User Manual

K-BUS KNX Gateway for RS485/RS232_V2.1

BTPG-04/03.1

KNX/EIB Home and Building Control System

Attentions

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

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

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

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

Content

Chapter 1 Summary

KNX Gateway for RS485/RS232 mainly applied in the intelligent control system and installed on the distribution boards with 35mm mounting rail, which can realize Modbus/RS485/RS232 and KNX bus communication.

This manual provides detailed technical information about the KNX Gateway for RS485/RS232 for users as well as assembly and programming details, and explains how to use the device by the application examples.

KNX Gateway for RS485/RS232 powered from KNX bus, and need a 12-30V DC auxiliary supply voltage. It is available to assign the physical address and configure the parameters by engineering design tools ETS with .knxprod (support edition ETS5.7 or higher).

The functions are summarized as followed:

- **Configure basic parameters of communication, such as Baud rate, data bit, stop bit, parity bit and etc.**
- **Channel supports Max.500 datapoints. Each function point is undirectional, which can be configured direction, name and datatype(1bit/2bit/4bit/1byte/2byte).**
- **As Modbus master, read register data(up to 125 registers at a time) from slave and communicate with KNX.**
- **As Modbus slave, report KNX data to master or BA system.**
- **As normal gateway, only converting data, without communication mechanisms and logic.**
- **Support DAIKIN, HITACHI, Mitsubishi and other VRV Air conditioners in Modbus RTU mode, independent control up to 64 devices.**
- **Support some manufacturers of electric curtains and background music host control protocol.**

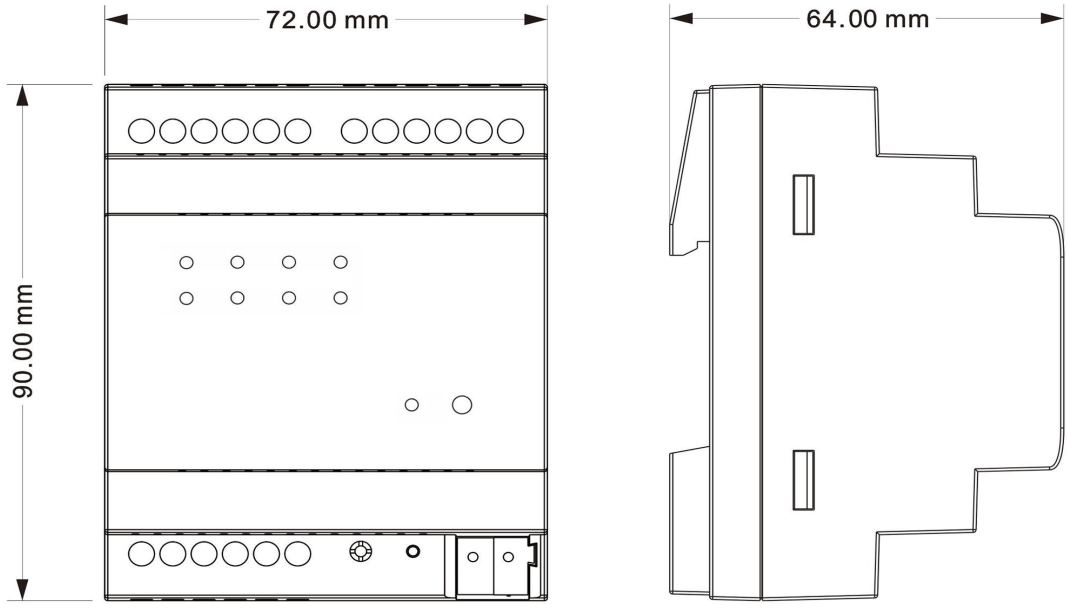
Chapter 2 Technical Data

Power Supply	Operation voltage	21-30V DC, via the KNX bus
	Bus current	<12mA 30V DC
	Bus power	<360mW
Auxiliary supply	Voltage	12-30V DC
	Current	<60mA 30V DC
	Power consumption	<1.8W
Connection	KNX	Bus connection terminal(red/black)
	Auxiliary power	Screw terminals
	RS485/RS232	Screw terminals
	Wire range	0.2-2.5mm^2
	Torque	0.4N-m
Operation and display	Channel power LED	Yellow, channel power normal
	Channel communication LED	Red flashing, telegram KNX->Other protocol Green flashing, telegram Other protocol ->KNX
	Programming button and LED	Red, assign physical address
Temperature	Operation	−5 °C ... + 45 °C
	Storage	−25 °C ... + 55 °C
	Transportation	− 25 °C ... + 70 °C
Ambient	Humidity	<93%, except dewing
Design	Standard 35 mm DIN rail installation	
Dimension	72 x 90 x 64mm	
Weight	0.17KG	

