KNX-Gateway-LG1
LMG-GWTY2/25.1.1
User Manual-V1.0

KNX International Standard Home and Building Control System

Contents

Version	Time	Revision Record
1.0	26-0107	Version 1.0 Released
1.1	26-0108	Added explanatory notes for some configuration parameters and communication objects.

1. Overview

The KNX Gateway is an intelligent device that integrates KNX technology with wireless communication technologies (ZigBee, Bluetooth protocol). It supports seamlessly uploading KNX devices to the Tuya platform, enabling users to remotely control and centrally manage KNX, ZigBee, and Bluetooth devices through the Tuya platform. Additionally, it serves as a bridge connecting various smart devices, facilitating interconnected control between devices using different protocols.

This manual provides users with detailed technical information about the KNX Gateway, along with practical use cases to illustrate its usage. The KNX Gateway requires connection to the KNX bus and a 21-30V DC auxiliary power supply for operation. Users need to configure the KNX channel devices of the gateway using the ETS software (version ETS5.7 or higher) and add the gateway device via the Letour Smart Life App to manage and integrate KNX, ZigBee, and Bluetooth devices.

The main features are summarized as follows:

- Supports uploading KNX devices to the Tuya platform for management, enabling control and status display of KNX devices.

- Compatible with KNX system device functions, including switching, dimming, RGB dimming, RGBW dimming, dual-color temperature adjustment, curtains (open/close curtains, roller blinds, Venetian blinds), scenes, value transmission, HVAC control (thermostat, air conditioning, ventilation systems, floor heating), background music, various sensors (air quality, temperature/humidity, brightness, gas, switch signals, etc.), as well as current and energy consumption information display. (Note: When the gateway is powered on, it requests the status of all connected devices.)

- Functions as a ZigBee gateway, connecting to the Tuya cloud platform. It supports adding/managing Tuya ecosystem ZigBee sub-devices to the cloud, enabling remote control and real-time status monitoring.

- Functions as a Bluetooth gateway, supporting the adding/management of Tuya private Bluetooth protocol and Bluetooth Mesh protocol (SIG) devices. It forwards data to the cloud for remote control and real-time status monitoring.

- Through automation creation and scene functions in the app, bidirectional control and mutual linkage between KNX devices, ZigBee devices, and Bluetooth devices can be achieved.

- Supports 8 KNX channel logic functions.

- Supports 8 sets of event functions (each set includes 8 configurable outputs).

2. Technical Parameters

Power Supply	KNX bus voltage	21~30V DC, obtained through EIB bus
	Current consumption via bus	≤12mA (30V DC)
	Power consumption via bus	≤360mW
Auxiliary Power Supply	Auxiliary Voltage	21~30V DC
	Power Consumed via Auxiliary Power Supply	
Wiring	EIB/KNX bus	Terminal connection (red/black)
	Output terminal	0.8mmØ, connection with copper posts with screw terminals
	Use wire diameter	0.5-4mm^2
	Torque	0.8N-m
Operation & Instruction	Program button	Used for device programming physical address and diagnostics
	Red LED indicator	Instructs the device to enter programming mode
	Green LED indicator	Instructs the device to enter

Temperature Range	Operating	-5℃……+45℃
	Storage	-25℃……+55℃
	Transport	-25℃……+70℃
Environmental Conditions	Humidity	Max. air humidity <93%, except condensation
Installation	Installation	Installed in a standard 86mm electrical box
Dimension/ Weight	Dimension	72mm×90mm×64mm
	Weight	0.149kg

Application Program:

Application Program	Max. Communication Objects	Max. Group Addresses	Max. Association Entries
KNXGatewayLG1V1_3	6616	6616	6616

3. Dimension Drawing and Wiring Diagram

3.1 Dimension Drawing
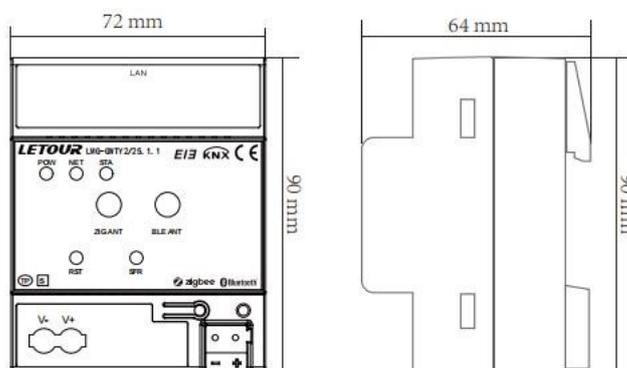
3.2 Wiring Diagram
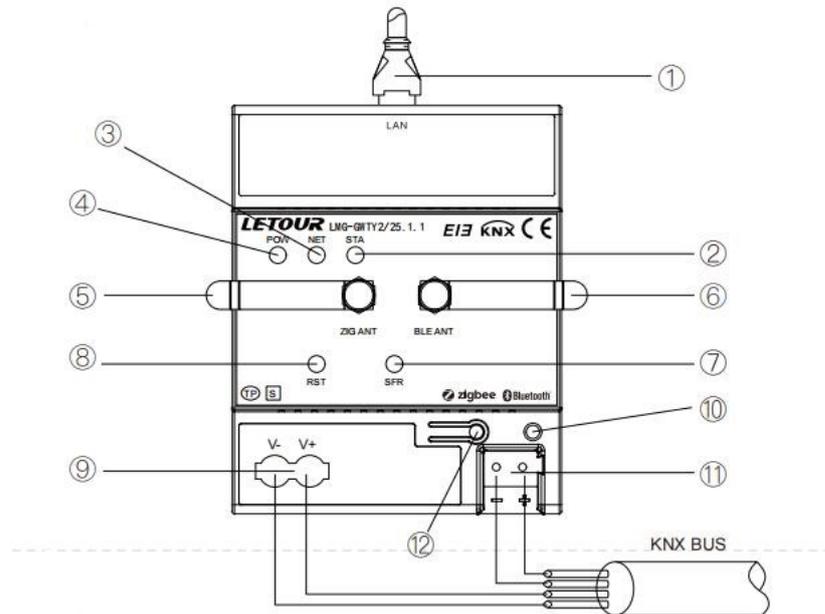
①   Network Interface

②   Bluetooth or ZigBee Sub-device Pairing Indicator (flashes while searching for devices)

③   Pairing Indicator (turns off after successful network connection; stays lit if connection fails)

④   POW Indicator, indicates normal power supply (solid light)

⑤   Bluetooth and ZigBee Antennas

⑥   Bluetooth and ZigBee Antennas

⑦   Reset Button. Press and hold for 5 seconds to reset the device.

⑧   Hardware reset. Press briefly to restart the hardware

⑨   Auxiliary Power Supply

⑩   Programming Button Indicator

⑪   KNX/EIB Bus Connection Terminal

⑫   Programming Button

4. ETS System Parameter Setting Guide

4.1 Parameter Setting Interface "General"

4.1.1 Parameter Setting Interface - "General Setting"

Figure 4.1.1 "General Setting" Parameter Setting Interface

*Parameter "Device ID"

This parameter sets the device ID. Options: 0 … 65535

*Parameter "Device Name"

This parameter sets the device name. Up to 32 characters can be entered.

*Parameter "Send delay after power on [0..15]s"

This parameter sets the delay time for the gateway to send messages to the bus after a power-on reset. Options: 0..15

This setting does not include the gateway initialization time, and bus messages received during the delay period will be recorded.

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

This parameter sets the time interval for the module to cyclically send messages via the bus indicating normal operation. When set to "0", the "In operation" object will not send messages. If set to a value other than "0", the "In operation" object will send a message with logic "1" to the bus at the set cycle. Options: 0 … 240 (0 = cyclic sending disabled).

To minimize bus load as much as possible, the maximum time interval should be selected based on actual requirements.

Note: Functions marked with * are temporarily not supported for configuration or use.

4.2 Parameter Setting Interface "KNX Channel"

4.2.1 Parameter Setting Interface "Channel setting"

Figure 4.2.1 "General Setting" Parameter Setting Interface

Parameter "Number of KNX Channels"

This parameter sets the number of KNX channels. Options: 1-200.

*Parameter "Status object read request after restart"

This parameter determines whether to send a status read request message upon device restart.

*Parameter "Send request delay between status objects"

This parameter is visible only when the previous parameter is enabled. It sets the interval for sending power-on status request messages. Options:

50ms

100ms

200ms

Device online status request setting for common x (x = 1–10)

*Parameter "Time period request for common x [0..255, 0 = inactive] min" (x = 1–10)

This parameter sets the period for sending read requests to query the online status of common devices. A value of 0 disables the query function. Options: 0–255.

This function is primarily used to query the online status of KNX devices. Requests begin after the gateway's power-on send delay time has elapsed.

Note: Functions marked with * are temporarily not supported for configuration or use.

4.2.2 Parameter Setting Interface "Channel x" (x=1~200)
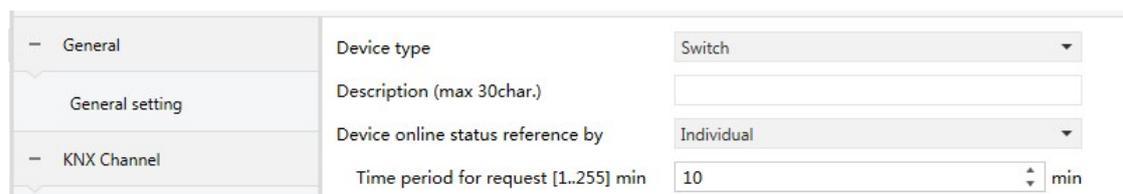
Figure 4.2.2 "Channel 1" Parameter Setting Interface

These parameters are common configuration settings for KNX device types and will not be reiterated in subsequent chapters.

Parameter "Device type"

Switch	Floor heating
Switch/Dimming	Ventilation system
RGB dimming	Temperature/Humidity sensor
RGBW dimming	Audio control
Color temperature	Audio control (with on/off)
Curtain step/move	Air quality sensor
Roller blind step/move	CO2sensor
Curtain position	PM2.5sensor
Roller blind position	VOC sensor
Venetian blind position and slat	IR sensor

Curtain position and slat	Presence sensor
Scene control	Motion sensor
Value sender	Brightness sensor
Air conditioner	I/O signal
Air conditioner (with swing)	SOS sensor
Room temperature unit	Energy metering
Room temperature unit(with operation mode)	
Room temperature unit (with operation mode&fan speed)	

Note: Channels 1-200 support all functional options.

Parameter "Description (max 30 char.)"

This parameter sets the device name description for the current channel. A maximum of 30 characters can be entered.

Parameter "Device online status reference by"

This parameter sets the reference type for sending read requests to KNX devices. Options include selecting requests for individual devices or common device requests, with 10 common device request options available. Options:

Common 1

...

Common 10

Individual

Always online

When selecting "Common", it refers to common device requests, suitable for devices with multiple circuits. For example, if multiple channels of the gateway control multiple circuits of the same KNX device, each channel can share a single request.

When selecting "Individual", it is suitable for devices controlled by a single channel of the gateway.

When selecting "Always online", it is suitable for KNX devices without heartbeat

packets, particularly for scenarios. Such devices are considered permanently online once configured.

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

This parameter is visible when "Individual" is selected. It sets the time period for individual device online status requests. Options: 1..255

4.2.2.1 Basic Function Parameters

This section describes the basic control function parameters of the KNX channels, including switching, dimming, curtains, color, color temperature, value transmission, and KNX scene switching functions.

"Switch" Parameter Settings

"Switch/Dimming" Parameter Settings

Curtain Parameter Settings

"RGB Dimming" Parameter Settings

"RGBW Dimming" Parameter Settings

"Color Temperature" Parameter Settings

"Value Sender" Parameter Settings

"Scene Control" Parameter Settings

Figure 4.2.2.1 Basic Control Function Parameters

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

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

When the device type is set to "Switch/Dimming", these two parameters are visible. They define the upper and lower threshold limits for brightness.

Upper threshold options: 0..50

Lower threshold options: 51..100

Parameter "Object datatype"

When the device type is set to "RGB dimming" or "RGBW dimming", this parameter is visible. It configures the object data type for RGB or RGBW dimming.

For RGB type:

1x 3byte: RGB dimming controlled via a single 3-byte object

3x 1byte: RGB dimming controlled via three 1-byte objects

For RGBW type:

1x 6byte: RGBW dimming controlled via a single 6-byte object

4x 1byte: RGBW dimming controlled via four 1-byte objects

Parameter "Min. color temperature [2000..7000]K"

Parameter "Max. color temperature [2000..7000]K"

When the device type is set to "Color temperature", these two parameters are visible. They set the upper and lower threshold limits for color temperature. Options: 2000..7000.

If the minimum and maximum values are set incorrectly (e.g., the minimum value is greater than the maximum), the full range will be applied. If the values are equal, only a single value will be used.

Note: In the advanced version BT MO-TY/00.3(4), for color temperature, the minimum value must always be less than the maximum value. If this condition is not met, the parameter cannot be set in ETS, and a red warning box will appear, as shown below：

| Min. color temperature [2000..7000] | 7000 | ÷ | K |
| Max. color temperature [2000..7000] | 6500 | ÷ | K |

Parameter "Output object datatype for trigger"

This parameter is visible when the device type is set to "Value sender". It configures the data type of the message value sent to the KNX bus when a command is triggered from 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 the device type is set to "Value sender" and the corresponding data type is selected. It defines the value of the message sent to the KNX bus when a command is triggered from the app. Options vary by data type:

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

This parameter is visible when the device type is set to "Scene control". It sets the time period for requests. Options: 1..255 minutes.

Parameter "Recall scene No."

This parameter is visible when the device type is set to "Scene control". It configures the reception of scene recall commands from the KNX system. Options: 1..64.

4.2.2.2 Air Conditioning Function Parameters

This section describes the air conditioning functions of the KNX channel, including basic air conditioning control and control with airflow direction. The following parameters are visible when "Air conditioner" or "Air conditioner (with swing)" is selected.

Figure 4.2.2.2(1) Air Conditioning Function Parameters

Figure 4.2.2.2(2) Air Conditioning Function Parameters

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

This parameter sets the time interval for sending requests to the indoor

temperature sensor. Options: 0..255.

A read request is automatically sent when the device restarts.

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

These two parameters define the adjustable range for the temperature setpoint. If the temperature setpoint exceeds this range, the limit value is used as the output. Options:

16°C

17°C

...

32°C

For temperature setpoints, the minimum value must always be less than the maximum value. If this condition is not met, the parameter cannot be set in ETS.

Control Mode Setting

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

When these parameters are checked, the corresponding mode settings become visible.

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

These parameters are visible when the mode is enabled, and they set the output value when switching to each mode. Options: 0..255.

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

These parameters are visible when the mode is enabled, and they set the status feedback value for each mode. Options: 0..255.

Parameter "Object datatype of 1byte fan speed"

This parameter sets the data type for the 1-byte fan speed object. 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 set the output values for switching to different fan speed levels:

auto, low, medium, and high. The options displayed depend on the object type selected in the previous parameter: 0..255 or 0..100.

Status feedback for fan speed

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

These parameters set the status feedback values for different fan speed levels: auto, low, medium, and high. The device updates the fan speed display based on these feedback values. The options displayed depend on the object type selected in the previous parameter: 0..255 / 0..100.

Note: The output values and status feedback values for fan speed must satisfy the condition: low < medium < high. There is no restriction for auto mode. If this condition is not met, the parameter cannot be set in ETS, and a red warning box will appear, as shown below:

Output value for fan speed auto	0	%
Output value for fan speed low	68	%
Output value for fan speed medium	67	%
Output value for fan speed high	100	%

4.2.2.3Temperature Control Function Parameters

This section describes the temperature control functions of the KNX channel, including basic control, control with operation mode, and control with fan speed. The following parameters are visible when selecting "Room temperature unit," "Room temperature unit (with operation mode)," or "Room temperature unit (with operation mode & fan speed)."

Figure 4.2.2.3 Temperature Control Function Parameters

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

This parameter is used to set the time interval for sending requests to the indoor temperature sensor. Options: 0..255.

A read request is automatically sent by default when the device restarts.

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

These two parameters are used to set the adjustable range for the temperature setpoint limit. If the temperature setpoint exceeds the limit range, the limit value will be used as the output. Options:

5°C

6°C

...

40°C.

For the setpoint temperature, the minimum value must always be less than the maximum value. If this condition is not met, the parameter cannot be configured in ETS.

Parameter "Control mode"

This parameter is used to set the temperature control mode, supporting three types: heating, cooling, and heating/cooling. Options:

Heating

Cooling

Heating and Cooling

Fan speed setting

Fan speed settings are only visible when "Room temperature unit (with operation mode & fan speed)" is selected.

Parameter "Object datatype of 1byte fan speed"

This parameter is used to set the data type for the 1-byte 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 set the output values for switching to low, medium, and high fan speed levels. Options are displayed based on the object type selected in the previous parameter: 0..255 or 0..100.

Status feedback for fan speed

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

These parameters set the status feedback values for low, medium, and high fan speed levels. The device will update the fan speed display based on these feedback values. Options are displayed based on the object type selected in the previous parameter: 0..255 or 0..100.

Note: The output values and status feedback values for fan speed must satisfy the condition: low < medium < high. If this condition is not met, the parameter cannot be configured in ETS, and a red warning box will be displayed, as shown below:

Output value for Fan speed low	68	%
Output value for Fan speed medium	67	%
Output value for Fan speed high	100	%

Parameter "1 bit object function for fan speed"

This parameter is used to enable or disable the 1-bit object control function for fan speed. When enabled, the 1-bit control objects for each fan speed level become visible. When all three objects are set to 0, the fan is turned off.

Parameter "1 bit object for fan speed off"

This parameter becomes visible when the previous parameter is enabled. It sets whether to enable the visibility of the "1-bit fan speed off" object.

Parameter "Power on/off is"

This parameter is used to enable or disable the power on/off function for the thermostat.

Options:

Disable

Enable

Note: Currently, the thermostat device type panel only supports the "Power on/off is" parameter being configured as "Enable".

4.2.2.4. Floor Heating Function Parameters

This section describes the floor heating function of the KNX channel. The following parameters are visible when "Floor heating" is selected.
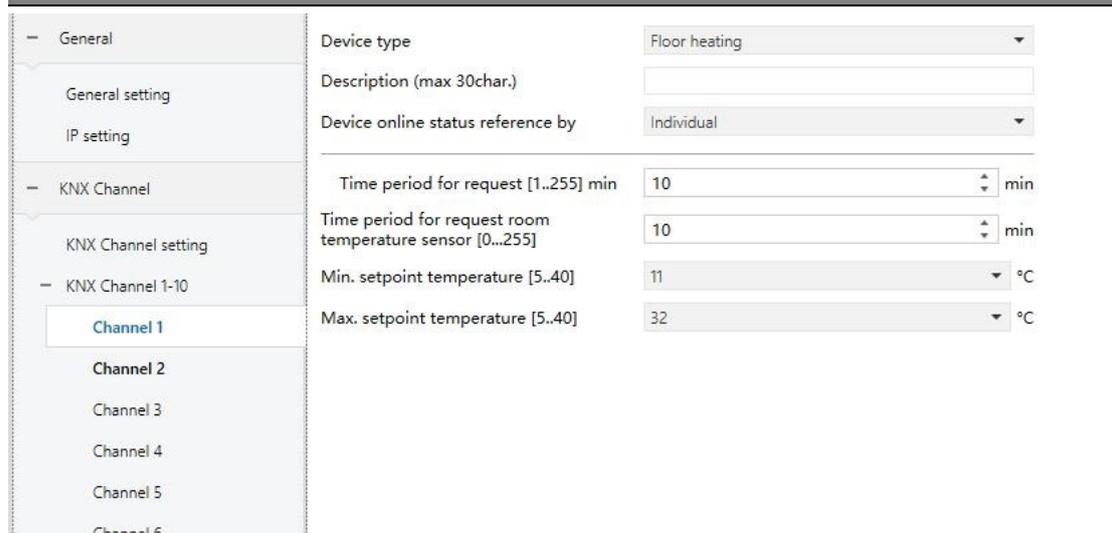
Figure 4.2.2.4 Floor Heating Function Parameters

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

This parameter is used to set the time interval for sending requests to the indoor temperature sensor. Options: 0..255.

A read request is automatically sent by default when the device restarts.

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

These two parameters are used to define the adjustable range for the temperature setpoint limits. If the temperature setpoint exceeds the limit range, the limit value will be used as the output. Options:

5°C

6°C

...

40°C.

For the temperature setpoint, the minimum value must always be less than the maximum value. If this condition is not met, the parameter cannot be configured in ETS.

4.2.2.5. Ventilation System Function Parameters

This section describes the fresh air system function of the KNX channel. The following parameters are visible when "Ventilation system" is selected.

Figure 4.2.2.5 Ventilation System Function Parameters

Parameter "Default fan speed after ventilation on"

This parameter sets the initial fan speed when the fresh air system is turned on. After downloading or resetting, the fresh air system's switch status is read from the bus. If the status cannot be read, it defaults to off. Options:

Low

Medium

High

Last status

Parameter "Data type of fan speed"

This parameter sets the data type for the fresh air system's fan speed. Options:

1bit

1byte

Parameter "Object datatype of 1byte fan speed"

This parameter is visible when the fan speed object type is "1byte" and sets the data type for the 1-byte 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 set the output values for switching to low, medium, and high fan speed levels. A value of 0 indicates the fan speed is off. Options are displayed based on the object type selected in the previous parameter: 0..255 or 0..100.

Status feedback for fan speed

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

These parameters set the status feedback values for low, medium, and high fan speed levels. A value of 0 indicates the fan speed is off. The device will update the fan speed display based on these feedback values. Options are displayed based on the object type selected in the previous parameter: 0..255 / 0..100.

