (with on/off) interface

①Return to the previous interface.

②Device name, shown it above the interface. Click icon , you can view, change device settings, and remove devices.

③Click the icon to adjust the volume in increasing or decreasing increments.

④Slide the slider to adjust the volume decrease/increase.

⑤Click the icon to mute/exit mute: mute , exit mute .

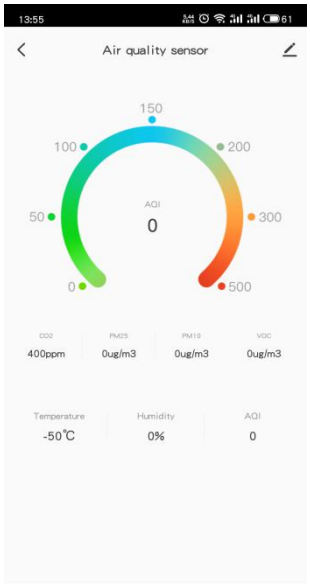
⑥Click the button to switch the audio play mode. Update the status via the object: single loop , list play , random play .

⑦Click the icon to select previous song or next song.

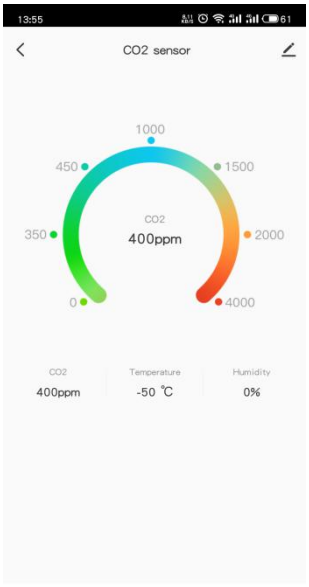
⑧Click the icon to play or stop music. And update the status via the object.

⑨Audio control switch button, gray indicates status off, bright color indicates status on. When power off, except for the status of power on/off, the other icon status can not be updated via the bus or manually controlled. Click this icon, device sends a telegram of value 1/0 (On/Off) via the object Power on/off.

7.12. Sensor interface

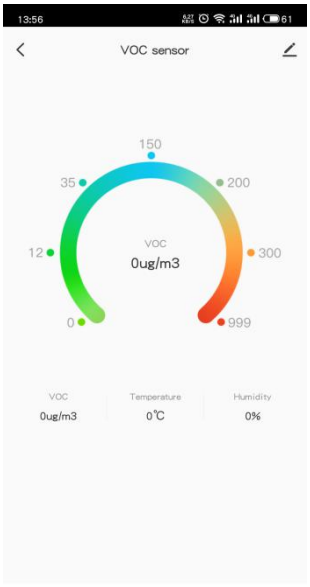


Air quality sensor interface



CO2 sensor interface

PM2.5 sensor interface



VOC sensor interface

Air quality sensor displays 7 reference values, independently is: CO2, PM2.5, PM10, VOC, Temperature, Humidity, AQI. What displays can be configured through the database. The CO2, PM2.5 and VOC have separate detectors, and all can show the internal temperature and humidity on the interface.

The ring on the interface and the reference values of corresponding devices. The update of the display value can be achieved by rewriting the object via the bus.

Detection range of AQI: 0..500

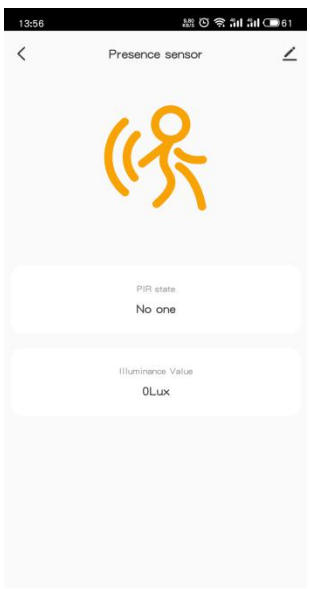
Detection range of CO2: 0..4000ppm

Detection range of PM2.5: 0..999ug/m^3

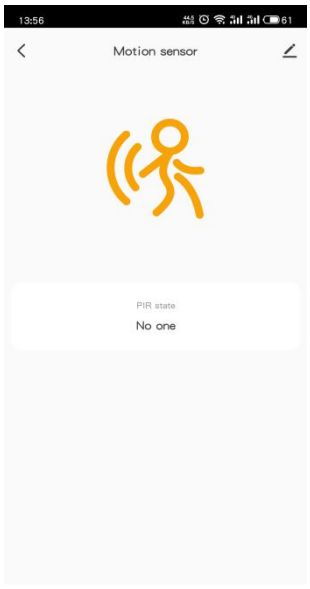
Detection range of VOC: 0..9.99ppm

Detection range of temperature: -50...90°C

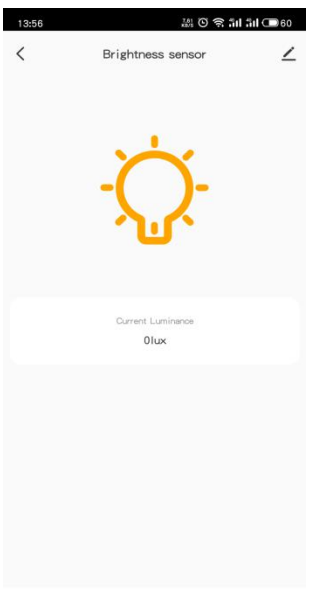
Detection range of humidity: 0..100%



Presence sensor interface



Motion sensor interface



Brightness sensor interface

I/O signal interface

Presence sensor can detect PIR state, and illuminance, detection range: 0...65535lux

Motion sensor can detect PIR state.

Brightness sensor can detect the illuminance, detection range: 0...65535lux

I/O signal can detect the device switch status.

7.13. Current/Energy metering interface

Current metering

Energy metering

Current metering detects the current and power, and display the changing status in real-time, the values unit and resolution can be rewrite via the object on the bus.

Energy metering detects the current, power, voltage and energy, and display the changing status in real-time, the values unit and resolution can be rewrite via the object on the bus.