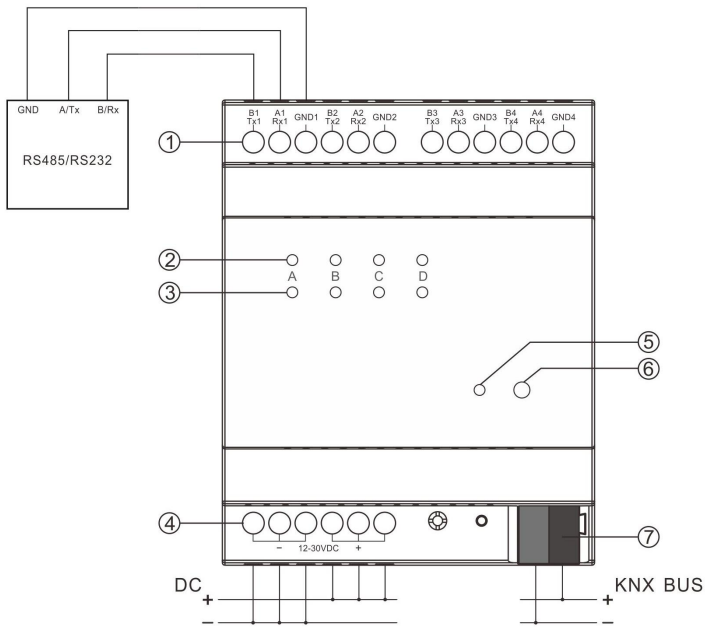
Note: The RS485 differential interface of this device complies with EIA/TIA RS-485-A standard and is compatible with 120Ω characteristic impedance industrial shielded twisted-pair cable. For bus networking, connect a 120Ω terminating resistor between terminals A and B of the first and last devices on the bus; no terminating resistor is required for intermediate devices. The terminating resistor may be omitted when the communication distance is ≤ 10 m and the baud rate is ≤ 19.2 kbps. (The 120Ω 1/2W terminating resistors are included in the product packaging)

Chapter 3 Dimension and Connection Diagram

3.1. Dimension diagram



3.2. Connection diagram



① RS485/RS232 interface

B1, A1 as RS485 interface of Channel 1.

TX1, RX1 as RS232 interface of Channel 1.

Other channels are same as channel 1, not repeat it here.

Note:

Two types of communication of RS485&RS232 need to correspond to the product hardware module and database setting.

220V strong current not allowed to access!

② Channel power LED indicator

③ Channel communication LED indicator

④ Auxiliary power screw terminals

⑤ Programming LED

⑥ Programming button

⑦ KNX bus connection terminal

Chapter 4 Project design and programming

Applications	Maximum of communication objects	Maximum number of group addresses	Maximum number of associations
KNX Gateway for RS485/RS232/1.1	**2049**	**4000**	**4000**

General function

General function includes heartbeat packet, KNX telegram delay time setting and enabling channel functions.

Communication datapoints

Channel supports up to 500 function datapoints. Each function point is undirectional, which can be configured direction, name and datatype(1bit/2bit/4bit/1byte/2byte).

Modbus master function

As Modbus master, read register data(up to 125 registers at a time) from slave and communicate with KNX.

Modbus slave function

As Modbus slave, report KNX data to master or BA system.

Communication data conversion

As normal gateway, only converting data, without communication mechanisms and logic.

Air-conditioning connection

Support DAIKIN, HITACHI, Mitsubishi and other VRV Air conditioners in Modbus RTU mode, independent control up to 64 devices.

Other

Support some manufacturers of electric curtains and background music host control protocol.

Chapter 5 Parameter setting description in the ETS

5.1.Parameter window “General setting”