Note: The output values and status feedback values for fan speed must satisfy the condition: low < medium < high. If this condition is not met, the parameter cannot be configured in ETS, and a red warning box will appear, as shown below:

Output value for Fan speed low	68	%
Output value for Fan speed medium	67	%
Output value for Fan speed high	100	%

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

This parameter is visible when the fan speed object type is "1bit" and sets the control values and status feedback values for switching to different fan speeds. Supports four speeds: off, low, medium, and high. The device will update the fan speed display based on the feedback values. 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 the fan speed object type is "1bit" and sets the delay time for fan speed switching. Options: 0..100.

When switching fan speeds, the fan is turned off first, and after the delay time, the new speed is turned on. If the delay time is set to 0, the fan will not turn off first but will directly switch to the next speed.

Parameter "Fan automatic operation is"

This parameter sets whether to enable or disable the automatic operation function for the fresh air system. Options:

Disable

Enable

Note: Currently, the fresh air system device type panel only supports the "Fan automatic operation is" parameter being configured as "Enable".

4.2.2.5. Background Music Function Parameters

This section describes the background music function of the KNX channel, including basic and on/off-enabled background music functions. The following parameters are visible when selecting "Audio control" or "Audio control (with on/off)".

Figure 4.2.2.5 Background Music Function Parameters

Parameter "Object datatype of absolute volume"

This parameter sets the data type for the background music 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 set the control values for each playback mode, including single-cycle, sequential, and random playback. Options: 0..255

Status feedback for play mode

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

These parameters set the status values for each playback mode, including single-cycle, sequential, and random playback. The device will update the playback mode display based on the feedback values. Options: 0..255

4.2.2.6. Sensor Function Parameters

This section describes the sensor functions of the KNX channel, including air quality sensors, presence sensors, motion sensors, brightness sensors, I/O signals, etc.

"Air Quality Sensor" Parameter Settings

"CO2" Parameter Settings

+ General	Device type — PM2.5 sensor ▼
− KNX Channel	Description (max 30char.)
	Device online status reference by — Individual ▼
General setting	Time period for request [1..255] — 10 ⌃⌄ min
KNX Channel setting	Object datatype of PM2.5 — ○ Value in ug/m3(DPT_7.001)
− Channel 1-10	◉ Float value in ug/m3(DPT_9.030)

"PM2.5" Parameter Settings

+ General	Device type — VOC sensor ▼
− KNX Channel	Description (max 30char.)
	Device online status reference by — Individual ▼
General setting	Time period for request [1..255] — 10 ⌃⌄ min
KNX Channel setting	Object datatype of VOC — ○ Value in ug/m3(DPT_7.001)
− Channel 1-10	◉ Float value in ug/m3(DPT_9.030)

"VOC" Parameter Settings

− General	Device type — IR sensor ▼
General setting	Description (max 30char.)
IP setting	Device online status reference by — Individual ▼
− KNX Channel	Time period for request [1..255] min — 10 ⌃⌄ min
KNX Channel setting	

"IR sensor" Parameter Settings

+ General	Device type — Presence sensor ▼
− KNX Channel	Description (max 30char.)
	Device online status reference by — Individual ▼
General setting	Time period for request [1..255] — 10 ⌃⌄ min
KNX Channel setting	Object datatype of brightness(lux) — ○ Value in lux (DPT 7.013)
− Channel 1-10	◉ Float value in lux (DPT 9.004)

"Presence sensor" Parameter Settings

+ General	Device type — Motion sensor ▼
− KNX Channel	Description (max 30char.)
	Device online status reference by — Individual ▼
General setting	Time period for request [1..255] — 10 ⌃⌄ min
KNX Channel setting	

"Motion sensor" Parameter Settings

+ General	Device type — Brightness sensor ▼
− KNX Channel	Description (max 30char.)
	Device online status reference by — Individual ▼
General setting	Time period for request [1..255] — 10 ⌃⌄ min
KNX Channel setting	Object datatype of brightness(lux) — ○ Value in lux (DPT 7.013)
− Channel 1-10	◉ Float value in lux (DPT 9.004)

"Brightness sensor" Parameter Settings

"I/O signal" Parameter Settings

"SOS sensor" Parameter Settings

Figure 4.2.2.6 Sensor Function Parameters

Parameter "Object datatype of PM2.5"

This parameter is visible when "Air quality sensor" or "PM2.5 sensor" is selected. It sets the object data type for PM2.5. Options:

Value in μg/m³ (DPT 7.001)

Float value in μg/m³ (DPT 9.030)

Parameter "Object datatype of PM10"

This parameter is visible when "Air quality sensor" is selected. It sets the object data type for PM10. Options:

Value in μg/m³ (DPT 7.001)

Float value in μg/m³ (DPT 9.030)

Parameter "Object datatype of VOC"

This parameter is visible when "Air quality sensor" or "VOC sensor" is selected. It sets the object data type for VOC. Options:

Value in μg/m³ (DPT 7.001)

Float value in μg/m³ (DPT 9.030)

Parameter "Object datatype of CO2"

This parameter is visible when "Air quality sensor" or "CO2 sensor" is selected. It sets the object data type for 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 "Presence sensor" or "Brightness sensor" is selected. It sets the object data type for brightness. Options:

Value in lux (DPT 7.013)

Float value in lux (DPT 9.004)

4.2.2.7. Current Metering Function Parameters

This section describes the current metering function of the KNX channel. The following parameters are visible when "Current metering" is selected.
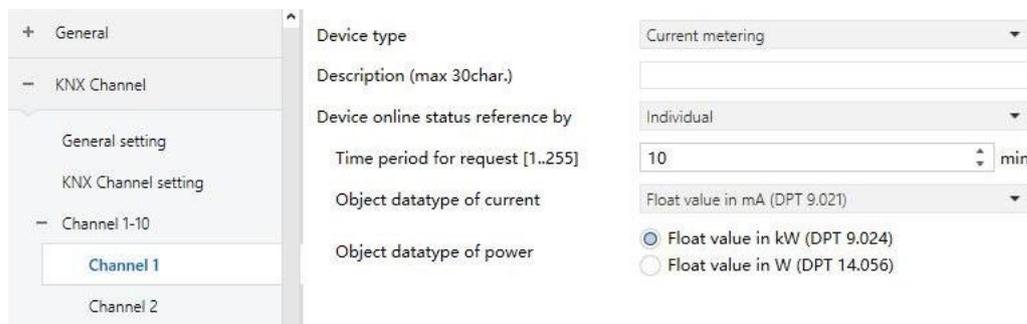
Figure 4.2.2.7 Current Metering Function Parameters

Parameter "Object datatype of current"

This parameter is used to set the object data type for 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 used to set the object data type for power. Options:

Float value in kW (DPT 9.024)

Float value in W (DPT 14.056)

4.2.2.8. Energy Metering Function Parameters

This section describes the energy metering function of the KNX channel. The following parameters are visible when "Energy metering" is selected.

Figure 4.2.2.8 Energy Metering Function Parameters

Parameter "Object datatype of current"

This parameter sets the object data type for 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 sets the object data type for voltage. Options:

Float value in mV (DPT 9.020)

Float value in V (DPT 14.027)

Parameter "Object datatype of power"

This parameter sets the object data type for power. Options:

Float value in kW (DPT 9.024)

Float value in W (DPT 14.056)

Parameter "Object datatype of energy"

This parameter sets the object data type for energy. Options:

Value in Wh (DPT 13.010)

Value in kWh (DPT 13.013)

4.3 Parameter Setting Interface "Logic"

Parameter "1st/2nd/3rd... Logic function"

This parameter is used to set the configuration interface corresponding to the logic. After selection, the corresponding logic function page is displayed. A maximum of 8 logic functions can be enabled.

Parameter "Function of channel"

This parameter sets the logic function for this channel. Options:

  AND

  OR

  XOR

  Gate Forwarding

  Threshold Comparator

  Format Conversion

  Gate Function

  Delay Function

  Staircase Lighting

AND/OR/XOR: The parameters and communication objects are similar, with only the logic algorithms differing. The following description uses the parameters of one option as an example.

4.3.1. "AND/OR/XOR" Function Parameters

Figure 4.3.1 "AND/OR/XOR" Function Parameters

Parameter "Input a/b/c/d/e/f/g//h"

This parameter is used to set whether the logical input x participates in the operation, and whether it participates normally or in inverted form. Options

Disconnected

Normal

Inverted

Disconnected: Not connected, does not participate in the operation.

Normal: The input value directly participates in the operation.

Inverted: The input value is inverted before participating in the operation.

Note: The inversion operation is not applied to the initial value.

Parameter "Default value"

This parameter is used to set the initial value of the logical input x. Options:

0

1

Parameter "Result is inverted"

This parameter sets whether to invert the logical operation result. Options:

No

Yes

No: Directly outputs the result.

Yes: Inverts the result before output.

Parameter "Read input object value after bus voltage recovery"

This parameter sets whether to send a read request to the logical input object after the device powers on or after programming. Options:

No

Yes

Parameter "Output send when"

This parameter sets the condition for sending the logical operation result. Options:

Receiving a new telegram

Every change of output object

Receiving a new telegram: The logical result is sent to the bus each time a new logical input value is received.

Every change of output object: The logical result is sent to the bus only when it changes.

Note: During the first logical operation, the result is sent even if it does not change.

Parameter "Send delay time"

Base:

None

0.1s

1s

...

10s

25s

Factor: 1..255

Sets the delay time for sending the logical operation result to the bus. Delay = Base × Factor. If the Base option is "None," there is no delay.

4.3.2. "Gate Forwarding" Function Parameters

Figure 4.3.2 "Gate Forwarding" Function Parameters

Parameter "Object type of Input/Output"

This parameter sets the data type of the input/output objects. Options:

1bit

4bit

1byte

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

This parameter sets the default initial scene for logical gate forwarding after the device starts up. This scene must be configured in the parameters. Options: 0..64 (0 = inactive).

Note: It is recommended to select a gate scene before operation; otherwise, the initial scene will be enabled by default.

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

This parameter sets the scene number for logical gate forwarding. Each logic supports up to 8 trigger scene configurations.

Options: 0..64 (0 = inactive).

Parameter "Input A/B/C/D send on"

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

    Disable

    Output A

    Output B

    ...

    Output B, C, D

Based on the selected option, one input can be forwarded to one or multiple outputs. The value of the input and the output(s) is the same.

4.3.3. "Threshold Comparator" Function Parameters
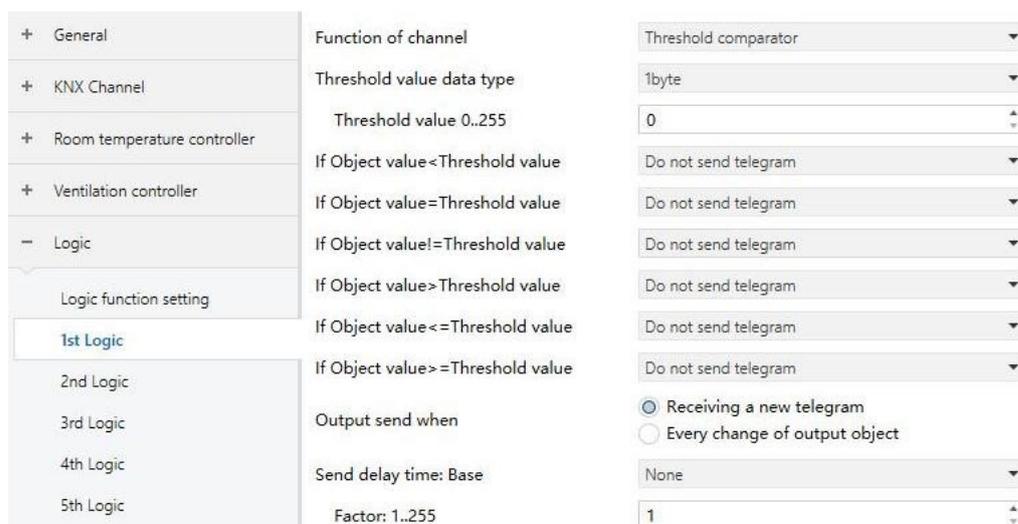
Figure 4.3.3 "Threshold Comparator" Function Parameters

Parameter "Threshold value data type"

This parameter sets the data type for the threshold value. Options:

4bit

1byte

2byte

4byte

Parameter "Threshold value 0..255"

This parameter sets the threshold value. The range of the threshold is determined by 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"

Parameter "If Object value >= Threshold value"

These parameters set the logical result value to be sent when the input object value is less than, equal to, not equal to, greater than, less than or equal to, or greater than or equal to the set threshold value. Options:

Do not send telegram

Send value "0"

Send value "1"

Do not send telegram: The parameter with this option is not considered.

Send value "0"/"1": When the condition is met, the telegram value 0 or 1 is sent.

If there are conflicts in the settings between parameters, the value sent based on the last parameter condition takes precedence. For example: If "If Object value =

Threshold value" is set to "Send value '0'"; and "If Object value <= Threshold value" is set to "Send value '1'"; then when the object value equals the threshold, the logical result will send the value "1".

Parameter "Output send when"

This parameter sets the condition for sending the logical operation result. Options:

Receiving a new telegram

Every change of output object

Receiving a new telegram: The logical result is sent to the bus each time a new logical input value is received.

Every change of output object: The logical result is sent to the bus only when it changes.

Note: During the first logical operation, the result is sent even if it does not change.

Parameter "Send delay time"

Base:

None

0.1s

1s

...

10s

25s

Factor: 1..255

Sets the delay time for sending the logical operation result to the bus. Delay = Base × Factor. If the Base option is "None," there is no delay.

4.3.4. "Format Converter" Function Parameters

Figure 4.3.4 "Format Convert" Function Parameters

Parameter "Function"

This parameter sets the condition for sending the logical operation result. 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 sets the condition for sending the logical operation result. Options:

Receiving a new telegram

Every change of output object

Receiving a new telegram: The logical result is sent to the bus each time a new logical input value is received.

Every change of output object: The logical result is sent to the bus only when it changes.

Note: During the first logical operation, the result is sent even if it does not change.

4.3.5. "Gate function" Function Parameters
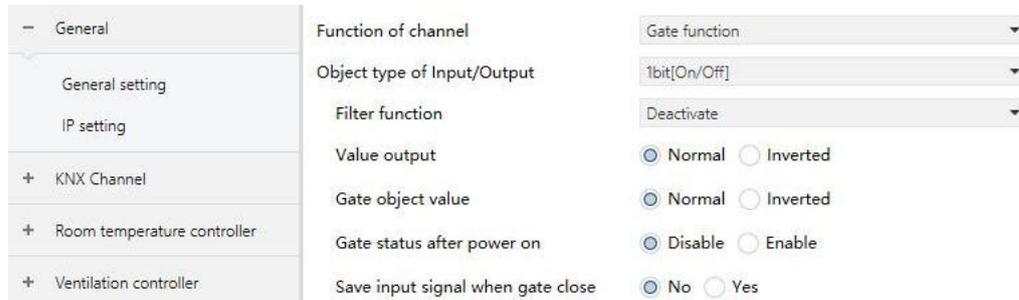
Figure 4.3.5 "Gate function" Function Parameters

Parameter "Object type of Input/Output"

This parameter sets the object type for 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. It configures whether to filter On or Off telegrams, allowing only one type to pass or both to pass. Options:

Deactivate

On filter out

Off filter out

Deactivate: Does not filter On or Off telegrams.

On filter out: Allows Off telegrams to pass, blocks On telegrams.

Off filter out: Allows On telegrams to pass, blocks Off telegrams.

Parameter "Value output"

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

    Normal

    Inverted

Parameter "Gate object value"

This parameter sets whether to invert the gate object value before outputting it. Options:

    Normal

    Inverted

Parameter "Gate status after power on"

This parameter sets the gate status after the device starts up. Options:

    Disable (Off)

    Enable (On)

Parameter "Save input signal when gate close"

This parameter sets whether to save input signals when the gate is closed. Options:

    No

    Yes

No: Input saving is disabled; input values received while the gate is closed are ignored.

Yes: Input saving is enabled; input values received while the gate is closed will be output when the gate opens (regardless of whether the input value has changed).

4.3.6. "Delay function" Function Parameters

Figure 4.3.6 "Delay function" Function Parameters

Parameter "Object type of Input/Output"

This parameter is used to set the object type for 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 sets the delay time for forwarding the value from the input object to the output object after receiving a message. Options: 0..6500 seconds.

Note: If a new input is received during the delay period, the timer will reset and restart.

4.3.7. "Staircase lighting" Function Parameters

Figure 4.3.7 "Staircase lighting" Function Parameters

Parameter "Trigger value"

This parameter sets the telegram value for the object "Trigger value". Options:

0

1

0 or 1

Parameter "Object type of output"

This parameter sets the object type for the output. Options:

1bit

1byte

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

This parameter sets the duration of staircase lighting after the stair light is triggered. Options: 10..6500 seconds.

Parameter "Send value 1 when trigger"

Parameter "Send value 2 after duration time"

These two parameters set the values to be sent. Value 1 is sent upon triggering, and value 2 is sent after the duration time elapses. Options are displayed based on the output object type:

For 1bit:

OFF

ON

For 1byte:

0..255

Parameter "Retriggering"

This parameter sets whether the timer resets and restarts if a trigger value is received again during the delay period. Options:

Disable

Enable

4.4 Parameter Setting Interface "Scene group function"

Figure 4.4.1 "Scene group function" Function Parameters

Figure 4.4.2 "Output X Function" Function Parameters

Parameter "Output X Function" (x: 1–8)

When this parameter is checked, the corresponding parameter functions become visible.

Since the functions of the 8 groups are identical, and the 8 output functions within each group are also the same, the following explanation uses one output from one group as an example for parameter description.

Parameter "Description for Output X function"

This parameter provides a description for the Output X function. A maximum of 30 characters can be entered.

Parameter "Object type of Output X"

Defines the data type for output y in group x. Options:

1bit

1byte

2byte

Parameter "z→Output 1 trigger scene NO. is [1~64, 0 = inactive]" (z: 1–8)

Defines the scene number that triggers output y in group x. Each output can support up to 8 trigger scenes. Options: 0..64, where 0 = inactive.

Parameter "Object value of Output y"

Sets the output value. The value range is determined by the data type of output y: 1bit: 0..1/ 1byte: 0..255/ 2byte: 0..65535

Parameter "Delay time for sending [0..255]"

Sets the delay time for sending the output value to the bus. Options: 0..255.

5. Communication Object Description

Communication objects serve as the medium for devices to communicate with other devices on the bus. Only communication objects can engage in bus communication.

Note: In the attribute column of the tables below: **"C"** indicates that the communication function of the communication object is enabled. **"W"** indicates that the value of the communication object can be rewritten via the bus. **"R"** indicates that the value of the communication object can be read via the bus. **"T"** indicates that the

communication object has transmission functionality. **"U"** indicates that the value of the communication object can be updated.

5.1. "General" Communication Objects

NO	Object Function	Name	Type	Attributes	DPT
1	In operation	General	1bit	C,R,T	1.001 switch
This communication object is used to periodically send the message "1" on the bus to indicate that the device is operating normally. **Note:** This communication object is not utilized in the current version.					

Table 5.1 "General" Communication Object Table

5.2. "KNX Channel" Communication Objects

NO	Object Function	Name	Type	Attributes	DPT
2...	Device online common x, status	KNX General	1bit	C,W,T,U	1.001switch
This communication object is used to query the online status of KNX devices (common device query) and can also receive responses. **0**: Offline **1**: Online It is designed for devices with multiple circuits. For instance, if multiple gateway channels control different circuits of the same KNX device, all channels can share a					

single query request.					
16...	Device online,status	KNX:{{Channel1}}	1bit	C,W,T,U	1.001switch

This communication object is used to send requests for querying the online status of KNX devices (individual device query) to the bus and can also receive feedback.

0: Offline

1: Online

It is suitable for devices controlled exclusively by a single channel of the gateway.

The name in parentheses changes according to the parameter "Description (max 30 char.)". If the parameter description is empty, it defaults to "... Channel x". The same applies below.

Table 5.2 KNX General Communication Objects Table

5.2.1. Basic Function Communication Objects

17	KNX:Channel 1	Switch	1 bit	C - - T -	switch	低
18	KNX:Channel 1	Switch, status	1 bit	C - W T U	switch	低

Switch

17	KNX:Channel 1	Switch	1 bit	C - - T -	switch	低
18	KNX:Channel 1	Brightness dimming	1 byte	C - - T -	percentage (0..100%)	低
19	KNX:Channel 1	Brightness, status	1 byte	C - W T U	percentage (0..100%)	低

Switch/Dimming

17	KNX:Channel 1	Switch	1 bit	C - - T -	switch	低
18	KNX:Channel 1	Switch, status	1 bit	C - W T U	switch	低
19	KNX:Channel 1	RGB dimming value	3 bytes	C - - T -	RGB value 3x(0..255)	低
23	KNX:Channel 1	RGB brightness, status	3 bytes	C - W T U	RGB value 3x(0..255)	低
19	KNX:Channel 1	Red dimming value	1 byte	C - - T -	percentage (0..100%)	低
20	KNX:Channel 1	Green dimming value	1 byte	C - - T -	percentage (0..100%)	低
21	KNX:Channel 1	Blue dimming value	1 byte	C - - T -	percentage (0..100%)	低
23	KNX:Channel 1	Red brightness, status	1 byte	C - W T U	percentage (0..100%)	低
24	KNX:Channel 1	Green brightness, status	1 byte	C - W T U	percentage (0..100%)	低
25	KNX:Channel 1	Blue brightness, status	1 byte	C - W T U	percentage (0..100%)	低

RGB dimming

17	KNX:Channel 1	Switch	1 bit	C - - T -	switch	低
18	KNX:Channel 1	Switch, status	1 bit	C - W T U	switch	低
19	KNX:Channel 1	RGBW dimming value	6 bytes	C - - T -	RGBW value 4x(0..100%)	低
23	KNX:Channel 1	RGBW brightness, status	6 bytes	C - W T U	RGBW value 4x(0..100%)	低
19	KNX:Channel 1	Red dimming value	1 byte	C - - T -	percentage (0..100%)	低
20	KNX:Channel 1	Green dimming value	1 byte	C - - T -	percentage (0..100%)	低
21	KNX:Channel 1	Blue dimming value	1 byte	C - - T -	percentage (0..100%)	低
22	KNX:Channel 1	White dimming value	1 byte	C - - T -	percentage (0..100%)	低
23	KNX:Channel 1	Red brightness, status	1 byte	C - W T U	percentage (0..100%)	低
24	KNX:Channel 1	Green brightness, status	1 byte	C - W T U	percentage (0..100%)	低
25	KNX:Channel 1	Blue brightness, status	1 byte	C - W T U	percentage (0..100%)	低
26	KNX:Channel 1	White brightness, status	1 byte	C - W T U	percentage (0..100%)	低

RGBW dimming

■⇄|17	KNX:Channel 1	Switch	1 bit	C	-	-	T	-	switch	低
■⇄|18	KNX:Channel 1	Switch, status	1 bit	C	-	W	T	U	switch	低
■⇄|19	KNX:Channel 1	Color temperature value	2 bytes	C	-	-	T	-	absolute colour tempera...	低
■⇄|20	KNX:Channel 1	Brightness value	1 byte	C	-	-	T	-	percentage (0..100%)	低
■⇄|23	KNX:Channel 1	Color temperature, status	2 bytes	C	-	W	T	U	absolute colour tempera...	低
■⇄|24	KNX:Channel 1	Brightness, status	1 byte	C	-	W	T	U	percentage (0..100%)	低

Color temperature

■⇄|17	KNX:Channel 1	Open/Close	1 bit	C	-	-	T	-	open/close	低
■⇄|18	KNX:Channel 1	Stop	1 bit	C	-	-	T	-	step	低

Curtain step/move

■⇄|17	KNX:Channel 1	Up/Down	1 bit	C	-	-	T	-	up/down	低
■⇄|18	KNX:Channel 1	Stop	1 bit	C	-	-	T	-	step	低

Roller blind step/move

■⇄|17	KNX:Channel 1	Open/Close	1 bit	C	-	-	T	-	open/close	低
■⇄|18	KNX:Channel 1	Stop	1 bit	C	-	-	T	-	step	低
■⇄|19	KNX:Channel 1	Blind position	1 byte	C	-	-	T	-	percentage (0..100%)	低
■⇄|21	KNX:Channel 1	Blind position, status	1 byte	C	-	W	T	U	percentage (0..100%)	低

Curtain position

	KNX:Channel 1	Device online, status				W			switch	低
⇄|17	KNX:Channel 1	Up/Down	1 bit	C	-	-	T	-	up/down	低
⇄|18	KNX:Channel 1	Stop	1 bit	C	-	-	T	-	step	低
⇄|19	KNX:Channel 1	Blind position	1 byte	C	-	-	T	-	percentage (0..100%)	低
⇄|21	KNX:Channel 1	Blind position, status	1 byte	C	-	W	T	U	percentage (0..100%)	低

Roller blind position

■⇄|17	KNX:Channel 1	Up/Down	1 bit	C	-	-	T	-	open/close	低
■⇄|18	KNX:Channel 1	Stop/Slat adj.	1 bit	C	-	-	T	-	step	低
■⇄|19	KNX:Channel 1	Blind position	1 byte	C	-	-	T	-	percentage (0..100%)	低
■⇄|20	KNX:Channel 1	Slat position	1 byte	C	-	-	T	-	percentage (0..100%)	低
■⇄|21	KNX:Channel 1	Blind position, status	1 byte	C	-	W	T	U	percentage (0..100%)	低
■⇄|22	KNX:Channel 1	Slat position, status	1 byte	C	-	W	T	U	percentage (0..100%)	低

Venetian blind position and slat

■⇄|17	KNX:Channel 1	Recall scene No.	1 byte	C	-	-	T	-	scene number	低
■⇄|18	KNX:Channel 1	Scene No, status	1 byte	C	-	W	T	U	scene number	低

Scene control

■⇄|17	KNX:Channel 1	Send 1bit value	1 bit	C	-	-	T	-	switch	低
■⇄|17	KNX:Channel 1	Send 2bit value	2 bit	C	-	-	T	-	switch control	低
■⇄|17	KNX:Channel 1	Send 1byte percent value	1 byte	C	-	-	T	-	percentage (0..100%)	低
■⇄|17	KNX:Channel 1	Send 1byte unsigned value	1 byte	C	-	-	T	-	counter pulses (0..255)	低
■⇄|17	KNX:Channel 1	Recall scene No.	1 byte	C	-	-	T	-	scene number	低
■⇄|17	KNX:Channel 1	Send 2byte float value	2 bytes	C	-	-	T	-	2-byte float value	低
■⇄|17	KNX:Channel 1	Send 2byte unsigned value	2 bytes	C	-	-	T	-	pulses	低

Value sender

Figure 6.2.1 Basic Function Communication Objects

NO	Object Function	Name	Type	Attributes	DPT
17	Switch	KNX:{{Channel1}}	1bit	C,T	1.001switch
18	Switch, status	KNX:{{Channel1}}	1bit	C,W,T,U	1.001switch
These two communication objects are applicable to switches, dimmers, color, and color temperature control. 0: Light off 1: Light on Obj.311: Used to send on/off messages to the bus to control the light's on/off state. Obj.312: Used to receive on/off status responses from other bus devices, such as the on/off state of dimmers or switches.					
18	Brightness dimming	KNX:{{Channel1}}	1byte	C,T	5.001percentage (0..100%)
19	Brightness, status	KNX:{{Channel1}}	1byte	C,W,T,U	5.001percentage (0..100%)
These two communication objects are applicable to dimmers. Message values: 0…100% Obj.312: Used to send dimming messages to the bus, i.e., transmitting brightness values. Obj.313: Used to receive the brightness status of lights from the dimmer's response.					
19	Red dimming value	KNX:{{Channel1}}	1byte	C,T	5.001percentage (0..100%)
20	Red brightness, status	KNX:{{Channel1}}	1byte	C,W,T,U	5.001percentage (0..100%)
These two communication objects are visible when the RGB object type is set to 3x1byte or the RGBW object type is set to 4x1byte. They are applicable for brightness control of multi-color lights and also support color temperature adjustment. Message values: 0…100% Obj.19: Used to send the brightness value for the R (red) channel to the bus. Obj.20: Used to receive messages on the bus controlling the brightness value for the R (red) channel.					
20	Green dimming value	KNX:{{Channel1}}	1byte	C,T	5.001percentage (0..100%)
24	Green brightness, status	KNX:{{Channel1}}	1byte	C,W,T,U	5.001percentage (0..100%)
These two communication objects are visible when the RGB object type is set to 3x1byte or the RGBW object type is set to 4x1byte. They are applicable for brightness control of multi-color lights and also support color temperature adjustment. Message values: 0…100%					

Obj.19: Used to send the brightness value for the G (green) channel to the bus.

Obj.20: Used to receive messages on the bus controlling the brightness value for the G (green) channel.

21	Blue dimming value	KNX:{{Channel1}}	1byte	C,T	5.001percentage (0..100%)
25	Blue brightness, status	KNX:{{Channel1}}	1byte	C,W,T,U	5.001percentage (0..100%)

These two communication objects are visible when the RGB object type is set to 3x1byte or the RGBW object type is set to 4x1byte. They are applicable for brightness control of multi-color lights and also support color temperature adjustment. Message values: 0…100%

Obj.19: Used to send the brightness value for the B (blue) channel to the bus.

Obj.20: Used to receive messages on the bus controlling the brightness value for the B (blue) channel.

22	White dimming value	KNX:{{Channel1}}	1byte	C,T	5.001percentage (0..100%)
26	White brightness, status	KNX:{{Channel1}}	1byte	C,W,T,U	5.001percentage (0..100%)

These two communication objects are visible when the RGBW object type is set to 4x1byte. They are applicable for brightness control of multi-color lights and also support color temperature adjustment. Message values: 0…100%

Obj.19: Used to send the brightness value for the W (white) channel to the bus.

Obj.20: Used to receive messages on the bus controlling the brightness value for the W (white) channel.

19	RGB dimming value	KNX:{{Channel1}}	3byte	C,T	232.600 RGB value 3x(0..255)
23	RGB brightness, status	KNX:{{Channel1}}	3byte	C,W,T,U	232.600 RGB value 3x(0..255)

These two communication objects are visible when the RGB object type is set to 1x3byte. They are applicable for brightness control of multi-color lights and also support color temperature adjustment.

Obj.19: Used to send the brightness values of the RGB three-color lights to the bus.

Obj.23: Used to receive messages on the bus carrying the brightness values of the RGB three-color lights.

The encoding for the 3-byte RGB dimming object data type is: U8 U8 U8, with details 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

19	RGBW dimming value	KNX:{{Channel1}}	6byte	C,T	251.600 DPT Colour RGBW
23	RGBW brightness, status	KNX:{{Channel1}}	6byte	C,W,T,U	251.600 DPT Colour RGBW

These two communication objects are visible when the RGBW object type is set to 1x6byte. They are suitable for brightness control of multi-color lights and also support color temperature adjustment.

Obj.19: Used to send the brightness values of the RGBW four-color lights to the bus.

Obj.23: Used to receive messages on the bus carrying the brightness values of the RGBW four-color lights.

The encoding for the 6-byte RGBW dimming object data type is: U8 U8 U8 U8 R8 R4 B4, with details as follows:

6_{MSB}	5	4	3	2	1_{LSB}
R	G	B	W	remain	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

19	Color temperature value	KNX:{{Channel1}}	2byte	C,T	7.600 absolute color temperature
23	Color temperature, status	KNX:{{Channel1}}	2byte	C,W,T,U	7.600 absolute color temperature

These two communication objects are applicable for color temperature adjustment of monochrome lights. Message values: 2000..7000K

Obj.19: Used to send color temperature control messages to the bus.

Obj.23: Used to receive color temperature control messages on the bus

20	Brightness value	KNX:{{Channel1}}	1byte	C,T	5.001percentage (0..100%)
24	Brightness, status	KNX:{{Channel1}}	1byte	C,W,T,U	5.001percentage (0..100%)

These two communication objects are applicable for brightness control of monochrome lights. Message values: 0…100%

Obj.20: Used to send dimming messages to the bus, i.e., transmitting brightness values.

Obj.24: Used to receive the brightness status of lights from the dimmer's response.

17	Open/Close	KNX:{{Channel1}}	1bit	C,T	1.009 open/close
18	Stop	KNX:{{Channel1}}	1bit	C,T	1.007 step

Curtain Step/Move: These two communication objects are applicable for open/close curtains. They support open, close, and stop operations.

Obj.311: Used to send messages to the bus to control the opening or closing of the curtain. Message values: 1: Close the curtain. 0: Open the curtain

Obj.18: Used to send messages to the bus to stop curtain movement. Message values: 1: Stop

17	Up/Down	KNX:{{Channel1}}	1bit	C,T	1.008 up/down
18	Stop	KNX:{{Channel1}}	1bit	C,T	1.007 step

Roller Blind Step/Move: These two communication objects are applicable for roller blinds. They support open, close, and stop operations.

Obj.17: Used to send messages to the bus to control the opening or closing of the roller blind. Message values: 1: Move downward to close the blind. 0: Move upward to open the blind

Obj.18: Same as above.

17	Open/Close	KNX:{{Channel1}}	1bit	C,T	1.009 open/close
18	Stop	KNX:{{Channel1}}	1bit	C,T	1.007 step
19	Blind position	KNX:{{Channel1}}	1byte	C,T	5.001percentage (0..100%)
21	Blind position, status	KNX:{{Channel1}}	1byte	C,W,T,U	5.001percentage (0..100%)

Curtain Position: Applicable to open/close curtains. Supports open, close, stop, position adjustment, and position status feedback.

Obj.17: Used to send messages to the bus to control the opening or closing of the curtain. Message values: 1: Close the curtain. 0: Open the curtain

Obj.18: Used to send messages to the bus to stop curtain movement. Message values: 1: Stop

Obj.19: Used to send messages to the bus to control the curtain position. Message values: 0…100%

Obj.21: Used to receive the curtain position status from the curtain actuator on the bus. Message values: 0…100%

17	Up/Down	KNX:{{Channel1}}	1bit	C,T	1.008 up/down

18	Stop	KNX:{{Channel1}}	1bit	C,T	1.007 step
19	Blind position	KNX:{{Channel1}}	1byte	C,T	5.001percentage (0..100%)
21	Blind position, status	KNX:{{Channel1}}	1byte	C,W,T,U	5.001percentage (0..100%)

Roller Blind Position: Applicable to roller blinds (without slats). Supports open, close, stop, position adjustment, and position status feedback.

Obj.17: This communication object is used to send messages to the bus to control the opening or closing of the roller blind. Message values: 1: Move downward to close the blind. 0: Move upward to open the blind

Obj.18, Obj.19, and Obj.21: Same as above.

17	Up/Down	KNX:{{Channel1}}	1bit	C,T	1.008 up/down
18	Stop/Slat adj.	KNX:{{Channel1}}	1bit	C,T	1.007 step
19	Blind position	KNX:{{Channel1}}	1byte	C,T	5.001percentage (0..100%)
20	Slat position	KNX:{{Channel1}}	1byte	C,T	5.001percentage (0..100%)
21	Blind position, status	KNX:{{Channel1}}	1byte	C,W,T,U	5.001percentage (0..100%)
22	Slat position, status	KNX:{{Channel1}}	1byte	C,W,T,U	5.001percentage (0..100%)

Venetian Blind Position and Slat: Applicable to Venetian blinds, with slat angle adjustment. Supports open, close, stop, position and angle adjustment, as well as position and angle status feedback.

Obj.17, Obj.18, and Obj.19: Same as above.

Obj.20: Used to send messages to the bus to stop curtain movement or adjust the slat angle. Message values: **1**: Stop/decrease slat angle. **0**: Stop/increase slat angle

Obj.21: Used to send messages to the bus to control the slat angle position of the Venetian blind. Message values: 0…100%

Obj.22: Used to receive the slat angle position status from the Venetian blind actuator on the bus. Message values: 0…100%

17	Send 1 bit value	KNX:{{Channel1}}	1biton/off	C,T	1.001 switch
	Send 2 bit value		2bit 0..3		2.001 switch control
	Send 1 byte percent value		1byte 0..100%		5.001 percentage (0..100%)
	Send 1 byte unsigned value		1byte 0..255		5.010 counter pulses
	Recall scene No.		1byte 1..64		17.001scene
	Send 2 byte float value		2byte -671088.6		

	Send 2 byte unsigned value			4..670760.96 2byte 0..65535		number 9.x float value 7.001pulses
This communication object is used to send predefined messages to the KNX bus when a command is triggered from the app. The object type and value range are determined by the data type configured in the parameter settings.						
17	Trigger scene NO.	KNX:{{Channel1}}	2 byte	C,W		17.001 scene number
This communication object is used to receive scene trigger messages from the KNX system. Message value: 1..64.						

Table 6.2.1 Basic Function Communication Objects Table

5.2.2 Air Conditioning Function Communication Objects

Air conditioner

Air conditioner (with swing)

Figure 5.2.2 Air Conditioning Function Communication Objects

NO	Object Function	Name	Type	Attributes	DPT
17	Power on/off	KNX:{{Channel1}}	1 bit	C,T	1.001switch
This communication object controls the air conditioner's on/off state via the app and sends the value to the KNX bus. Message value: 1 = On. 0 = Off					
18	Current setpoint adjustment	KNX:{{Channel1}}	2 byte	C,T	9.001temperature
This communication object is used to adjust the set temperature value via the APP and send the corresponding message value to the bus.					
19	Control mode	KNX:{{Channel1}}	1byte	C,T	20.105 HVAC control mode

This communication object is used to send control messages for various air conditioning modes to the bus. Different message values correspond to different operating modes: 0 - Auto, 1 - Heating, 3 - Cooling, 9 - Fan, 14 - Dehumidification, others reserved.					
20	Fan speed	KNX:{{Channel1}}	1byte	C,T	5.001 percentage 5.100 fan stage
This communication object is used to send control messages for different fan speed levels to the bus. The message value is determined by the data type configured in the parameters.					
21	Vanes swing (1-swing,0-stop)	KNX:{{Channel1}}	1 bit	C,T	1.010start/stop
This communication object is visible only when "Air conditioner (with swing)" is selected. It is used to send control messages for left-right fan swing to the bus. Message values: 1: Swing. 0: Stop					
23	Power on/off, status	KNX:{{Channel1}}	1 bit	C,W	1.001switch
This communication object is used to receive the on/off status of the air conditioner from the bus and reflect it for display on the app. Message values: 1: On. 0: Off					
24	Room temperature sensor	KNX:{{Channel1}}	2byte	C,W,T,U	9.001temperature
This communication object is used to receive room temperature data from the bus, periodically send read requests, and reflect the information for display on the app.					
25	Current temperature setpoint, status	KNX:{{Channel1}}	2byte	C,W,U C,W,U	9.001temperature
This communication object is used to receive the current set temperature value from the bus and reflect it for display on the app.					
26	Control mode,status	KNX:{{Channel1}}	1byte	C,W	20.105 HVAC control mode
This communication object is used to receive the current control mode from the bus and reflect it for display on the app. Different message values correspond to different operating modes: 0 - Auto, 1 - Heating, 3 - Cooling, 9 - Fan, 14 - Dehumidification, others reserved.					
27	Fan speed,status	KNX:{{Channel1}}	1byte	C,W	5.001 percentage 5.100 fan stage
This communication object is used to receive the current fan speed from the bus and reflect it for display on the app. The message value is determined by the data type configured in the parameters.					
22	Vanes swing (1-swing,0-stop	KNX:{{Channel1}}	1 bit	C,W	1.010start/stop

), status					

This communication object is visible only when "Air conditioner (with swing)" is selected. It is used to receive the status of fan swing from the bus. Message values: 1: Swing. 0: Stop

Table 5.2.2 Air Conditioning Function Communication Objects Table

5.2.3 Thermostat Function Communication Objects

17	KNX:Channel 1	Heating/Cooling mode	1 bit	C - - T -	cooling/heating	低
30	KNX:Channel 1	Power on/off, status	1 bit	C - W T U	switch	低
31	KNX:Channel 1	Room temperature sensor	2 bytes	C - W T U	temperature (°C)	低
32	KNX:Channel 1	Current temperature setpoint, status	2 bytes	C - W T U	temperature (°C)	低
33	KNX:Channel 1	Heating/Cooling mode, status	1 bit	C - W T U	switch	低
35	KNX:Channel 1	Power on/off	1 bit	C - - T -	switch	低
36	KNX:Channel 1	Current setpoint adjustment	2 bytes	C - - T -	temperature (°C)	低

Room temperature unit

17	KNX:Channel 1	Heating/Cooling mode	1 bit	C - - T -	cooling/heating	低
18	KNX:Channel 1	Operation mode	1 byte	C - - T -	HVAC mode	低
30	KNX:Channel 1	Power on/off, status	1 bit	C - W T U	switch	低
31	KNX:Channel 1	Room temperature sensor	2 bytes	C - W T U	temperature (°C)	低
32	KNX:Channel 1	Current temperature setpoint, status	2 bytes	C - W T U	temperature (°C)	低
33	KNX:Channel 1	Heating/Cooling mode, status	1 bit	C - W T U	switch	低
35	KNX:Channel 1	Power on/off	1 bit	C - - T -	switch	低
36	KNX:Channel 1	Current setpoint adjustment	2 bytes	C - - T -	temperature (°C)	低
38	KNX:Channel 1	Operation mode, status	1 byte	C - W T U	HVAC mode	低

Room temperature unit (with operation mode)

31	KNX:Channel 1	Room temperature sensor	2 bytes	C - W T U	temperature (°C)	低
32	KNX:Channel 1	Current temperature setpoint, status	2 bytes	C - W T U	temperature (°C)	低
33	KNX:Channel 1	Heating/Cooling mode, status	1 bit	C - W T U	switch	低
34	KNX:Channel 1	Fan speed off	1 bit	C - - T -	switch	低
35	KNX:Channel 1	Power on/off	1 bit	C - - T -	switch	低
36	KNX:Channel 1	Current setpoint adjustment	2 bytes	C - - T -	temperature (°C)	低
37	KNX:Channel 1	Fan automatic operation	1 bit	C - - T -	enable	低
38	KNX:Channel 1	Operation mode, status	1 byte	C - W T U	HVAC mode	低
39	KNX:Channel 1	Fan speed, status	1 byte	C - W T U	percentage (0..100%)	低
40	KNX:Channel 1	Fan speed low, status	1 bit	C - W T U	switch	低
41	KNX:Channel 1	Fan speed medium, status	1 bit	C - W T U	switch	低
42	KNX:Channel 1	Fan speed off, status	1 bit	C - W T U	switch	低

Room temperature unit (with operation mode& fan speed)(1)

| 28 | KNX:Channel 1 | Fan speed high, status | 1 bit | C - W T U | switch | 低 |
| 29 | KNX:Channel 1 | Fan automatic operation, status | 1 bit | C - W T U | enable | 低 |

Room temperature unit (with operation mode& fan speed)(2)

Figure 5.2.3 Thermostat Function Communication Objects

NO	Object Function	Name	Type	Attributes	DPT
35	Power on/off	KNX:{{Channel1}}	1 bit	C,T	1.001switch
This communication object is only visible when the "Power on/off is" parameter is configured as enabled. It is used to control the on/off status of the thermostat on the KNX bus via the APP and to send the corresponding message to the bus. Message values: 1: On. 0: Off					
36	Current setpoint	KNX:{{Channel1}}	2byte	C,T	9.001temperature

	adjustment				
This communication object is used to adjust the set temperature value via the APP and send the corresponding message value to the bus.					
17	Heating/Cooling mode	KNX:{{Channel1}}	1 bit	C,T	1.100cooling/ heating
This communication object is used to switch between heating/cooling via the APP and send the corresponding message to the bus. Message values: 1: Heating. 0: Cooling					
18	Operation mode	KNX:{{Channel1}}	1byte	C,T	20.102 HVAC mode
This communication object is used to send messages for room operation modes to the bus. Different message values correspond to different operating modes: 1 - Comfort, 2 - Standby, 3 - Energy Saving, 4 - Protection, others reserved.					
23	Fan speed	KNX:{{Channel1}}	1byte	C,T	5.001 percentage 5.100 fan stage
25	Fan speed low	KNX:{{Channel1}}	1 bit	C,T	1.001switch
26	Fan speed medium	KNX:{{Channel1}}	1 bit	C,T	1.001switch
27	Fan speed high	KNX:{{Channel1}}	1 bit	C,T	1.001switch
34	Fan speed off	KNX:{{Channel1}}	1 bit	C,T	1.001switch
These communication objects are used to control fan speed via the APP and send the corresponding messages to the bus. For 1-bit objects, the display is based on parameter settings: Object 25 – Low speed Object 26 – Medium speed Object 27 – High speed Object 34 – Fan speed off. When switching to a specific fan speed, only the corresponding object sends message "1". If the 1-bit off object is disabled, switching to the "off" speed will cause all objects to send message "0". If the 1-bit off object is enabled, switching to the "off" speed will cause only the off-speed object to send message "1" (suitable for integration with other manufacturers' fan actuators). For 1-byte objects: The message value for each fan speed level is determined by the data type configured in the parameters. When a specific fan speed is activated via the APP, object 319 sends the corresponding message value for that speed to the bus.					
37	Fan automatic operation	KNX:{{Channel1}}	1 bit	C,T	1.003 enable
This communication object is used to send automatic fan speed control messages to the bus. Message values: 1 – Automatic. 0 – Exit automatic					
30	Power	KNX:{{Channel1}}	1 bit	C,W	1.001switch

	on/off,status				

This communication object is only visible when the "Power on/off is" parameter is configured as enabled. It is used to receive the on/off status of the thermostat from the bus and reflect it for display on the APP. Message values: 1: On. 0: Off

31	Room temperature sensor	KNX:{{Channel1}}	2byte	C,W,T,U	9.001temperature

This communication object is used to receive the room temperature from the bus, periodically send read requests, and reflect the data for display on the APP.