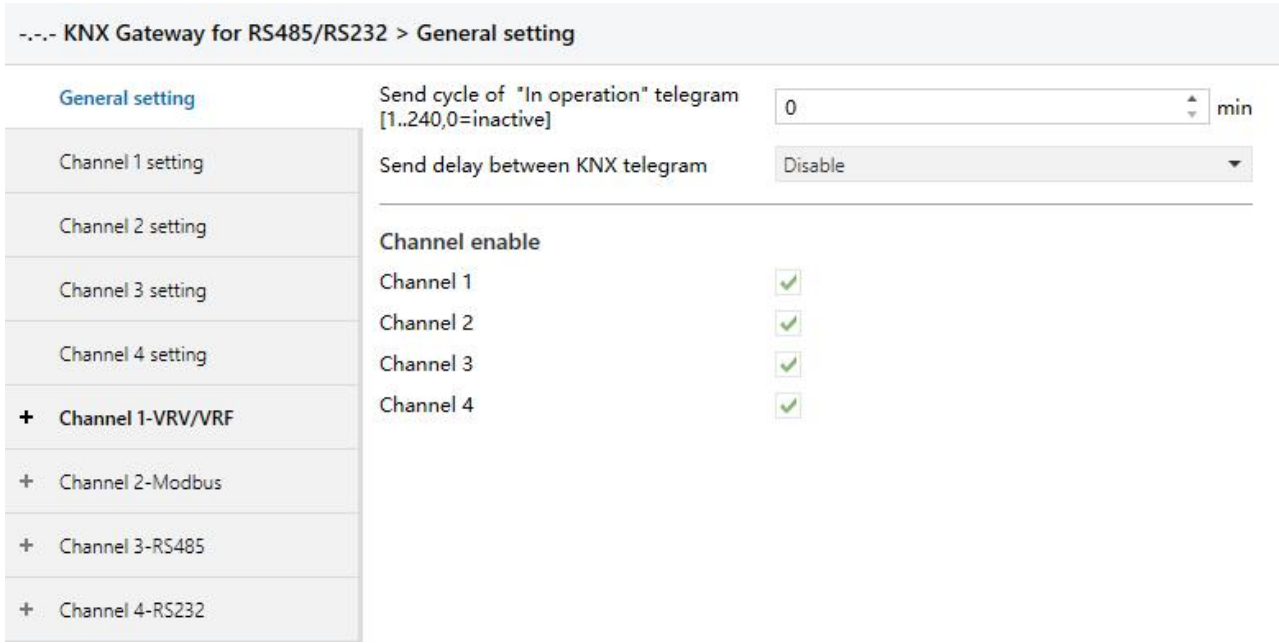


Fig.5.1“General setting” parameter window

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

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

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

Parameter "Send delay between KNX telegram”

This parameter is for setting the send delay between KNX telegram. When set the “Disable”, KNX telegram is sent without delay. Options:

Disable

50ms

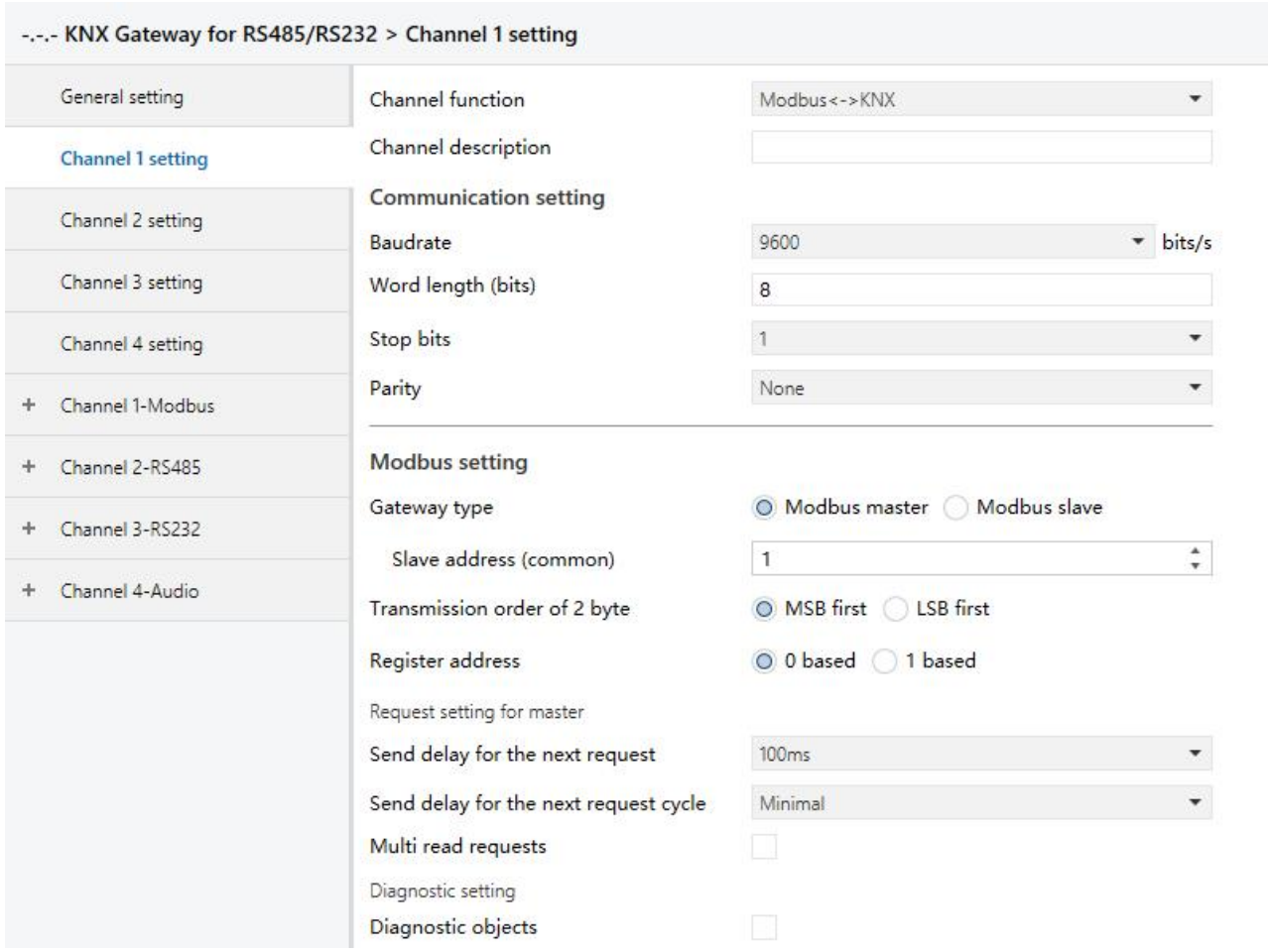
...