32	Current temperature setpoint, status	KNX:{{Channel1}}	2byte	C,W,T,U	9.001temperature

This communication object is used to receive the current set temperature value from the bus and reflect it for display on the APP.

33	Heating/ Cooling mode,status	KNX:{{Channel1}}	1bit	C,W	1.100cooling/ heating

This communication object is used to receive the heating and cooling status from the bus and reflect it for display on the APP. Message values: 1 – Heating. 0 – Cooling

38	Operation mode, status	KNX:{{Channel1}}	1byte	C,W	20.102 HVAC mode

This communication object is used to receive messages for room operation modes from the bus. Different message values correspond to different operating modes: 1 - Comfort, 2 - Standby, 3 - Energy Saving, 4 - Protection, others reserved.

39	Fan speed, status	KNX:{{Channel1}}	1byte	C,W	5.001 percentage 5.100 fan stage
40	Fan speed low, status	KNX:{{Channel1}}	1bit	C,W	1.001switch
41	Fan speed medium, status	KNX:{{Channel1}}	1bit	C,W	1.001switch
28	Fan speed high, status	KNX:{{Channel1}}	1bit	C,W	1.001switch
42	Fan speed off, status	KNX:{{Channel1}}	1bit	C,W	1.001switch

Fan speed status feedback can be received via four 1-bit objects or one 1-byte object.
For 1-bit objects:
Object 40 – Low speed
Object 41 – Medium speed
Object 28 – High speed
Object 42 – Fan speed off

When an object receives message "1", the corresponding speed is activated and updated for display on the APP.

If the 1-bit off object is disabled, fan speed is considered off when the low/medium/high speed messages all have a value of "0".

If the 1-bit off object is enabled, fan speed is considered off when the off-speed object receives "1", or when the low/medium/high speed messages all have a value of "0".

For 1-byte objects: The fan speed status value is determined by the data type configured in the parameters. When object 334 receives the specified value, the fan speed status is updated for display on the APP.

29	Fan automatic operation, status	KNX:{{Channel1}}	1bit	C,W	1.003enable
This object is used to receive status feedback for automatic fan speed control. Message values: 1 – Automatic. 0 – Exit automatic					

5.2.4 Floor Heating Function Communication Objects

30	KNX:Channel 1	Power on/off, status	1 bit	C - W T U	switch	低
31	KNX:Channel 1	Room temperature sensor	2 bytes	C - W T U	temperature (°C)	低
32	KNX:Channel 1	Current temperature setpoint, status	2 bytes	C - W T U	temperature (°C)	低
35	KNX:Channel 1	Power on/off	1 bit	C - - T -	switch	低
36	KNX:Channel 1	Current setpoint adjustment	2 bytes	C - - T -	temperature (°C)	低

NO	Object Function	Name	Type	Attributes	DPT
35	Power on/off	KNX:{{Channel1}}	1 bit	C,T	1.001switch
This communication object is used to control the on/off status of the floor heating on the KNX bus via the APP and to send the corresponding message value to the bus. Message values: 1 – On. 0 – Off					
30	Power on/off,status	KNX:{{Channel1}}	1 bit	C,W	1.001switch
This communication object is used to receive the on/off status of the floor heating from the bus and reflect it for display on the APP. Message values: 1 – On. 0 – Off					
31	Room temperature sensor	KNX:{{Channel1}}	2byte	C,W,T,U	9.001temperature
This communication object is used to receive room temperature data from the bus, periodically send read requests, and reflect the information for display on the APP.					
32	Current temperature setpoint,status	KNX:{{Channel1}}	2byte	C,W,U	9.001temperature
This communication object is used to receive the current set temperature value from the bus and reflect it for display on the APP.					
36	Current setpoint adjustment	KNX:{{Channel1}}	2byte	C,T	9.001temperature

This communication object is used to adjust the set temperature value via the APP and send the corresponding message value to the bus.

5.2.5 Ventilation Function Communication Objects

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

Figure 5.2.5 Ventilation Function Communication Objects

NO	Object Function	Name	Type	Attributes	DPT
35	Power on/off	KNX:{{Channel1}}	1 bit	C,T	1.001switch
This communication object is used to control the on/off status of the fresh air system on the KNX bus via the APP and to send the corresponding message value to the bus. Message values: 1 – On. 0 – Off					
23	Fan speed	KNX:{{Channel1}}	1byte	C,T	5.001 percentage 5.100 fan stage
25	Fan speed low	KNX:{{Channel1}}	1 bit	C,T	1.001switch
26	Fan speed medium	KNX:{{Channel1}}	1 bit	C,T	1.001switch
27	Fan speed high	KNX:{{Channel1}}	1 bit	C,T	1.001switch
These communication objects are used to send control messages for fan speed to the bus. For 1-byte objects: The message value for each fan speed level is determined by the data type configured in the parameters. When a specific fan speed is activated via the APP, object 312 sends the corresponding message value for that speed to the bus. For 1-bit objects: Object 25 – Low speed					

Object 26 – Medium speed

Object 27 – High speed

When a specific fan speed is activated via the APP, the corresponding object sends message "1" to the bus, while the fan-off message sends "0".

37	Fan automatic operation	KNX:{{Channel1}}	1bit	C,T	1.003enable

This communication object is only visible when the "Fan automatic operation is" parameter is configured as enabled. It is used to send automatic fan speed control messages to the bus. Message values: 1 – Automatic. 0 – Exit automatic

317	Heat recovery	KNX:{{Channel1}}	1bit	C,T	1.003enable

This communication object is used to control the heat exchange mode via the APP and send the corresponding message to the bus. Message values: 1 – Activated. 0 – Not activated

42	Power on/off,status	KNX:{{Channel1}}	1 bit	C,W	1.001switch

This communication object is used to receive the on/off status of the fresh air system from the bus and reflect it for display on the APP. Message values: 1 – On. 0 – Off

40	Fan speed, status	KNX:{{Channel1}}	1byte	C,W	5.001 percentage 5.100 fan stage
39	Fan speed low, status	KNX:{{Channel1}}	1bit	C,W	1.001switch
41	Fan speed medium, status	KNX:{{Channel1}}	1bit	C,W	1.001switch
28	Fan speed high, status	KNX:{{Channel1}}	1bit	C,W	1.001switch

Fan speed status feedback can be received via three 1-bit objects or one 1-byte object.

For 1-byte objects: The fan speed status value is determined by the data type configured in the parameters. When object 319 receives the specified value, the fan speed status is updated for display on the APP.

For 1-bit objects:

Object 39 – Low speed

Object 41 – Medium speed

Object 28 – High speed

When an object receives message "1", the corresponding speed is activated and

updated for display on the APP. When the fan speed is off, all speed objects must have a message value of "0".

29	Fan automatic operation, status	KNX:{{Channel1}}	1bit	C,W	1.003enable

This communication object is only visible when the "Fan automatic operation is" parameter is configured as enabled. It is used to receive status feedback for automatic fan speed control. Message values: 1 – Automatic. 0 – Exit automatic

324	Heat recovery, status	KNX:{{Channel1}}	1bit	C,W	1.003enable

This communication object is used to receive the heat exchange mode status from the bus and reflect it for display on the APP. Message values: 1 – Activated. 0 – Not activated

325	Filter alarm, status	KNX:{{Channel1}}	1bit	C,W	1.005 alarm

This communication object is used to receive the filter status from the bus and reflect it for display on the APP. Message values: 1 – Normal operation. 0 – Replacement alert

Table 5.2.4 Fresh Air System Function Communication Objects Table

5.2.6 Background Music Function Communication Objects

Audio control

Audio control (with on/off)

Figure 5.2.6 Background Music Function Communication Objects

NO	Object Function	Name	Type	Attributes	DPT
35	Power on/off	KNX:{{Channel1}}	1 bit	C,T	1.001switch

This communication object applies only to background music functions with switch

control. It is used to control the on/off status of background music on the KNX bus via the APP and to send the corresponding message value to the bus.Message values: 1 – On. 0 – Off

19	Play=1/Pause=0	KNX:{{Channel1}}	1 bit	C,T	1.010start/stop
This communication object is used to control the playback/stop of music in the background music module via the APP. Message values: 1 – Play music. 0 – Stop playing music					
20	Next track=1 /Previous track=0	KNX:{{Channel1}}	1 bit	C,T	1.007step
This communication object is used to switch between songs in the background music module via the APP, enabling playback of the previous or next song. Message values: 1 – Play next song. 0 – Play previous song					
21	Volume+=1/Volume-=0	KNX:{{Channel1}}	1 bit	C,T	1.007step
17	Absolute volume	KNX:{{Channel1}}	1 byte	C,T	5.001percentage 5.004percentage
This communication object is used to adjust the volume of the background music module via the APP. The message value is determined by the different object data types. For 1-bit objects, message values: 1 – Volume increase 0 – Volume decrease For 1-byte objects, message values depend on the object type: 0..100 / 0..255					
22	Mute	KNX:{{Channel1}}	1 bit	C,T	1.003enable
This communication object applies only to background music functions without switch control, enabling mute control of background music via the APP. Message values: 1 – Mute. 0 – Unmute					
23	Play mode	KNX:{{Channel1}}	1 byte	C,T	5.010counterpulses
This communication object is used to send control messages for background music playback modes, with message values for different modes preset by parameters.					
30	Power on/off,status	KNX:{{Channel1}}	1 bit	C,W	1.001switch
This communication object applies only to background music functions with switch control. It is used to receive the on/off control status feedback of background music from the bus.					
24	Play=1/Pause=0, status	KNX:{{Channel1}}	1 bit	C,W C,W,T,U	1.010start/stop
This communication object is used to receive feedback on the music playback/stop					

status from the bus.

For background music functions with switch control, this object has the attributes: C, W.

For background music functions without switch control, this object has the attributes: C, W, T, U.

18	Absolute volume, status	KNX:{{Channel1}}	1 byte	C,W,T,U	5.001percentage 5.004percentage

For volume adjustment functions, the 1-byte object applies only to background music functions without switch control. It is used to receive the volume status of the background music module. The message value depends on the object type: 0..100 or 0..255.

25	Mute status	KNX:{{Channel1}}	1bit	C,W,T,U	1.003enable

This communication object applies only to background music functions without switch control. It is used to receive the mute status of the background music module.

26	Play mode,status	KNX:{{Channel1}}	1 byte	C,W / C,W,T,U	5.010counterpulses

This communication object is used to receive feedback on the background music playback mode status from the bus, with message values for different modes preset by parameters.

For background music functions with switch control, this object has the attributes: C, W.

For background music functions without switch control, this object has the attributes: C, W, T, U.

27	Music source	KNX:{{Channel1}}	1 byte	C,T	5.010counterpulses

This communication object is used to send control messages for the audio source of background music, with message values for different audio sources preset by parameters.

28	Music source,status	KNX:{{Channel1}}	1 byte	C,W,T,U	5.010counterpulses

This communication object is used to receive feedback on the background music audio source status from the bus, with message values for different audio sources preset by parameters.

Table 5.2.6 Background Music Function Communication Objects Table

5.2.7 Sensor Function Communication Objects

17	KNX:Channel 1	PM2.5 value	2 bytes	C	-	W	T	U	pulses	低
18	KNX:Channel 1	PM10 value	2 bytes	C	-	W	T	U	pulses	低
19	KNX:Channel 1	VOC value	2 bytes	C	-	W	T	U	pulses	低
20	KNX:Channel 1	CO2 value	2 bytes	C	-	W	T	U	pulses	低
21	KNX:Channel 1	AQI value	2 bytes	C	-	W	T	U	pulses	低

	17	KNX:Channel 1	PM2.5 value	2 bytes	C - W T U	concentration (µg/m³)	低
	18	KNX:Channel 1	PM10 value	2 bytes	C - W T U	concentration (µg/m³)	低
	19	KNX:Channel 1	VOC value	2 bytes	C - W T U	concentration (µg/m³)	低
	20	KNX:Channel 1	CO2 value	2 bytes	C - W T U	parts/million (ppm)	低
	21	KNX:Channel 1	AQI value	2 bytes	C - W T U	pulses	低
	24	KNX:Channel 1	Humidity value	2 bytes	C - W T U	humidity (%)	低
	25	KNX:Channel 1	Temperature value	2 bytes	C - W T U	temperature (°C)	低

Air quality sensor

	17	KNX:Channel 1	CO2 value	2 bytes	C - W T U	pulses	低
	17	KNX:Channel 1	CO2 value	2 bytes	C - W T U	parts/million (ppm)	低
	24	KNX:Channel 1	Humidity value	2 bytes	C - W T U	humidity (%)	低
	25	KNX:Channel 1	Temperature value	2 bytes	C - W T U	temperature (°C)	低

CO2sensor

	17	KNX:Channel 1	PM2.5 value	2 bytes	C - W T U	pulses	低
	17	KNX:Channel 1	PM2.5 value	2 bytes	C - W T U	concentration (µg/m³)	低
	24	KNX:Channel 1	Humidity value	2 bytes	C - W T U	humidity (%)	低
	25	KNX:Channel 1	Temperature value	2 bytes	C - W T U	temperature (°C)	低

PM2.5 sensor

	17	KNX:Channel 1	VOC value	2 bytes	C - W T U	concentration (µg/m³)	低
	17	KNX:Channel 1	VOC value	2 bytes	C - W T U	concentration (µg/m³)	低
	24	KNX:Channel 1	Humidity value	2 bytes	C - W T U	humidity (%)	低
	25	KNX:Channel 1	Temperature value	2 bytes	C - W T U	temperature (°C)	低

VOC sensor

| | 17 | KNX:Channel 1 | IR detector | 1 bit | C - W T U | occupancy | 低 |

IR sensor

	17	KNX:Channel 1	Brightness value(lux)	2 bytes	C - W T U	brightness (lux)	低
	18	KNX:Channel 1	Presence detector	1 bit	C - W T U	occupancy	低
	17	KNX:Channel 1	Brightness value(lux)	2 bytes	C - W T U	lux (Lux)	低

Presence sensor

| | 17 | KNX:Channel 1 | Motion detector | 1 bit | C - W T U | occupancy | 低 |

Motion sensor

| | 17 | KNX:Channel 1 | Brightness value(lux) | 2 bytes | C - W T U | brightness (lux) | 低 |
| | 17 | KNX:Channel 1 | Brightness value(lux) | 2 bytes | C - W T U | lux (Lux) | 低 |

Brightness sensor

| | 17 | KNX:Channel 1 | I/O signal value | 1 bit | C - W T U | switch | 低 |

I/O signal

| | 17 | KNX:Channel 1 | Alarm value | 1 bit | C - W T U | alarm | 低 |

SOS sensor

Figure 5.2.7 Sensor Function Communication Objects

NO	Object	Name	Type	Attributes	DPT

	Function				
17	PM2.5 value	KNX:{{Channel1}}	2byte	C,W,T,U	7.001pulse 9.030 concentration(ug/m3)
	This communication object is used to receive PM2.5 value inputs, retrieve the corresponding values from the bus, and update the display on the APP. The unit is µg/m³. Range: 0...999 µg/m³, and the object's data type is configured by parameters.				
18	PM10 value	KNX:{{Channel1}}	2byte	C,W,T,U	7.001pulse 9.030 concentration(ug/m3)
	This communication object is used to receive PM10 value inputs, retrieve the corresponding values from the bus, and update the display on the APP. The unit is µg/m³. Range: 0...999 µg/m³, and the object's data type is configured by parameters.				
19	VOC value	KNX:{{Channel1}}	2byte	C,W,T,U	7.001pulse 9.030 concentration(ug/m3)
	This communication object is used to receive VOC value inputs, retrieve the corresponding values from the bus, and update the display on the APP. The unit is µg/m³. Range: 0...999 µg/m³, and the object's data type is configured by parameters.				
21	CO_2 value	KNX:{{Channel1}}	2byte	C,W,T,U	7.001pulse 9.008 Parts/million(ppm)
	This communication object is used to receive CO_2 value inputs, retrieve the corresponding values from the bus, and update the display on the APP. The unit is ppm. Range: 0...4000 ppm.				
17	IR detector	KNX:{{Channel1}}	1 bit	C,W,T,U	1.018 occupancy
	This communication object retrieves the IR request time from the bus. Range: 0...255.				
21	AQI value	KNX:{{Channel1}}	2byte	C,W,T,U	7.001pulses
	This communication object is used to receive AQI value inputs, retrieve the corresponding values from the bus, and update the display on the APP. Range: 0...500.				
24	Temperature value	KNX:{{Channel1}}	2byte	C,W,T,U	9.001temperature
	This communication object is used to receive temperature measurement values from temperature sensors on the bus. Range: -40°C...40°C.				
25	Humidity value	KNX:{{Channel1}}	2byte	C,W,T,U	9.001 humidity
	This communication object is used to receive humidity measurement values from humidity sensors on the bus. Range: 0...100%.				
17	Brightness value(lux)	KNX:{{Channel1}}	2byte	C,W,T,U	7.013brightness(lux) 9.004lux(lux)
	This communication object is used to receive illuminance value inputs, retrieve the corresponding values from the bus, and update the display on the APP. The unit is lux. Range: 0...65535 lux, and the object's data type is determined by parameters.				

17	Motion detector	KNX:{{Channel1}}	1 bit	C,W,T,U	1.018Occupancy
This communication object is used to receive 1-bit presence signal inputs. Message values: 0 – No presence. 1 – Presence detected					
17	I/O signal value	KNX:{{Channel1}}	1 bit	C,W,T,U	1.001switch
This communication object is used to receive 1-bit switch signal inputs. Message values: 0 – Open (disconnected). 1 – Closed (connected)					
17	Alarm value	KNX:{{Channel1}}	1 bit	C,W,T,U	1.005alarm
This communication object retrieves the SOS request time from the bus. Range: 0...255.					

Table 5.2.7 Sensor Function Communication Objects Table

5.2.8 Current/Energy Metering Function Communication Objects

Current metering

Energy metering

Figure 5.2.8 Current/Energy Metering Function Communication Objects

NO	Object Function	Name	Type	Attributes	DPT
17	Current value	KNX:{{Channel1}}	2byte 4byte	C,W,T,U	7.012current(mA) 9.021current(mA) 14.019electric current(A)
This communication object is used to receive current values from the bus and update them for display on the APP. The data type of the object is determined by the parameters.					

If a 2-byte integer type is selected: Display range: 0...65535 mA, resolution: 1 mA.
If a 2-byte floating-point type is selected: Display range: -670760...670760 mA, resolution: 0.01 mA.

If a 4-byte type is selected: Display range: -99999999.9...99999999.9 A, resolution: 0.1 A.					
20	Voltage value	KNX:{{Channel1}}	2byte 4byte	C,W,T,U	9.020 voltage(mV) 14.027electric potential(V)
This communication object is used to receive voltage values from the bus and update them for display on the APP. The data type of the object is determined by the parameters. If a 2-byte type is selected: Display range: -670760 mV...670760 mV, resolution: 0.01 mV. If a 4-byte type is selected: Display range: -99999999.9...99999999.9 V, resolution: 0.1 V.					
18	Power value	KNX:{{Channel1}}	2byte 4byte	C,W,T,U	9.024 power(kW) 14.056power(W)
This communication object is used to receive power values from the bus and update them for display on the APP. The data type of the object is determined by the parameters. If a 2-byte type is selected: Display range: -670760...670760 kW, resolution: 0.01 kW. If a 4-byte type is selected: Display range: -99999999.9...99999999.9 W, resolution: 0.1 W.					
19	Energy value	KNX:{{Channel1}}	4byte	C,W,T,U	13.010 active energy(mA) 13.013active energy(A)
This communication object is used to receive energy consumption values from the bus and update them for display on the APP. The data type of the object is determined by the parameters. Display range: -2147483648...2147483647 Wh, resolution: 1 Wh. Display range: -2147483648...2147483647 kWh, resolution: 1 kWh.					

Table 5.2.8 Current/Energy Metering Function Communication Objects Table

5.3. "Logic" Communication Objects

5.3.1 "AND/OR/XOR" Communication Objects

6480	1st Logic	Input a	1 bit	C - W T U	boolean	低
6481	1st Logic	Input b	1 bit	C - W T U	boolean	低
6482	1st Logic	Input c	1 bit	C - W T U	boolean	低
6483	1st Logic	Input d	1 bit	C - W T U	boolean	低
6484	1st Logic	Input e	1 bit	C - W T U	boolean	低
6485	1st Logic	Input f	1 bit	C - W T U	boolean	低
6486	1st Logic	Input g	1 bit	C - W T U	boolean	低
6487	1st Logic	Input h	1 bit	C - W T U	boolean	低
6488	1st Logic	Logic result	1 bit	C - - T -	boolean	低

NO	Object Function	Name	Type	Attributes	DPT
6/...	Input X	1st/.../8thLogic	1bit	C,W,T,U	1.002boolean
This communication object is used to receive the value of logical input Input x.					
6488	Logic result	1st/.../8thLogic	1bit	C,T	1.002boolean
This communication object is used to send the result of a logical operation.					

Table 5.3.1 "AND/OR/XOR" Communication Objects Table

5.3.2 "Gate Forwarding" Communication Objects

6480	1st Logic	Gate value select	1 byte	C - W - -	scene number	低
6481	1st Logic	Input A	1 bit	C - W - -	switch	低
6482	1st Logic	Input B	1 bit	C - W - -	switch	低
6483	1st Logic	Input C	1 bit	C - W - -	switch	低
6484	1st Logic	Input D	1 bit	C - W - -	switch	低
6485	1st Logic	Output A	1 bit	C - - T -	switch	低
6486	1st Logic	Output B	1 bit	C - - T -	switch	低
6487	1st Logic	Output C	1 bit	C - - T -	switch	低
6488	1st Logic	Output D	1 bit	C - - T -	switch	低

Figure 5.3.2 Gate Forwarding Communication Objects

NO	Object Function	Name	Type	Attributes	DPT
6480	Gate value select	1st/.../8thLogic	1byte	C,W	17.001 scene number
This communication object is used to select the scene for logical gate forwarding.					
6/.../6	Input X	1st/.../8thLogic	1bit 4bit 1byte	C,W	1.001switch 3.007dimming control 5.010counter pulses(0..255)
This communication object is used to receive the value of logical gate input Input x.					
7/.../10	Output X	1st/.../8thLogic	1bit 4bit	C,T	1.001switch 3.007dimming

			1byte		control 5.010counter pulses(0..255)
This communication object is used to output the value after logical gate forwarding. The output value is the same as the input value, but a single input can be forwarded to one or multiple outputs based on parameter configuration.					

Table 5.3.2 "Gate Forwarding" Communication Objects Table

5.3.3 "Threshold Comparator" Communication Objects

Figure 5.3.3 "Threshold Comparator" Communication Objects

NO	Object Function	Name	Type	Attributes	DPT
6480	Threshold value input	1st/.../8thLogic	4bit 1byte 2byte 4byte	C,W,U	3.007 dimming control 5.010 counter pulses(0..255) 7.001 pulses 12.001 counter pulses
This communication object is used to input threshold values.					
6488	Logic result	1st/.../8thLogic	1bit	C,T	1.002boolean
This communication object is used to send the result of the logical operation, i.e., the value to be sent after comparing the object input threshold with the parameter-set threshold.					

Table 5.3.3 "Threshold Comparator" Communication Objects Table

5.3.4 "Format Convert" Communication Objects

"2x1bit → 1x2bit" function: Converts two 1-bit values into one 2-bit value. For example, Input bit1 = 1, bit0 = 0 → Output 2-bit = 2.

6480	1st Logic	Input 1bit-bit0	1 bit	C	-	W	-	U	boolean	低
6481	1st Logic	Input 1bit-bit1	1 bit	C	-	W	-	U	boolean	低
6482	1st Logic	Input 1bit-bit2	1 bit	C	-	W	-	U	boolean	低
6483	1st Logic	Input 1bit-bit3	1 bit	C	-	W	-	U	boolean	低
6484	1st Logic	Input 1bit-bit4	1 bit	C	-	W	-	U	boolean	低
6485	1st Logic	Input 1bit-bit5	1 bit	C	-	W	-	U	boolean	低
6486	1st Logic	Input 1bit-bit6	1 bit	C	-	W	-	U	boolean	低
6487	1st Logic	Input 1bit-bit7	1 bit	C	-	W	-	U	boolean	低
6488	1st Logic	Output 1byte	1 byte	C	-	-	T	-	counter pulses (0..255)	低

"8x1bit → 1x1byte" function: Converts eight 1-bit values into one 1-byte value. For example, Input bit2 = 1, bit1 = 1, bit0 = 1, with all other bits as 0 → Output 1-byte = 7.

| 6480 | 1st Logic | Input 1byte | 1 byte | C | - | W | - | U | counter pulses (0..255) | 低 |
| 6488 | 1st Logic | Output 2byte | 2 bytes | C | - | - | T | - | pulses | 低 |

"1x1byte → 1x2byte" function: Converts a 1-byte value into a 2-byte value. For example, Input 1-byte = 125 → Output 2-byte = 125. Although the value remains unchanged, the data type of the value has been altered.

6480	1st Logic	Input 1byte-low	1 byte	C	-	W	-	U	counter pulses (0..255)	低
6481	1st Logic	Input 1byte-high	1 byte	C	-	W	-	U	counter pulses (0..255)	低
6488	1st Logic	Output 2byte	2 bytes	C	-	-	T	-	pulses	低

"2x1byte → 1x2byte" function: Converts two 1-byte values into one 2-byte value. For example, Input 1-byte-low = 255 ($FF), Input 1-byte-high = 100 ($64) → Output 2-byte = 25855 ($64FF).

6480	1st Logic	Input 2byte-low	2 bytes	C	-	W	-	U	pulses	低
6481	1st Logic	Input 2byte-high	2 bytes	C	-	W	-	U	pulses	低
6488	1st Logic	Output 4byte	4 bytes	C	-	-	T	-	counter pulses (unsigned)	低

"2x2byte → 1x4byte" function: Converts two 2-byte values into one 4-byte value. For example, Input 2-byte-low = 65530 ($FFFA), Input 2-byte-high = 32768 ($8000) → Output 4-byte = 2147549178 ($8000FFFA).

6480	1st Logic	Input 1byte	1 byte	C	-	W	-	U	counter pulses (0..255)	低
6481	1st Logic	Output 1bit-bit0	1 bit	C	-	-	T	-	boolean	低
6482	1st Logic	Output 1bit-bit1	1 bit	C	-	-	T	-	boolean	低
6483	1st Logic	Output 1bit-bit2	1 bit	C	-	-	T	-	boolean	低
6484	1st Logic	Output 1bit-bit3	1 bit	C	-	-	T	-	boolean	低
6485	1st Logic	Output 1bit-bit4	1 bit	C	-	-	T	-	boolean	低
6486	1st Logic	Output 1bit-bit5	1 bit	C	-	-	T	-	boolean	低
6487	1st Logic	Output 1bit-bit6	1 bit	C	-	-	T	-	boolean	低
6488	1st Logic	Output 1bit-bit7	1 bit	C	-	-	T	-	boolean	低

"1x1byte → 8x1bit" function: Converts one 1-byte value into eight 1-bit values. For example, Input 1-byte = 200 → Output bit0 = 0, bit1 = 0, bit2 = 0, bit3 = 1, bit4 = 0, bit5 = 0, bit6 = 1, bit7 = 1.

6482	1st Logic	Output 1bit-bit1	1 bit	C - - T -	boolean	低
6483	1st Logic	Output 1bit-bit2	1 bit	C - - T -	boolean	低
6484	1st Logic	Output 1bit-bit3	1 bit	C - - T -	boolean	低
6485	1st Logic	Output 1bit-bit4	1 bit	C - - T -	boolean	低

"1x2byte → 2x1byte" function: Converts one 2-byte value into two 1-byte values. For example, Input 2-byte = 55500 ($D8CC) → Output 1-byte-low = 204 ($CC), Output 1-byte-high = 216 ($D8).

6480	1st Logic	Input 4byte	4 bytes C - W - U	counter pulses (unsigned)	低
6481	1st Logic	Output 2byte-low	2 bytes C - - T -	pulses	低
6488	1st Logic	Output 2byte-high	2 bytes C - - T -	pulses	低

"1x4byte → 2x2byte" function: Converts one 4-byte value into two 2-byte values. For example, Input 4-byte = 78009500 ($04A6549C) → Output 2-byte-low = 21660 ($549C), Output 2-byte-high = 1190 ($04A6).

6480	1st Logic	Input 3byte	3 bytes C - W - U	RGB value 3x(0..255)	低
6481	1st Logic	Output 1byte-low	1 byte C - - T -	counter pulses (0..255)	低
6482	1st Logic	Output 1byte-middle	1 byte C - - T -	counter pulses (0..255)	低
6488	1st Logic	Output 1byte-high	1 byte C - - T -	counter pulses (0..255)	低

"1x3byte → 3x1byte" function: Converts one 3-byte value into three 1-byte values. For example, Input 3-byte = $7864C8 → Output 1-byte-low = 200 ($C8), Output 1-byte-middle = 100 ($64), Output 1-byte-high = 120 ($78).

6480	1st Logic	Input 1byte-low	1 byte C - W - U	counter pulses (0..255)	低
6481	1st Logic	Input 1byte-middle	1 byte C - W - U	counter pulses (0..255)	低
6482	1st Logic	Input 1byte-high	1 byte C - W - U	counter pulses (0..255)	低
6488	1st Logic	Output 3byte	3 bytes C - - T -	RGB value 3x(0..255)	低

"3x1byte → 1x3byte" function: Converts three 1-byte values into one 3-byte value. For example, Input 1-byte-low = 150 ($96), Input 1-byte-middle = 100 ($64), Input 1-byte-high = 50 ($32) → Output 3-byte = $326496.

Figure 5.3.4 "Format Convert" Communication Objects

NO	Object Function	Name	Type	Attributes	DPT
6480 6481	Input ...	1st/.../8thLogic	1bit 1byte 2byte 3byte 4byte	C,W,U	1.001switch 5.010counter pulses(0..255) 7.001pulses 232.600RGB value3x(0..255) 12.001counter pulses
This communication object is used to input the value to be converted.					
6480 ...	Output...	1st/.../8thLogic	1bit 2bit	C,T	1.001switch 2.001switch control

6488			1byte 2byte 3byte 4byte	5.010counter pulses(0..255) 7.001pulses 232.600RGB value3x(0..255) 12.001counter pulses
This communication object is used to output the converted value.				

<div align="center">Table 5.3.4 "Format Convert" Communication Objects Table</div>

5.3.5 "Gate Function" Communication Objects

	6480	1st Logic	Input	1 bit	C - W - -	switch	低
	6481	1st Logic	Gate input	1 bit	C - W - -	boolean	低
	6488	1st Logic	Output	1 bit	C - - T -	switch	低

<div align="center">Input/Output-1bit[On/Off]</div>

	6480	1st Logic	Input	1 byte	C - W - -	percentage (0..100%)	低
	6481	1st Logic	Gate input	1 bit	C - W - -	boolean	低
	6488	1st Logic	Output	1 byte	C - - T -	percentage (0..100%)	低

<div align="center">Input/Output-1byte[0..100%]</div>

	6480	1st Logic	Input	1 byte	C - W - -	counter pulses (0..255)	低
	6481	1st Logic	Gate input	1 bit	C - W - -	boolean	低
	6488	1st Logic	Output	1 byte	C - - T -	counter pulses (0..255)	低

<div align="center">Input/Output-1byte[0..255]</div>

	6480	1st Logic	Input	2 bytes	C - W - -	temperature (°C)	低
	6481	1st Logic	Gate input	1 bit	C - W - -	boolean	低
	6488	1st Logic	Output	2 bytes	C - - T -	temperature (°C)	低

<div align="center">Input/Output-2byte[Float]</div>

	6480	1st Logic	Input	2 bytes	C - W - -	pulses	低
	6481	1st Logic	Gate input	1 bit	C - W - -	boolean	低
	6488	1st Logic	Output	2 bytes	C - - T -	pulses	低

<div align="center">Input/Output-2byte[0..65535]</div>

<div align="center">Figure 5.3.5 "Gate Function" Communication Objects</div>

NO	Object Function	Name	Type	Attributes	DPT
6480	Input	1st/.../8thLogic	1bit 1byte 2byte	C,W	1.001switch 5.001percentage 5.010counterpulses 9.001temperature 7.001pulses
This communication object is used to input the values that require gate filtering.					
6481	Gate input	1st/.../8thLogic	1bit	C,W	1.002boolean
This communication object is used to control the on/off state of the gate input. When					

the gate is open, input signals are allowed to pass through and will be output, and any changes will also send the current input state. When the gate is closed, signals cannot pass through.

6488	Output	1st/.../8thLogic	1bit 1byte 2byte	C,T	1.001switch 5.001percentage 5.010counterpulses 9.001temperature 7.001pulses

This communication object is used to output the value after gate filtering. Output is only available when the gate input state is "open," as defined by the object "Gate input.

Table 5.3.5 "Gate Function" Communication Objects Table

5.3.6 "Delay Function" Communication Objects

| ⬛⇄ 6480 | 1st Logic | Input | | 1 bit | C | - | W | - | - | switch | 低 |
| ⬛⇄ 6488 | 1st Logic | Output | | 1 bit | C | - | - | T | - | switch | 低 |

Input/Output-1bit[On/Off]

| ⬛⇄ 6480 | 1st Logic | Input | | 1 byte | C | - | W | - | - | percentage (0..100%) | 低 |
| ⬛⇄ 6488 | 1st Logic | Output | | 1 byte | C | - | - | T | - | percentage (0..100%) | 低 |

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

| ⬛⇄ 6480 | 1st Logic | Input | | 1 byte | C | - | W | - | - | counter pulses (0..255) | 低 |
| ⬛⇄ 6488 | 1st Logic | Output | | 1 byte | C | - | - | T | - | counter pulses (0..255) | 低 |

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

| ⬛⇄ 6480 | 1st Logic | Input | | 2 bytes | C | - | W | - | - | temperature (°C) | 低 |
| ⬛⇄ 6488 | 1st Logic | Output | | 2 bytes | C | - | - | T | - | temperature (°C) | 低 |

Input/Output-2byte[Float]

| ⬛⇄ 6480 | 1st Logic | Input | | 2 bytes | C | - | W | - | - | pulses | 低 |
| ⬛⇄ 6488 | 1st Logic | Output | | 2 bytes | C | - | - | T | - | pulses | 低 |

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

Figure 5.3.6 "Delay Function" Communication Objects

NO	Object Function	Name	Type	Attributes	DPT
6480	Input	1st/.../8thLogic	1bit 1byte 2byte	C,W	1.001switch 5.001percentage 5.010counterpulses 9.001temperature 7.001pulses

This communication object is used to receive values from the bus that require delay.

6488	Output	1st/.../8thLogic	1bit 1byte 2byte	C,T	1.001switch 5.001percentage 5.010counterpulses 9.001temperature 7.001pulses
This communication object is used to send values that require delayed forwarding, with the delay time defined by the parameters.					

Table 5.3.6 "Delay Function" Communication Objects Table

5.3.7 "Staircase lighting" Communication Objects

Figure 5.3.7 "Staircase Lighting" Communication Objects

NO	Object Function	Name	Type	Attributes	DPT
6480	Trigger value	1st/.../8thLogic	1bit	C,W	1.017trigger
This communication object is used to receive the value from the bus that triggers the staircase light to turn on.					
6481	Light-on duration time	1st/.../8thLogic	2byte	C,W	7.005time(s)
This communication object is used to modify the duration of the staircase light. The modification range refers to the limits defined in the parameters, and values exceeding this range will be capped at the limit.					
6488	Output	1st/.../8thLogic	1bit 1byte	C,T	1.001switch 5.010counterpulses
This communication object is used to output value 1 when triggered and output value 2 after the delay has elapsed. The message value is determined by the data type configured in the parameters.					

Table 5.3.7 "Staircase Lighting" Communication Objects Table

5.4. "Scene group function" Communication Objects

6552	Scene Group	Main scene trigger	1 byte	C	-	W	-	-	scene number	低
6553	1st Scene Group-Output 1	1bit value	1 bit	C	-	-	T	-	switch	低
6554	1st Scene Group-Output 2	1byte unsigned value	1 byte	C	-	-	T	-	counter pulses (0..255)	低
6555	1st Scene Group-Output 3	2byte unsigned value	2 bytes	C	-	-	T	-	pulses	低
6556	1st Scene Group-Output 4	RGB value	3 bytes	C	-	-	T	-	RGB value 3x(0..255)	低
6557	1st Scene Group-Output 5	RGBW value	6 bytes	C	-	-	T	-	RGBW value 4x(0..100%)	低
6558	1st Scene Group-Output 6	1bit value	1 bit	C	-	-	T	-	switch	低
6559	1st Scene Group-Output 7	1bit value	1 bit	C	-	-	T	-	switch	低
6560	1st Scene Group-Output 8	1bit value	1 bit	C	-	-	T	-	switch	低

Figure 5.4.1 "Scene Group Function" Communication Objects

NO	Object Function	Name	Type	Attributes	DPT
6552	Main scene trigger	Scene Group	1bit	C,W	17.001scene number
This communication object triggers the events in the group by calling a scene number, causing each output in the group to send a specific value to the bus. Message range: 0..63.					
6553...	1bit value 1byte unsigned value 2byte unsigned value RGB value RGBW value	1st/.../8th Scene Group-Output	1bit 1byte 2bytes 3bytes 6bytes	C,T	1.001switch 5.010counter pulses 7.001pulses 232.600RGB value 3x(0..255) 251.600RGB value 4x(0..255)
When a specific scene is triggered, this communication object is used to send the corresponding output value for that scene to the bus. If the output is not configured for this scene, no message will be sent.					

Table 5.4.1 "Scene Group Function" Communication Objects Table

6. Letour Smart Life APP Interaction Guide

This section provides a detailed explanation of the interactive usage of the Tuya Smart APP, organized by functional modules. Before operating the APP, please ensure that the database has been downloaded to the KNX-Tuya ZigBee Smart Gateway. Only after the database is downloaded can the gateway and other devices be added in the APP.

6.1 Adding Devices

Click the desktop icon ZKPRO on your phone to launch the Tuya Smart APP, then register or log in to your Tuya Smart account. After successful login, go to the APP homepage and tap the icon ＋ in the upper-right corner to add a device.

6.1.1 Steps to Add the KNX Fusion Gateway-LG1 Device

(1) Select the corresponding device category. For this gateway, choose the multi-mode gateway (BLE+Zigbee) under the gateway/controller category. You can either manually add the device (steps 2–5) or tap the icon to automatically discover the device. Ensure the APP is in pairing mode (turn on Bluetooth and Wi-Fi). After discovering the device, tap to add it, then enter the Wi-Fi network name and password for the device to connect to. Tap the blue text in the image below to jump to the "Auto Discovery" or search-and-pair features, which also explain device reset methods.

If automatic device discovery fails, you can retry or choose to manually add the device.
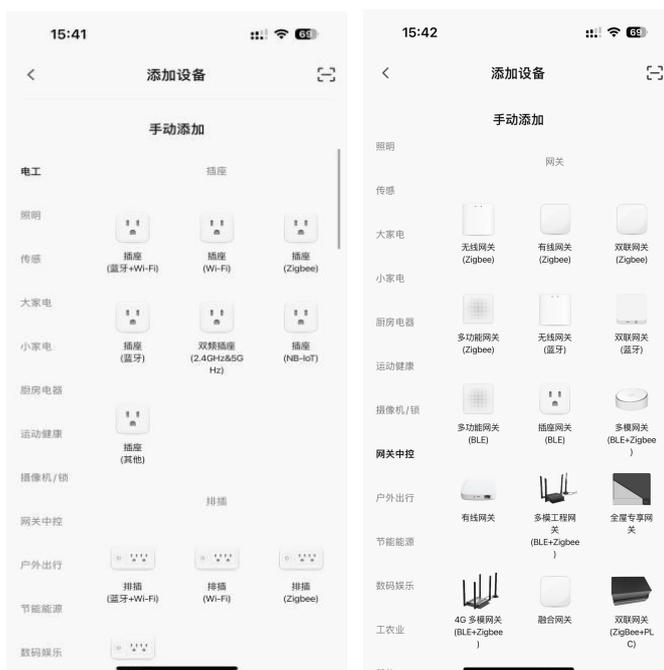
Note: Please ensure that Bluetooth, Wi-Fi, and location services are enabled on your mobile phone. Otherwise, the APP will not be able to detect the device and will display the following prompt:

(2) Click the device icon for the Multi-mode Gateway (BLE+Zigbee) to view the device connection steps. If the device has not been reset, follow the guided steps to pair it with the network. If it has already been reset, skip step 3. Please note the status of the indicator lights on the gateway.

(3) Reset the device. Press and hold the SFR button on the gateway for 5 seconds to remove the gateway from the cloud. After manually triggering the device reset, the NET and STA LED lights will stay solidly lit. When the STA LED light starts flashing, the gateway enters the ready-to-connect state. The gateway indicator lights behave as follows:

POW LED light:

Stays solidly lit when the gateway power supply is normal.

NET light:

Turns off when the gateway successfully connects to the network.

Connection Status LED (green light):

Stays solidly lit during gateway initialization;

Flashes when the gateway is in the ready-to-connect state;

Turns off when the gateway connects to the cloud.