500ms

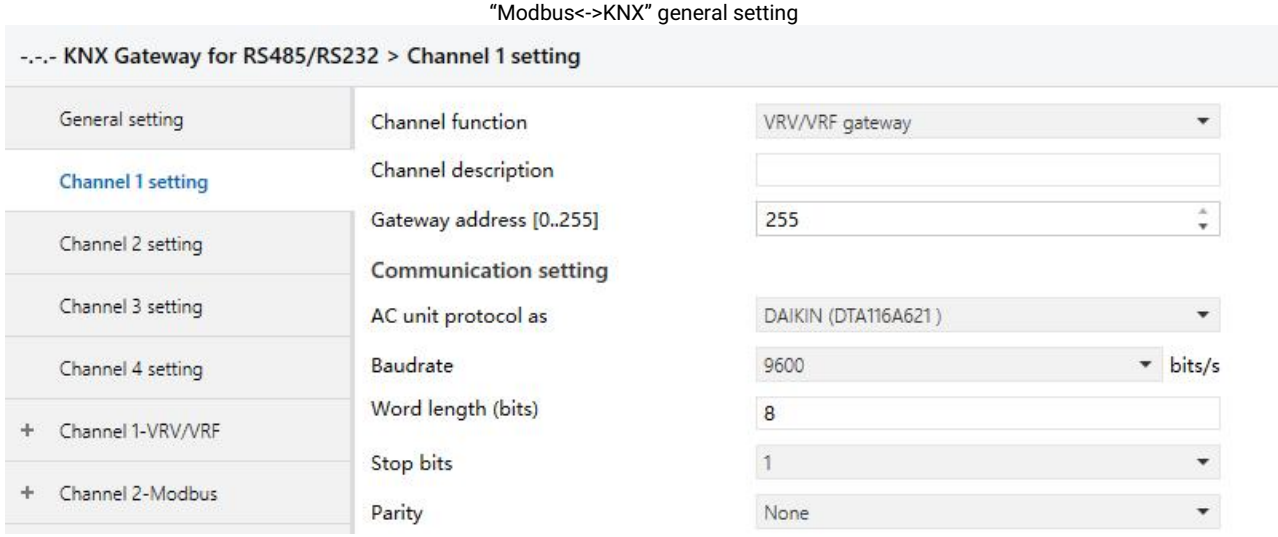
Parameter "Channel x" (x=1~4)

This parameter is for enabling the channel function.