(4) Confirm that the device reset was successful and that the phone and gateway are on the same network. Then verify that the relevant indicator lights on the device are showing the correct status. For this gateway, ensure that the NET LED light is solidly lit and the STA LED light is flashing, then click "Next" to proceed to the device connection state.
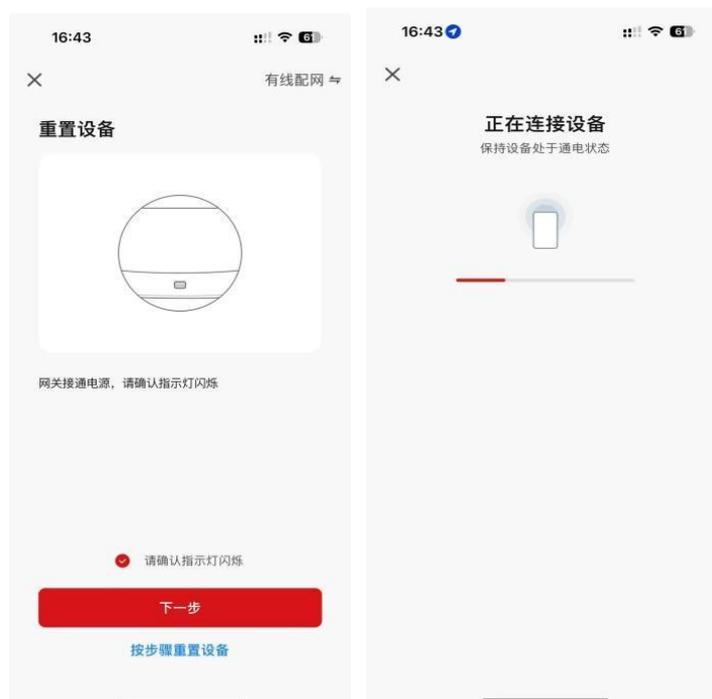
Note: The KNX Fusion Gateway-LG1 supports three network configuration methods: wired pairing, Bluetooth pairing, and hotspot pairing. Click the wired pairing option in the upper right corner to switch the gateway device's network configuration method.
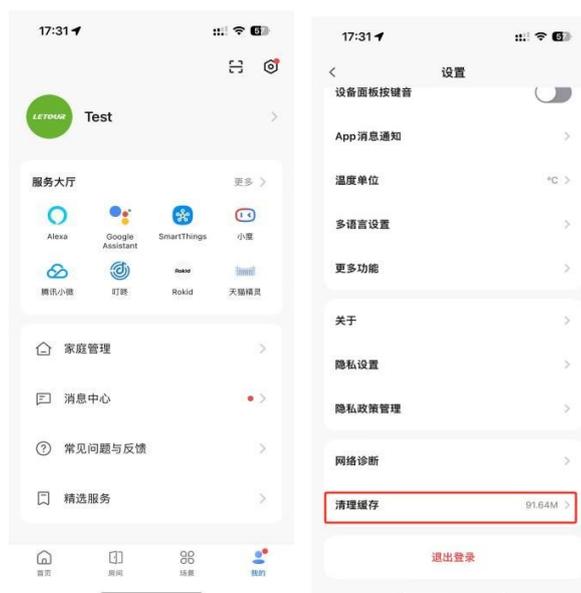
(5) Adding the device.

① For wired pairing or Bluetooth pairing, the "KNX Fusion Gateway-LG1" will be automatically detected and added once scanned. Wait for the device to be successfully added, then click "Done."
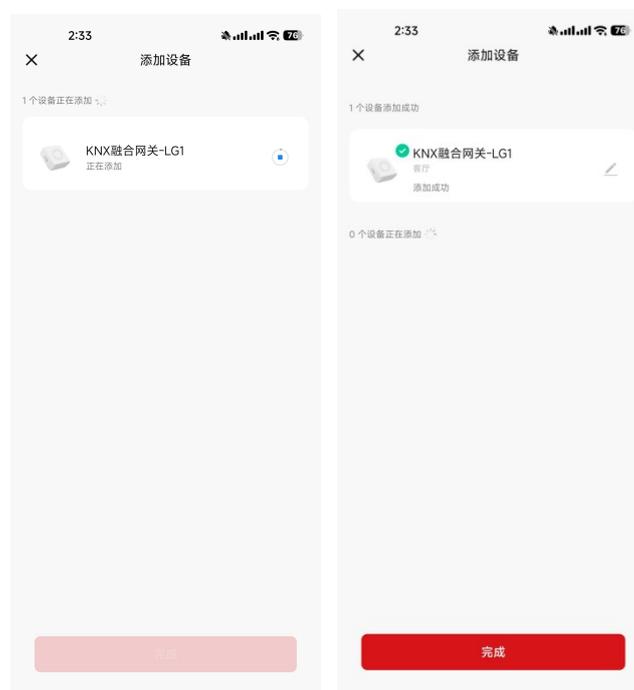
② For hotspot pairing, first select the 2.4GHz Wi-Fi network, enter the password, and then follow the prompts to connect to the device's hotspot. Wait for the device to be successfully added, then click "Done."
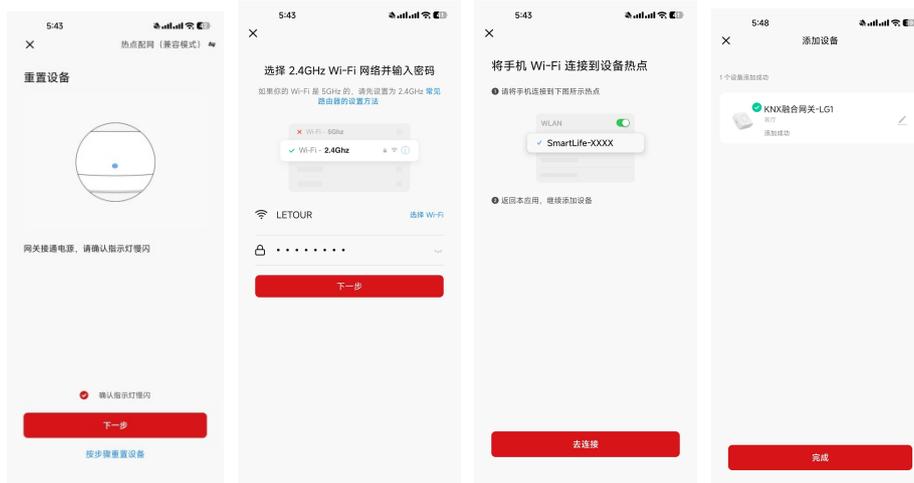
Note: For Android systems, if the gateway repeatedly fails to connect, you need to clear the cache. Go to the **[Me]** interface, tap the icon in the upper right corner to enter the settings interface, then select **[Clear Cache]**.


(6) After completing the gateway addition following the steps above, open the device interface of the KNX Fusion Gateway-LG1. The interface will display the Bluetooth device list, ZigBee device list, and KNX device list, showing the sub-devices of various protocols currently added to the gateway.

6.1.2 Adding KNX Channel Devices

In the gateway device interface, click "KNX Devices" to switch to the KNX device list. Then, tap "Add Device" at the bottom to automatically add the KNX channel devices that have already been configured in the gateway database.
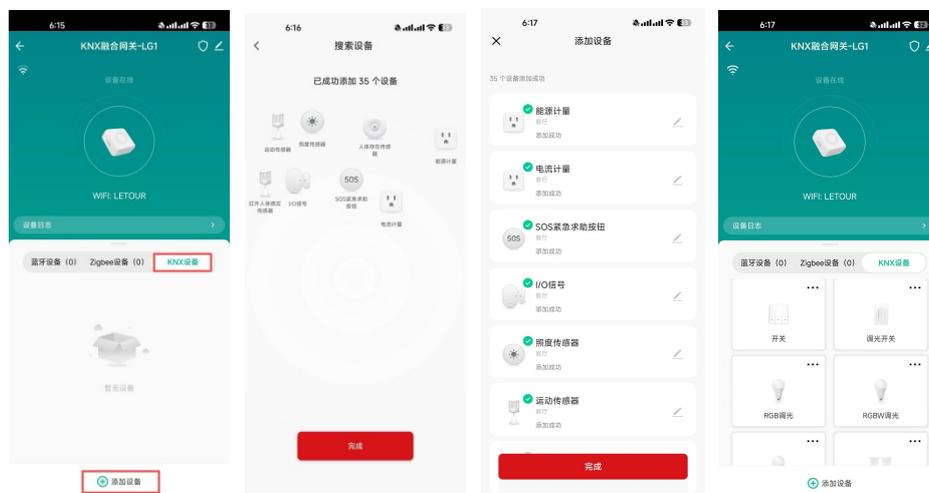
The following sections will explain the interactive usage of KNX channel devices one by one.

6.1.3 Adding Zigbee Devices

Under the gateway device interface, click "Zigbee Devices" to switch to the Zigbee device list, then tap "Add Device" at the bottom to automatically add Zigbee devices that are in pairing mode near the gateway.
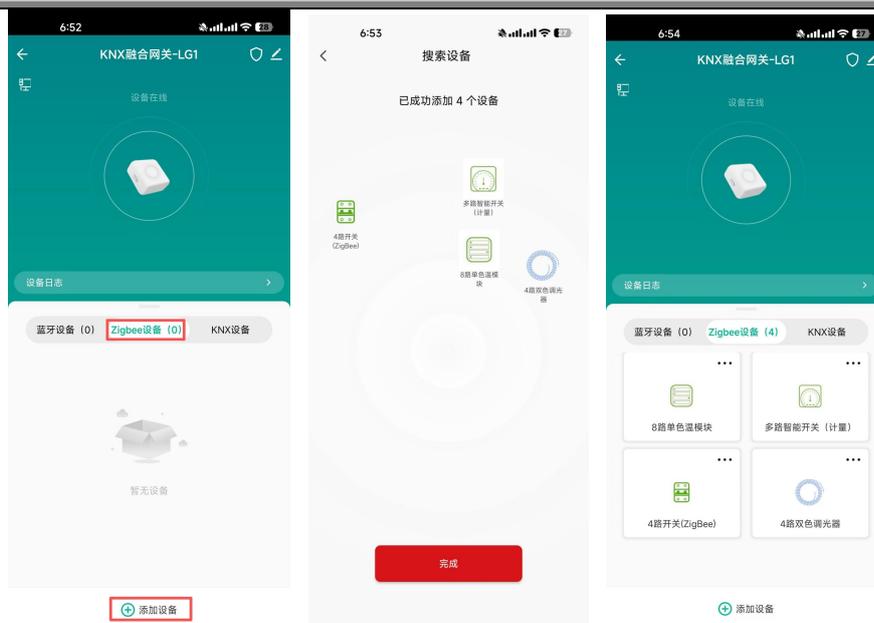
6.1.4 Adding Bluetooth Devices

The process of adding Bluetooth devices to the gateway differs from that of KNX and ZigBee devices. First, add the Bluetooth device to the APP using the mobile phone's Bluetooth functionality. Then, under the gateway device interface, click "Bluetooth Devices" to switch to the Bluetooth device list. Next, tap "Add Device" at the bottom and select "Link Bluetooth Device" from the options bar that appears. By binding the Bluetooth device to the gateway, remote control of the Bluetooth device can be achieved through the gateway.
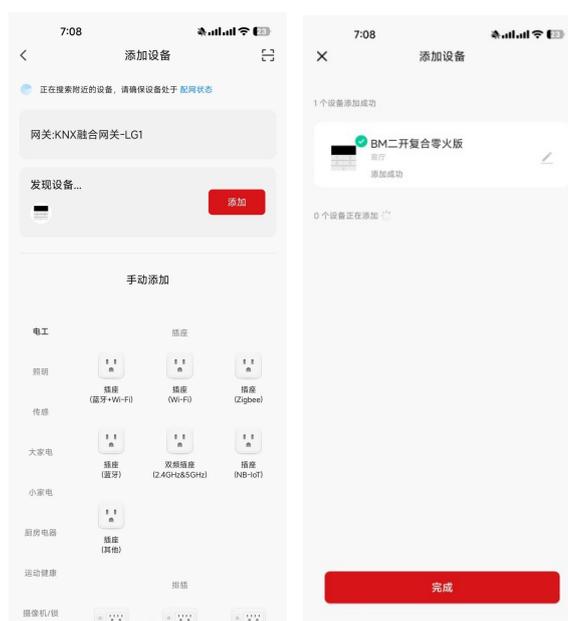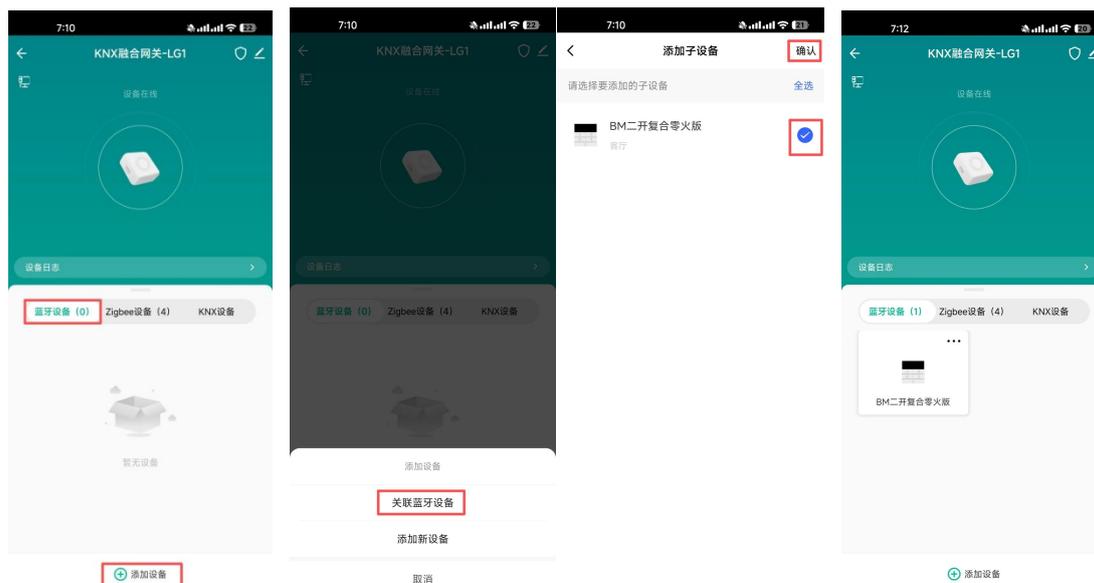
6.2 Gateway Information

In the APP, click the icon to enter the gateway settings interface, then select Device Information to view the gateway details, as shown in the figure below. Note that the IP address will have its last digit hidden.
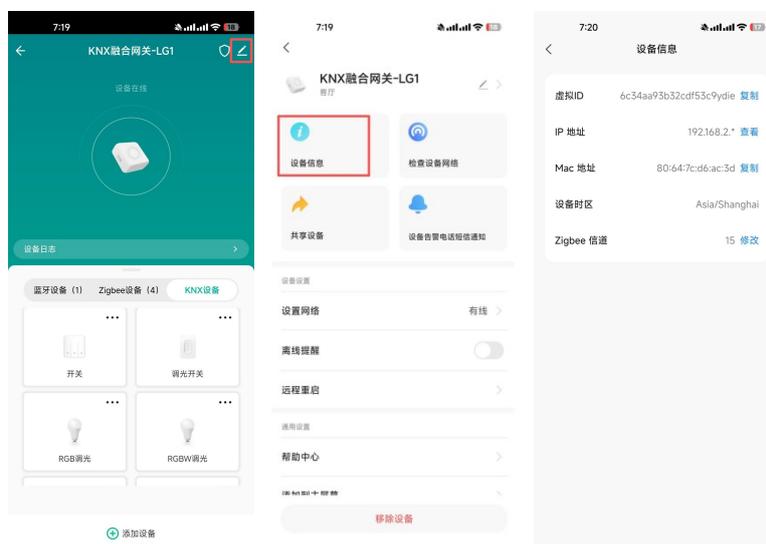
6.3 Switch Interface
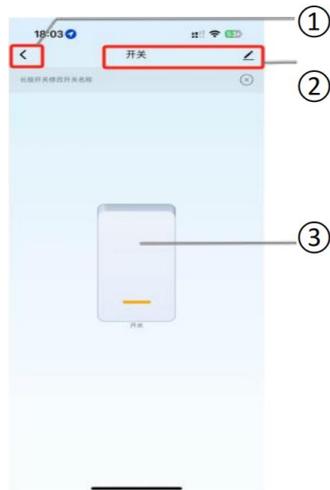
Figure 6.3(1) Switch Interface   **Figure 6.3(2)** Basic Device Settings   **Figure 6.3(3)** Information Editing Interface

① Return to the previous level interface.

② Device name, displayed at the top of the page. Click the icon ∠ to jump to Figure 6.3(2), where you can view, modify device settings, or remove the device. The basic device settings are shown in Figure 6.3(2), including device information, "One-Touch Execution" and "Automation," other settings, and removal settings. "One-Touch Execution" can be configured via KNX scene switches. For details, refer to Section 6.8.

Click ④ to jump to Figure 6.3(3), where you can edit the current device name, device icon, and device location.

③ Switch button: gray indicates the off state, bright color indicates the on state. Used for touch control to toggle the switch status.

6.4 Dimming Interface

Figure 6.4(1) Dimming Interface

① Return to the previous level interface.

② Click the icon ⋯ to modify the device name, change device settings, or remove the device.

③ Light switch button: gray indicates the off state, bright color indicates the on state. When the current brightness value is not 0, tapping this button will turn off the brightness, setting the brightness value to 0.

④ B: Brightness value adjustment. Adjust the brightness by sliding the slider, range: 0..100%.

6.5 RGB Dimming Interface

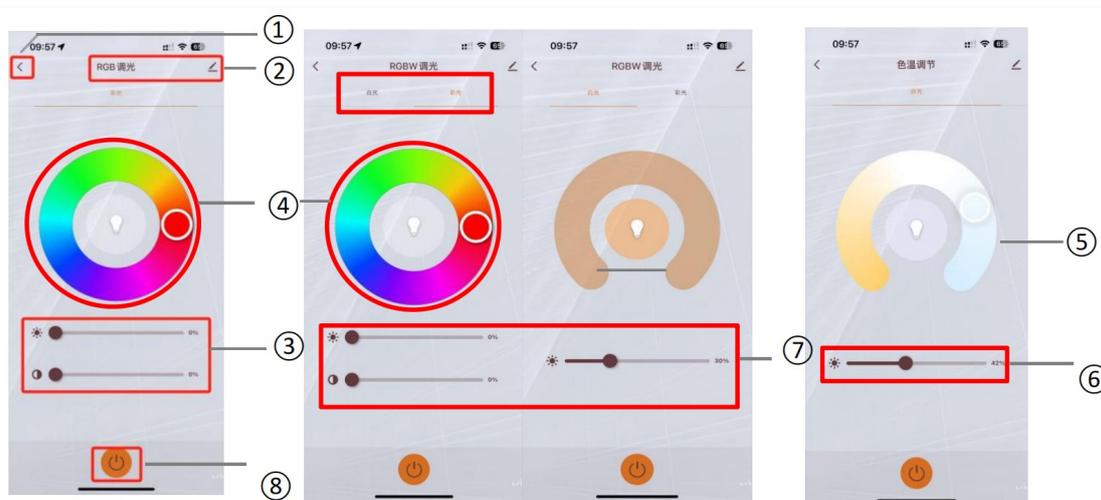Figure 6.5(1) RGB Dimming Interface   Figure 6.5(2) RGBW Dimming Interface Figure   6.5(3) Color Temperature Dimming Interface

① Return to the previous level interface.

② Device name, displayed at the top of the page. Click the icon ∠ to view, modify device settings, or remove the device.

RGB dimming functions are divided into three control types: RGB, RGBW, and Color Temperature. Among these: RGB is suitable for adjusting RGB three-color lights. RGBW is suitable for controlling RGBW light strips. Color Temperature is suitable for color temperature control of lights. As shown in Figures 7.4(1), (2), and (3).

Note: The APP panel converts RGB to HSI for adjustment, meaning it adjusts RGB values through hue (H), saturation (S), and brightness (I).

③ RGB value adjustment. Adjust the RGB color by sliding the brightness and saturation sliders. Range: 0%..100%.

④ RGB/RGBW value adjustment. Adjust the RGB color by sliding the hue slider.

⑤ Color temperature value adjustment. Adjust the color temperature by sliding the color temperature slider. Range: 2700..6500K.

⑥ Brightness value adjustment. Adjust the brightness by sliding the brightness slider. Range: 0..100%.

⑦ RGBW value adjustment. In the colored light interface, adjust the RGB color

by sliding the brightness and saturation sliders. Range: 0%..100%. In the white light interface, adjust the white light brightness by sliding the brightness slider.

⑧ Light switch button. Gray indicates the off state, bright color indicates the on state. When the current value is not 0, tapping this button will set the R/G/B/W or B/CW values to 0.
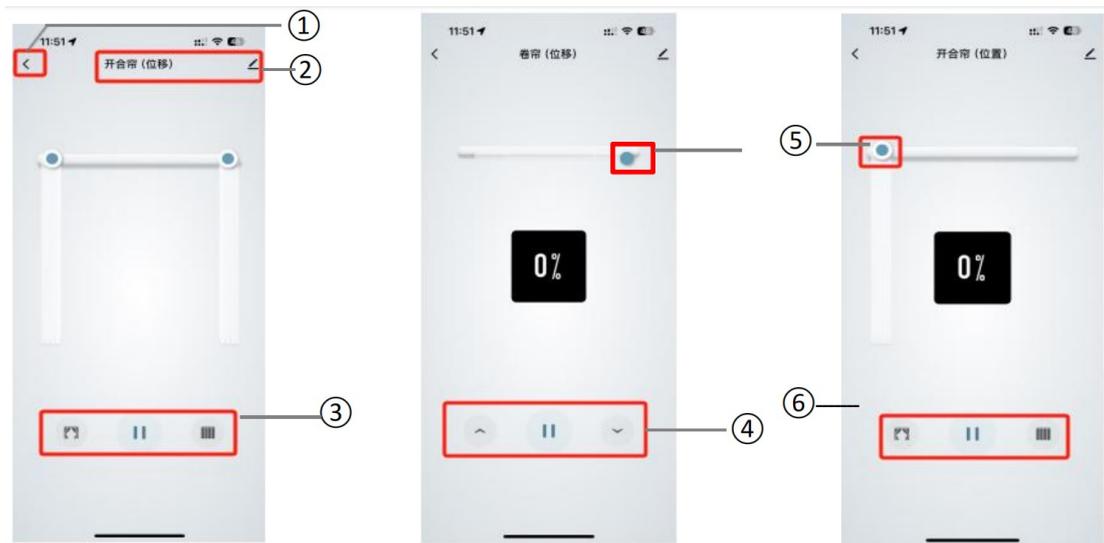
6.6 Curtain Control Interface

Figure 6.6(1) Displacement Open/Close Curtain   Figure 6.6(2) Displacement Roller Blind   Figure 6.6(3) Position Open/Close Curtain
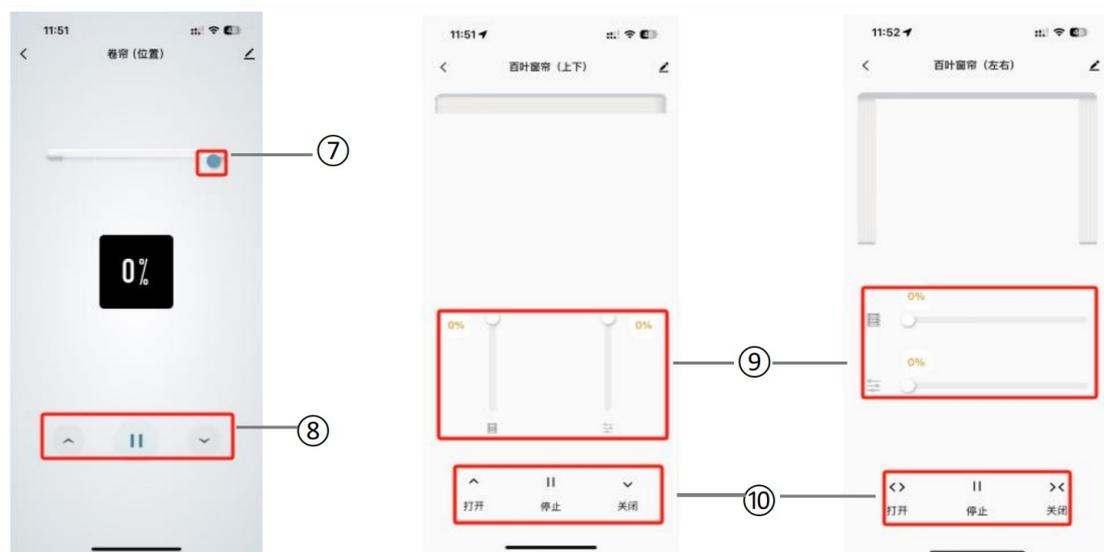

Figure 6.6(4) Position Roller Blind   Figure 6.6(5) Venetian Blind

① Return to the previous level interface.

② Device name, displayed at the top of the page. Click the icon ∠ to view, modify device settings, or remove the device.

③ Controlling displacement open/close curtains:

1) Tap the icon ⬚ to open the curtain. The device will send a message with the corresponding object (Open/Close) value set to 0.

2) Tap the icon ‖ to stop the curtain movement. The device will send a message with the corresponding object (Stop) value set to 1.

3) Tap the icon ▥ to close the curtain. The device will send a message with the corresponding object (Open/Close) value set to 1.

④⑤ Controlling displacement roller blinds:

1) Tap the icon ⌃ to move the blind upward (open). The device will send a message with the corresponding object (Up/Down) value set to 0.

2) Tap the icon ‖ to stop the blind movement. The device will send a message with the corresponding object (Stop) value set to 1.

3) Tap the icon ⌄ to move the blind downward (close). The device will send a message with the corresponding object (Up/Down) value set to 1.

4) Adjust the travel position of the curtain by sliding slider ⑤.

⑤⑥ **Controlling position open/close curtains:**

1) Adjust the travel position of the curtain by sliding slider ⑤.

2) The ⬚ ‖ ▥ open/close/stop operations are the same as above.

⑦⑧ **Controlling position roller blinds:**

1) Adjust the travel position of the blind by sliding slider ⑦.

2) The ⌃ ‖ ⌄ open/close/stop operations are the same as above.

⑨⑩ **Controlling Venetian blinds:**

1) Adjust the travel position and angle position of the Venetian blind by sliding slider ⑨.

2) The ^ 打开 || 停止 ∨ 关闭 open/close/stop operations are the same as above.

6.7 Value Sending Interface

Figure 6.7 Value Sending

① Return to the previous level interface.

② Device name, displayed at the top of the page. Click the icon ∠ to view, modify device settings, or remove the device.

③ Tap the icon to send a message to the KNX bus. The APP is only used to trigger commands and functions solely as a button operation.

The supported value types for sending are as follows: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]
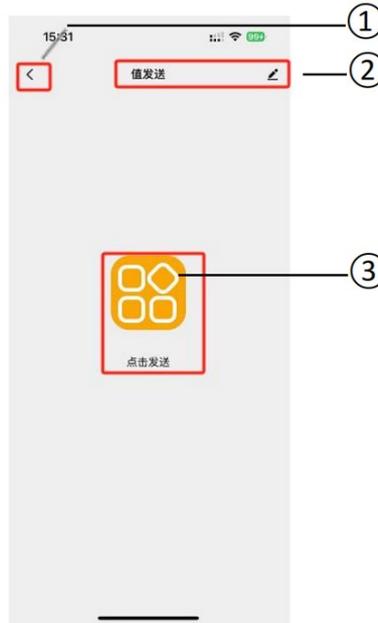
6.8 Scene Switch Interface
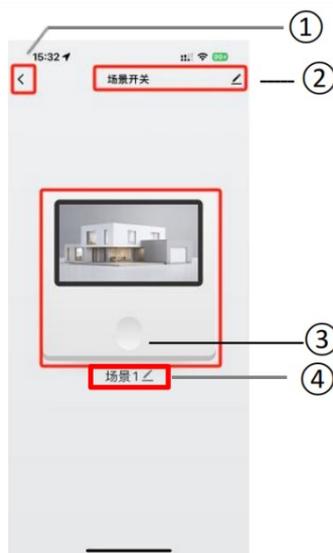
Figure 6.8(1) Scene Switch Interface

① Return to the previous level interface.

② Device name, displayed at the top of the page. Click the icon ∠ to view, modify device settings, or remove the device.

③ Press and hold this button to enter the scene settings. It can receive KNX scene call commands or be short-pressed in the APP to execute tasks configured in the scene settings, enabling linkage configurations between KNX device functions and Zigbee/Bluetooth device functions in the APP. Specific associated devices and trigger responses for the scene must be configured in the APP before operation.

④ Click "Scene 1" to navigate to Figure 6.8(2) for editing scene settings.

Add Scene Settings:

Figure 6.8(2) Edit Scene Settings   **Figure 6.8(3)** Create Scene   **Figure 6.8(4)** Add Task   **Figure 6.8(5)** Scene Display

①④⑦ Return to the previous level interface.

② Click to modify the scene name, scene settings, and scene image. Click to save the settings and return to the previous level interface.

③ Click to save the settings and return to the previous level interface.

On the APP home page, click "Scenes" to navigate to the Create Scene interface, as shown in Figure 6.8(3).

⑤ Click to create a "One-Touch Execution" task, which will jump to Figure 6.8(4). Click ⑧ to add a scene, including options in ⑥: all devices, existing smart scenes, selecting desired settings, state changes, and scheduled executions.

⑨ After creating the automation, click to save the settings and return to the previous level interface. Once successfully created, the task will be displayed in Figure 6.8(5).

Manage Scene Settings:
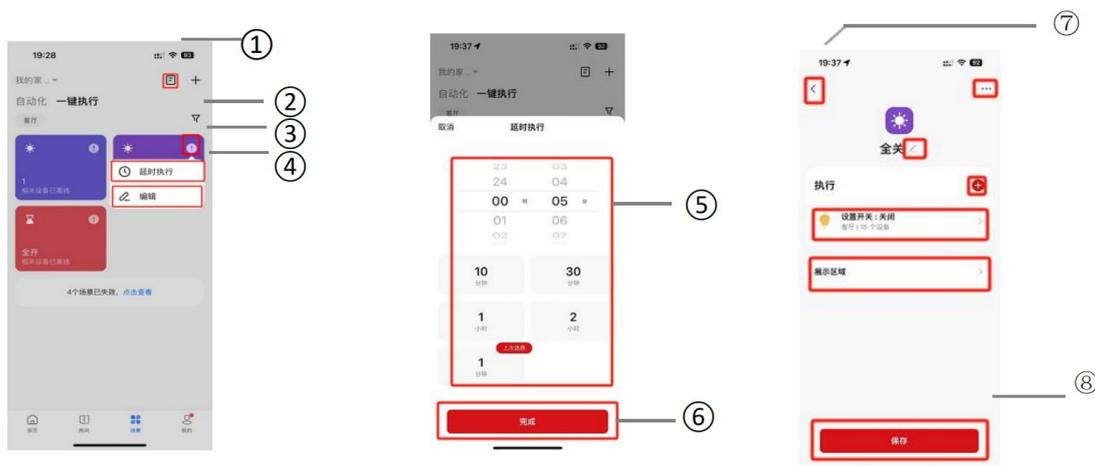
Figure 6.8(6) Smart Interface　**Figure 6.8(7)** Management Interface　**Figure 6.8(8)** Editing Interface

① Click the icon ▣ to view the scene log.

② Click the icon 🔵 to select Delay Execution or Edit. Click Delay Execution to jump to the delay execution interface, as shown in Figure 6.8(7). In ⑤, freely set the delay time, then click ⑥ to save and automatically return to the previous level interface.

③ Click Edit to jump to the scene editing interface, as shown in Figure 6.8(8). In this interface, you can perform the following operations: add tasks, modify the name, select display areas, and delete the scene.

The editing interface also allows access to individual devices for operations, such as clicking "One-Touch Execution" and "Automation" in the scene interface of Figure 6.8(5) to execute the scene.

⑦ Click the icon ‹ to cancel editing and return to the previous level interface.

⑧ After completing the editing, click save and return to the previous level interface.

6.9 Air Conditioning Control Interface
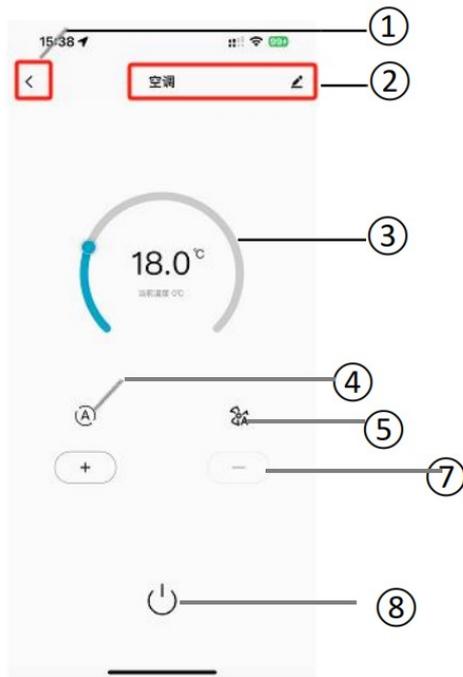
Figure 6.9(1) Air Conditioning Interface   Figure 6.9(2) Air Conditioning (Directional) Interface

① Return to the previous level interface.

② Device name, displayed at the top of the page. Click the icon ∠ to view, modify device settings, or remove the device.

③ Circular slider: Slide this control to adjust the set temperature. When the temperature unit is set to degrees Celsius (°C), the adjustment range for the set temperature is limited by default to 15–40°C. The current temperature is displayed based on parameter configuration, which can be configured to show the temperature detected by the local sensor or the temperature detected by an external sensor.

④ Tap this icon to switch the air conditioner's control mode. The status can be updated via objects for the following modes: Heating ☼ , Cooling ❄ , Dehumidification◌, Fan ⇒ , Auto◌.

⑤ Tap this icon to switch the fan speed. The current fan speed level cycles through low speed...AUTO...low speed in the following sequence: Low speed ❀, Medium speed❀, High speed❀, Auto❀.

⑥ Tap this icon to switch the swing mode: Swing or Fixed.

⑦ Tap "+" to raise the set temperature, or tap "−" to lower the set temperature. Each adjustment changes the temperature by 1 degree.

⑧ Air conditioner power button: Gray indicates the off state, and bright color indicates the on state. When the unit is off, except for the power state, the status of other icons cannot be updated via the bus or manually controlled. Tapping this button will cause the device to send a message with value 1/0 (On/Off) via the Power on/off object.

6.10 Thermostat Control Interface

Figure 6.10(1) Thermostat Interface
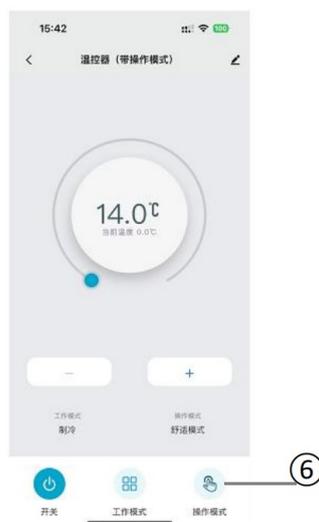
Figure 6.10(2) Thermostat

(Operation Mode) Interface
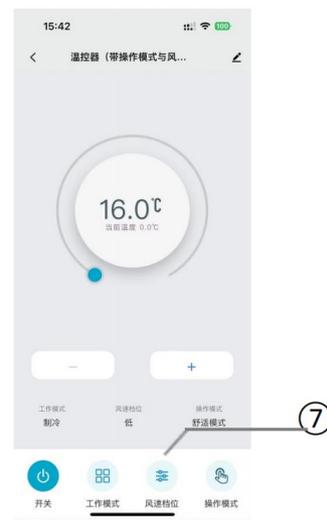
Figure 6.10(3) Thermostat

(Operation Mode & Fan Speed) Interface

① Return to the previous level interface.

② Device name, displayed at the top of the page. Click the icon ∠ to view, modify device settings, or remove the device.

③ Circular slider: Slide this control to adjust the set temperature. When the temperature unit is set to degrees Celsius (°C), the adjustment range for the set temperature is limited by default to 5–40°C. The current temperature is displayed based on parameter configuration, which can be configured to show the temperature

detected by the local sensor or the temperature detected by an external sensor.

④ Tap "+" to raise the set temperature, or tap "−" to lower the set temperature. Each adjustment changes the temperature by 1 degree.

⑤ Tap this icon to switch between heating and cooling modes. The status can be updated via objects for the following modes: Heating mode, Cooling mode.

When the thermostat control mode is set to "Heating and Cooling" in the database, the mode can be switched via touch or bus messages.

⑥ Tap this icon to switch the operation mode: Comfort mode, Standby mode, Energy-saving mode, Protection mode.

⑦ Tap this icon to switch the fan speed. The fan speed cycles from the current speed level in the sequence: low speed...AUTO...low speed, and the change takes effect immediately: Low speed, Medium speed, High speed, Auto speed.

⑧ Thermostat power button: Gray indicates the off state, and bright color indicates the on state. When the thermostat is off, except for the power state, the status of other icons cannot be updated via the bus or manually controlled. Tapping this button will cause the device to send a message with value 1/0 (On/Off) via the Power on/off object.

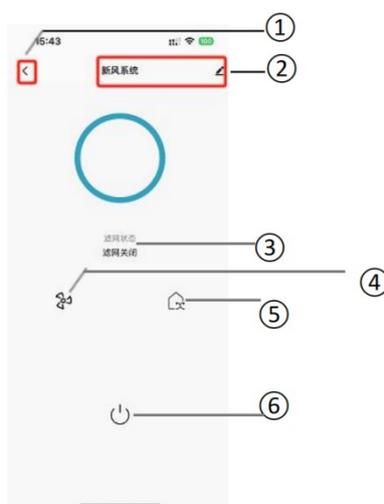## 6.11 Fresh Air System Control Interface

Figure 6.11 Fresh Air Control Interface

① Return to the previous level interface.

② Device name, displayed at the top of the page. Click the icon ∠ to view,

modify device settings, or remove the device.

③ Indicates the status of the filter (on/off).

④ Tap this icon to switch the fan speed. The fan speed cycles from the current level in the order: low...AUTO...low, and the change takes effect immediately: Low⚘, Medium❀, High❀, Auto⚘.

⑤ Tap this icon ⌂✕ to toggle the heat exchange function on/off.

⑥ Fresh air system power button: Gray indicates the off state, and bright color indicates the on state. When the system is off, except for the power status, the states of other icons cannot be updated via the bus or manually controlled. Tapping this button will cause the device to send a message with a value of 1/0 (On/Off) via the Power on/off object.

6.12 Floor Heating Control Interface

Figure 6.12 Floor Heating Control Interface

① Return to the previous level interface.

② Device name, displayed at the top of the page. Click the icon ∠ to view, modify device settings, or remove the device.

③ Circular slider: Slide this control to adjust the set temperature. When the temperature unit is set to degrees Celsius (°C), the adjustment range for the set temperature is limited by default to 5–40°C. The current temperature is displayed based on parameter configuration, which can be configured to show the temperature

detected by the local sensor or the temperature detected by an external sensor.

④ Tap "+" to raise the set temperature, or tap "−" to lower the set temperature. Each adjustment changes the temperature by 1 degree.

⑤ Thermostat power button: Gray indicates the off state, and bright color indicates the on state. When the thermostat is off, except for the power state, the status of other icons cannot be updated via the bus or manually controlled. Tapping this button will cause the device to send a message with value 1/0 (On/Off) via the Power on/off object.

6.13 Background Music Control Interface
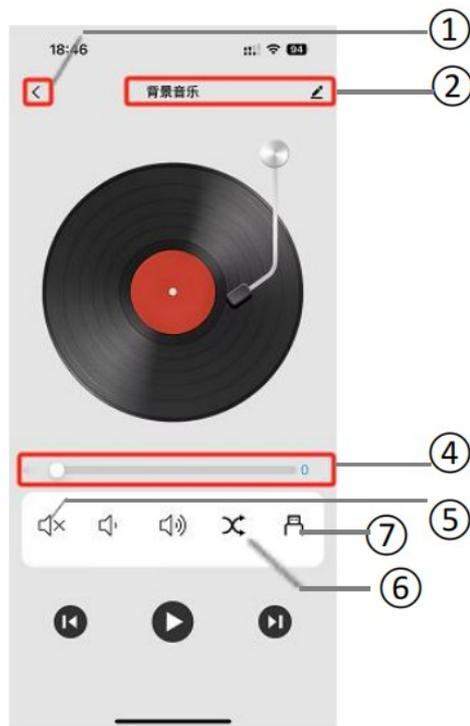
Figure 7.11(1) Background Music Control Interface   Figure 7.11(2) Background Music Control (with Switch) Interface
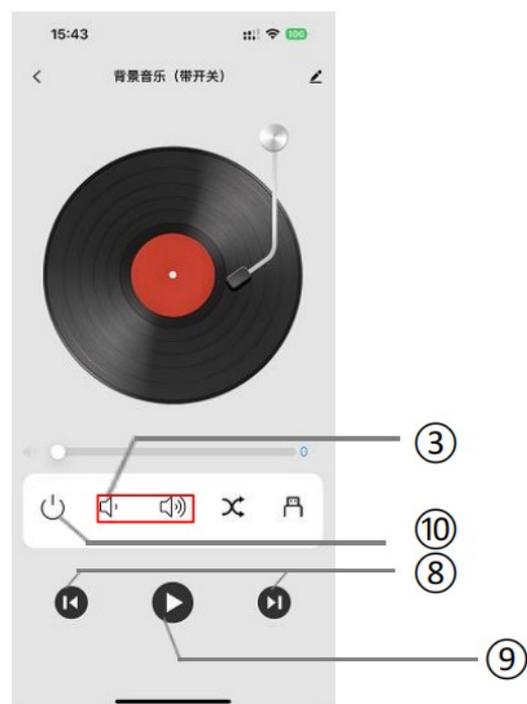
① Return to the previous level interface.

② Device name, displayed at the top of the page. Click the icon ∠ to view, modify device settings, or remove the device.

③ Tap the icon to increase or decrease the volume.

④ Slide the slider to adjust the volume down or up.

⑤ Tap the icon to mute or unmute: Mute◁×, Unmute◁».

⑥ Tap this button to switch the background music playback mode. The status can

be updated via objects: Single-cycle 🔁 , Sequential playback 🔄 , Random playback ✖.

⑦ Tap this button to switch the background music audio source. The status can be updated via objects: USB drive �️ , SD card 🖫 , AUX output ✎ , FM radio 📻 , Bluetooth 🔊 .

⑧ Tap the icon to select the previous or next track.

⑨ Tap the icon to play or pause the song. The status can be updated via objects.

⑩ Background music power button: Gray indicates the off state, and bright color indicates the on state. When the system is off, except for the power state, the status of other icons cannot be updated via the bus or manually controlled. Tapping this button will cause the device to send a message with value 1/0 (On/Off) via the Power on/off object.

6.14 Sensor Interface

Air quality sensor   Co2 sensor   PM2.5 sensor   VOC sensor

The air quality detector can display seven parameters: CO_2, PM2.5, PM10, VOC, current temperature, current humidity, and AQI. The displayed content can be configured via the database. CO_2, PM2.5, and VOC have individual detectors, and all can display indoor temperature and humidity on the interface.

The circular ring on the interface corresponds to the parameter values of the respective device. Updates to the displayed values can be achieved by modifying the corresponding objects via the bus.

AQI detection range: 0..500

CO_2 detection range: 0..4000 ppm

PM2.5 detection range: 0..999 µg/m³

VOC detection range: 0..999 µg/m³

Temperature detection range: -40...40°C

Humidity detection range: 0..100%
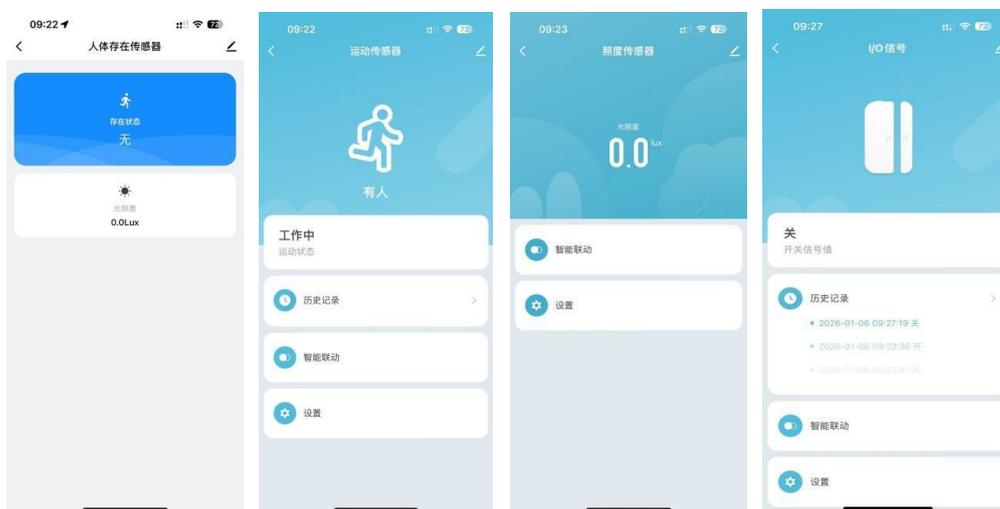
Human Presence Sensor   Motion Sensor   Light Sensor   Switch Signal Sensor

The human presence sensor can detect human presence status as well as illuminance, with a detection range of 0...65535 lux.

The motion sensor can only detect human presence status.

The light sensor can only detect illuminance, with a detection range of 0...65535 lux.

The switch signal sensor can detect the on/off status of a device.

6.15 Current/ Energy Information Interface
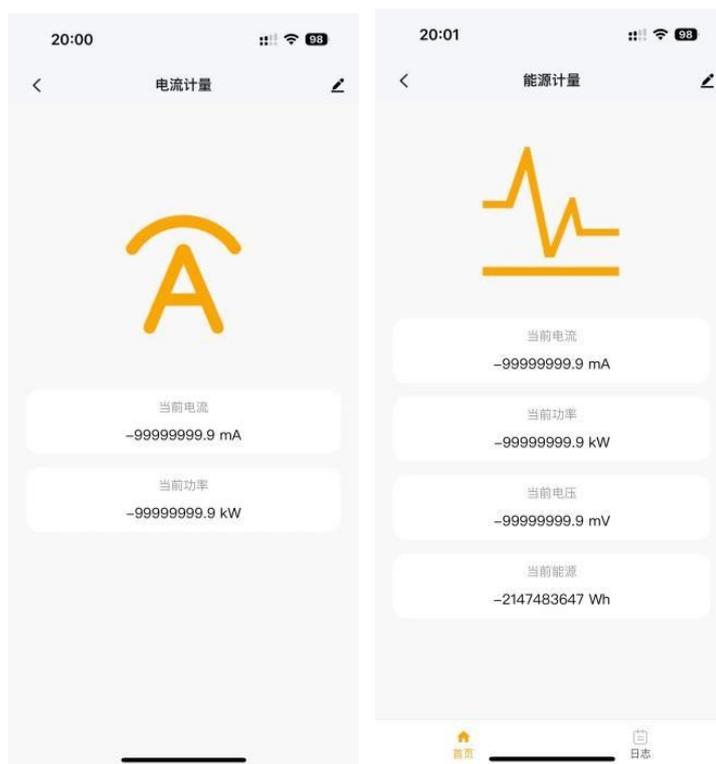
| Current Meter | Energy Meter |

The current meter detects the current and power of the device and displays real-time changes in status. The unit and resolution of the parameters can be modified by rewriting the corresponding objects via the bus.

The energy meter detects the current, power, voltage, and energy consumption of the device and displays real-time changes in status. The unit and resolution of the parameters can be modified by rewriting the corresponding objects via the bus.

6.16 Voice Control Commands

The KNX Fusion Gateway-LG1 currently supports voice control for KNX devices under the gateway using mainstream voice assistants such as Tuya Smart Assistant, Amazon Alexa, Google Assistant, Baidu Xiaodu, and Xiaomi Xiaoai.

This section primarily explains the intelligent voice control commands in the Tuya Smart APP. Users can refer to the following table to find voice control commands for device functions. Note: Currently, only devices listed below are supported; those not included are not supported.

ETS Corresponding Device Functions	Voice Command Instructions Note: The specific content within <> is determined by the name edited in the APP. The example commands are only supported in Chinese, and the same applies below.	
	Xiaozhi	Xiaozhi
Switch	Turn on <device name> Turn off <device name>	Xiaoai, turn on <device name> Xiaoai, turn off <device name>
Switch/ Dimming	**Light Switch:** Xiaozhi, turn on <device name> Xiaozhi, turn off <device name> Xiaozhi, check the switch status of <device name> **Brightness Adjustment:** Xiaozhi, adjust the brightness of <device name> to 15% Xiaozhi, increase the brightness of <device name> a little Xiaozhi, decrease the brightness of <device name> a little Xiaozhi, what is the brightness level of <device name>?	**Light Switch:** Xiaoai, turn on <device name> Xiaoai, turn off <device name> **Brightness Adjustment:** Xiaoai, adjust the brightness of <device name> to 50% Xiaoai, brighten <device name> a little Xiaoai, dim <device name> a little
RGB Dimming	**Light Switch:** Xiaozhi, turn on <device name> Xiaozhi, turn off <device name> Xiaozhi, check the switch status of <device name>	**Light Switch:** Xiaoai, turn on <device name> Xiaoai, turn off <device name>
RGBW Dimming	**Light Switch:** Xiaozhi, turn on <device name> Xiaozhi, turn off <device name> Xiaozhi, check the switch status of <device name>	**Light Switch:** Xiaoai, turn on <device name> Xiaoai, turn off <device name> **Light Switch:** Xiaoai, turn on <device

	Light Switch: Xiaozhi, turn on <device name> Xiaozhi, turn off <device name> Xiaozhi, check the switch status of <device name> **Brightness Adjustment:** Xiaozhi, adjust the brightness of <device name> to 15% Xiaozhi, increase the brightness of <device name> a little Xiaozhi, decrease the brightness of <device name> a little Xiaozhi, what is the brightness level of <device name>? **Color Temperature Setting:** Xiaozhi, adjust the color temperature of <device name> to 15% Xiaozhi, make the color temperature of <device name> a bit warmer Xiaozhi, make the color temperature of <device name> a bit cooler Xiaozhi, what is the color temperature of <device name>?	name> Xiaoai, turn off <device name> **Brightness Adjustment:** Xiaoai, adjust the brightness of <device name> to 50% Xiaoai, brighten <device name> a little Xiaoai, dim <device name> a little
Curtain step/ move	**Curtain Switch:** Xiaozhi, open <device name> Xiaozhi, close <device name> Xiaozhi, check the switch status of <device name> **Set Working State:**	**Curtain Switch:** Xiaoai, open <device name> Xiaoai, close <device name> **Set Working State:** Xiaoai, pause <device name>

		Xiaozhi, enable working state for <device name> Xiaozhi, pause working state for <device name>	Xiaoai, resume <device name>
Roller blind step/ move		**Curtain Switch:** Xiaozhi, open <device name> Xiaozhi, close <device name> Xiaozhi, check the switch status of <device name> **Set Working State:** Xiaozhi, enable working state for <device name> Xiaozhi, pause working state for <device name>	**Curtain Switch:** Xiaoai, open <device name> Xiaoai, close <device name> **Set Working State:** Xiaoai, pause <device name> Xiaoai, resume <device name>
Curtain position		**Curtain Switch:** Xiaozhi, open <device name> Xiaozhi, close <device name> Xiaozhi, check the switch status of <device name> **Set Working State:** Xiaozhi, enable working state for <device name> Xiaozhi, pause working state for <device name> **Percentage Position Adjustment:** Xiaozhi, adjust the position of <device name> to 35% Xiaozhi, open <device name> a bit more Xiaozhi, close <device name> a bit more	**Curtain Switch:** Xiaoai, open <device name> Xiaoai, close <device name> **Set Working State:** Xiaoai, pause <device name> Xiaoai, resume <device name> **Percentage Position Adjustment:** Xiaoai, adjust <device name> to 50%
Roller blind position		**Curtain Switch:** Xiaozhi, open <device name> Xiaozhi, close <device name>	**Curtain Switch:** Xiaoai, open <device name> Xiaoai, close <device name>

	Xiaozhi, check the switch status of <device name> **Set Working State:** Xiaozhi, enable working state for <device name> Xiaozhi, pause working state for <device name> **Percentage Position Adjustment:** Xiaozhi, adjust the position of <device name> to 35% Xiaozhi, open <device name> a bit more Xiaozhi, close <device name> a bit more	**Set Working State:** Xiaoai, pause <device name> Xiaoai, resume <device name> **Percentage Position Adjustment:** Xiaoai, adjust <device name> to 50%
Venetian blind position and slat	**Curtain Switch:** Xiaozhi, open <device name> Xiaozhi, close <device name> Xiaozhi, check the switch status of <device name> **Set Working State:** Xiaozhi, enable working state for <device name> Xiaozhi, pause working state for <device name> **Percentage Position Adjustment:** Xiaozhi, adjust the position of <device name> to 35% Xiaozhi, open <device name> a bit more Xiaozhi, close <device name> a bit more **Blind Angle:** Temporarily not set	**Curtain Switch:** Xiaoai, open <device name> Xiaoai, close <device name> **Set Working State:** Xiaoai, pause <device name> Xiaoai, resume <device name> **Percentage Position Adjustment:** Xiaoai, adjust <device name> to 50%
Scene Switch	Turn on <device name> Turn off <device name>	Xiaoai, turn on <device name>

LETOUR

		Xiaoai, turn off <device name>
Air conditioner	**Power Switch:** Xiaozhi, turn on <device name> Xiaozhi, turn off <device name> Xiaozhi, check the switch status of <device name> **Set Control Mode:** Xiaozhi, adjust the mode of <device name> to <mode name> **Set Temperature:** Xiaozhi, set the temperature of <device name> to 27 degrees Xiaozhi, increase the temperature of <device name> a bit Xiaozhi, decrease the temperature of <device name> a bit **Wind Speed (Gear Type):** Xiaozhi, adjust the wind speed of <device name> to <speed level> Xiaozhi, check the wind speed of <device name>	**Power Switch:** Xiaoai, turn on <device name> Xiaoai, turn off <device name> **Set Control Mode:** Xiaoai, switch <device name> to <mode name> mode **Set Temperature:** Xiaoai, set the temperature of <device name> to 35 degrees Xiaoai, raise the temperature of <device name> a bit Xiaoai, lower the temperature of <device name> a bit **Query Ambient Temperature:** Xiaoai, query the temperature for <device name>
Air condition （with swing）	**Power Switch:** Xiaozhi, turn on <device name> Xiaozhi, turn off <device name> Xiaozhi, check the switch status of <device name> **Set Control Mode:** Xiaozhi, adjust the mode of <device name> to <mode name> **Set Temperature:**	**Power Switch:** Xiaoai, turn on <device name> Xiaoai, turn off <device name> **Set Control Mode:** Xiaoai, switch <device name> to <mode name> mode **Set Temperature:** Xiaoai, set the temperature of <device name> to 35

	Xiaozhi, set the temperature of <device name> to 27 degrees Xiaozhi, raise the temperature of <device name> a bit Xiaozhi, lower the temperature of <device name> a bit **Wind Speed (Gear Type):** Xiaozhi, adjust the wind speed of <device name> to <speed level> Xiaozhi, check the wind speed of <device name> **Left-Right Swing:** Xiaozhi, turn on left-right swing for <device name> Xiaozhi, turn off left-right swing for <device name> Xiaozhi, check the left-right swing status of <device name>	degrees Xiaoai, raise the temperature of <device name> a bit Xiaoai, lower the temperature of <device name> a bit **Query Ambient Temperature:** Xiaoai, query the temperature for <device name>
Room Temperature unit	**Power Switch:** Xiaozhi, turn on <device name> Xiaozhi, turn off <device name> Xiaozhi, check the switch status of <device name> **Set Control Mode:** Xiaozhi, adjust the mode of <device name> to <mode name> **Set Temperature:** Xiaozhi, set the temperature of <device name> to 27 degrees Xiaozhi, increase the temperature of <device name> a bit Xiaozhi, decrease the	**Power Switch:** Xiaoai, turn on <device name> Xiaoai, turn off <device name> **Set Temperature:** Xiaoai, set the temperature of <device name> to 35 degrees Xiaoai, raise the temperature of <device name> a bit Xiaoai, lower the temperature of <device name> a bit **Query Ambient Temperature:** Xiaoai, query the temperature for <device

|		
	temperature of <device name> a bit **Query Ambient Temperature:** Xiaozhi, query the ambient temperature	name>
Room Temperature unit (with operation mode)	**Power Switch:** Xiaozhi, turn on <device name> Xiaozhi, turn off <device name> Xiaozhi, check the switch status of <device name> **Set Control Mode:** Xiaozhi, adjust the mode of <device name> to <mode name> **Set Temperature:** Xiaozhi, set the temperature of <device name> to 27 degrees Xiaozhi, increase the temperature of <device name> a bit Xiaozhi, decrease the temperature of <device name> a bit **Query Ambient Temperature:** Xiaozhi, query the ambient temperature **Note:** Operation modes such as Comfort, Energy Saving, etc., are temporarily not supported.	**Power Switch:** Xiaoai, turn on <device name> Xiaoai, turn off <device name> **Set Temperature:** Xiaoai, set the temperature of <device name> to 35 degrees Xiaoai, raise the temperature of <device name> a bit Xiaoai, lower the temperature of <device name> a bit **Query Ambient Temperature:** Xiaoai, query the temperature for <device name>
Room Temperature unit (with operation mode& fan speed)	**Power Switch:** Xiaozhi, turn on <device name> Xiaozhi, turn off <device name> Xiaozhi, check the switch	**Power Switch:** Xiaoai, turn on <device name> Xiaoai, turn off <device name> **Set Temperature:**

		status of <device name>	Xiaoai, set the temperature of <device name> to 35 degrees
		Set Control Mode: Xiaozhi, adjust the mode of <device name> to <mode name>	Xiaoai, raise the temperature of <device name> a bit
		Set Temperature: Xiaozhi, set the temperature of <device name> to 27 degrees Xiaozhi, increase the temperature of <device name> a bit Xiaozhi, decrease the temperature of <device name> a bit	Xiaoai, lower the temperature of <device name> a bit
			Query Ambient Temperature: Xiaoai, query the temperature for <device name>
		Query Ambient Temperature: Xiaozhi, query the ambient temperature	
		Wind Speed (Gear Type): Xiaozhi, adjust the wind speed of <device name> to <speed level> Xiaozhi, check the wind speed of <device name>	
		Note: Operation modes such as Comfort, Energy Saving, etc., are temporarily not supported.	
Floor heating		**Power Switch:** Xiaozhi, turn on <device name> Xiaozhi, turn off <device name> Xiaozhi, check the switch status of <device name>	**Power Switch:** Xiaoai, turn on <device name> Xiaoai, turn off <device name>
		Set Temperature: Xiaozhi, set the temperature of <device name> to 27 degrees Xiaozhi, increase the	**Set Temperature:** Xiaoai, set the temperature of <device name> to 35 degrees Xiaoai, raise the temperature of <device name> a bit

	temperature of <device name> a bit Xiaozhi, decrease the temperature of <device name> a bit **Query Ambient Temperature:** Xiaozhi, query the ambient temperature	Xiaoai, lower the temperature of <device name> a bit **Query Ambient Temperature:** Xiaoai, query the temperature for <device name>

6.17 Cross-Protocol Device Interoperability Control

The KNX Fusion Gateway-LG1 enables mutual linkage and bidirectional control among KNX, Zigbee, and Bluetooth devices under the gateway through the APP's scene creation and automation features.

To create scenes or automation, you can click the icon + in the upper-right corner of the APP homepage, or switch to the scene/automation interface by tapping the scene icon at the bottom, then click "Create Smart Scene" to begin setup.
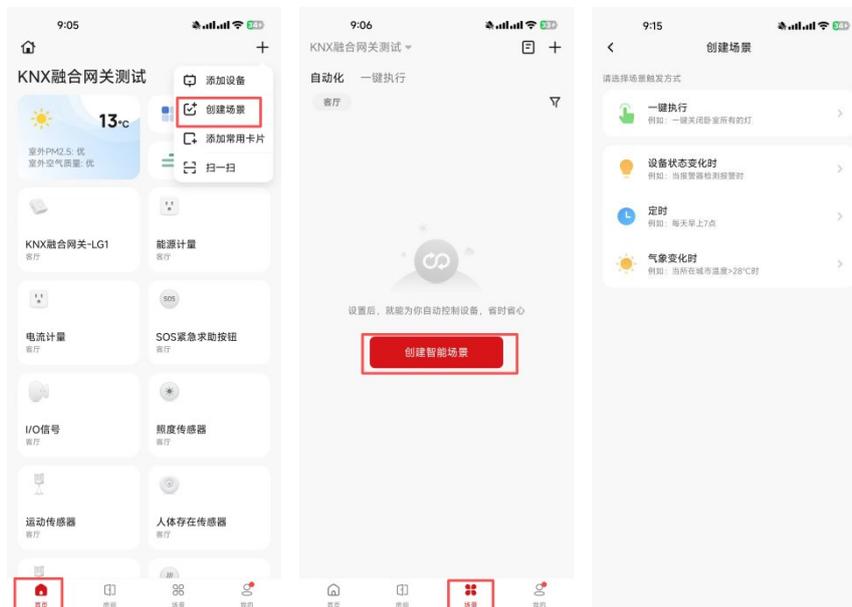

Note: Automation functions created in the APP are automatically saved locally on the gateway. Even if the gateway is offline or disconnected from the network, KNX devices, Zigbee devices, and Bluetooth devices under the gateway can still
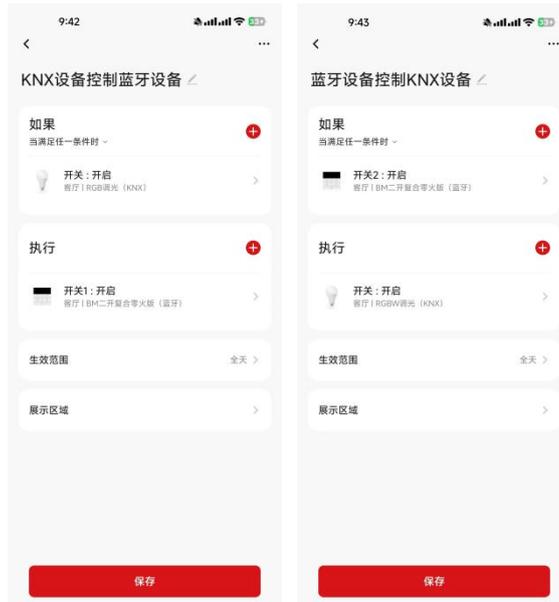
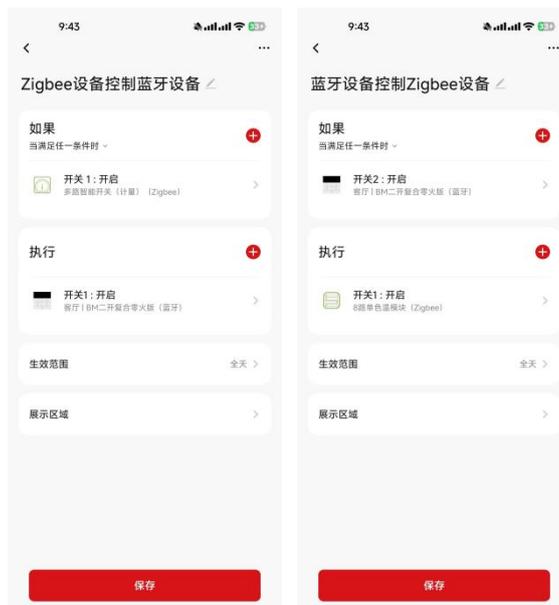perform cross-device linkage and control according to the pre-configured automation rules.

(1) Interconnected Control between KNX Devices and Zigbee Devices
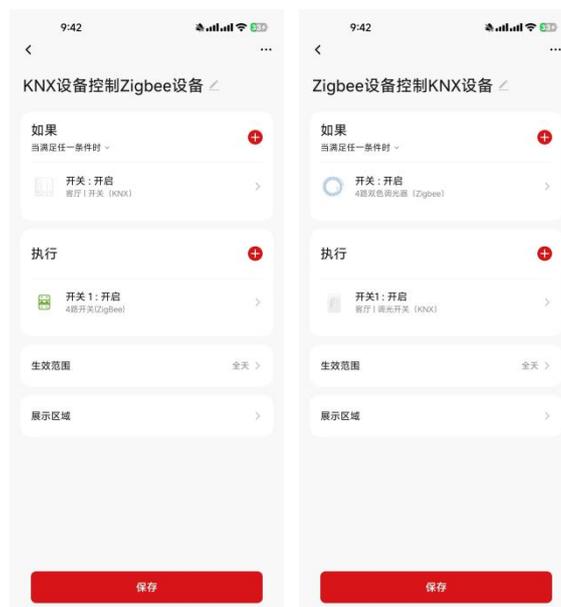
(2) Interconnected Control between KNX Devices and Bluetooth Devices

(3) Interconnected Control between ZigBee Devices and Bluetooth Devices

6.18 Adding Backup Network to Gateway

In the APP, click the icon ✎ to enter the gateway settings interface, then select "Network Settings" to access the gateway network configuration page. Click "Add Backup Network" to configure a backup Wi-Fi network for the gateway. If the wired network or the primary Wi-Fi network fails, the gateway will automatically switch to the backup network.