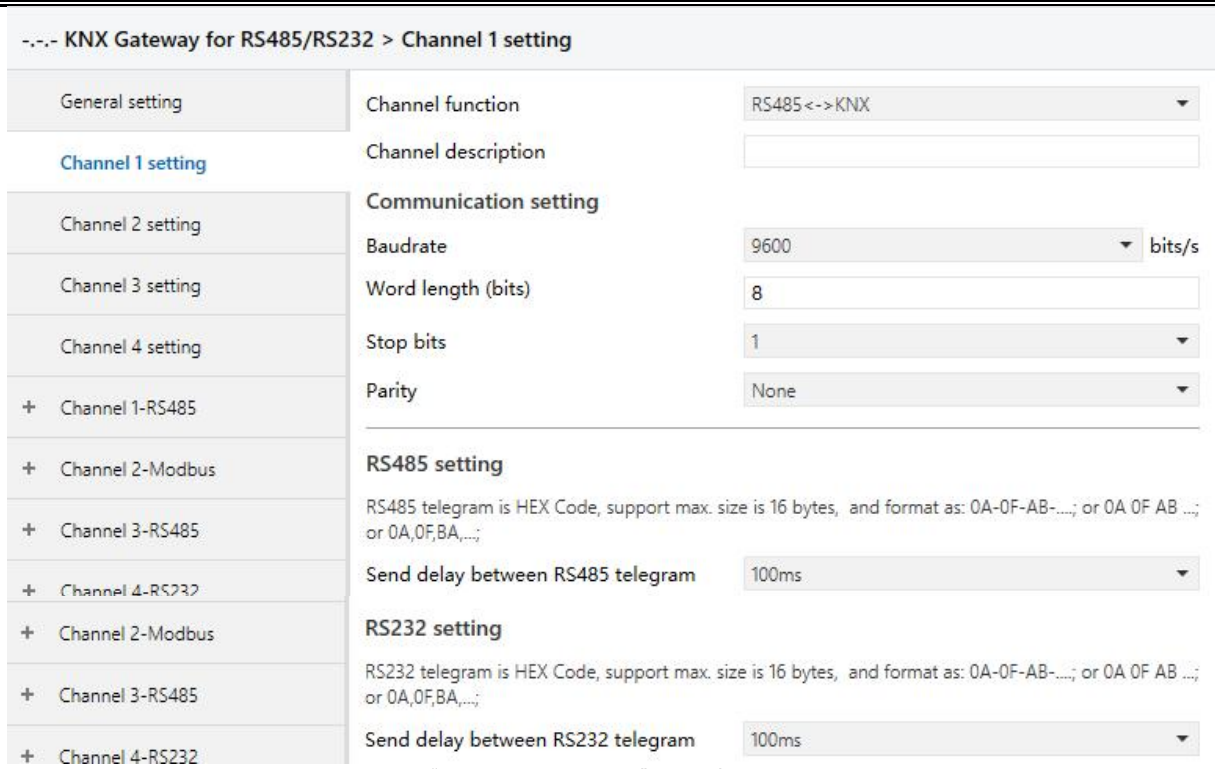
5.2. Parameter window “Channel x setting”



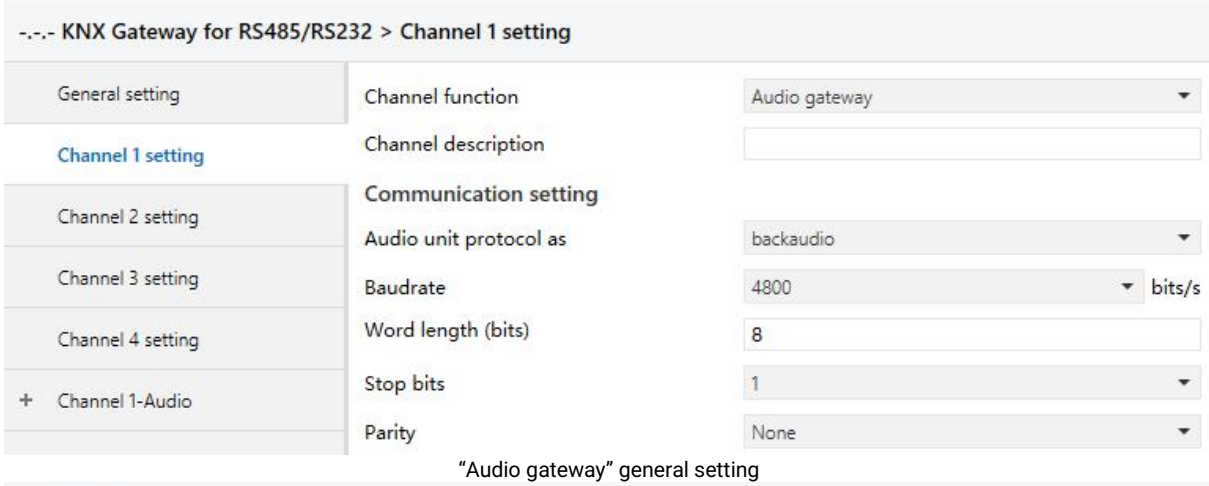
“Modbus<->KNX” general setting



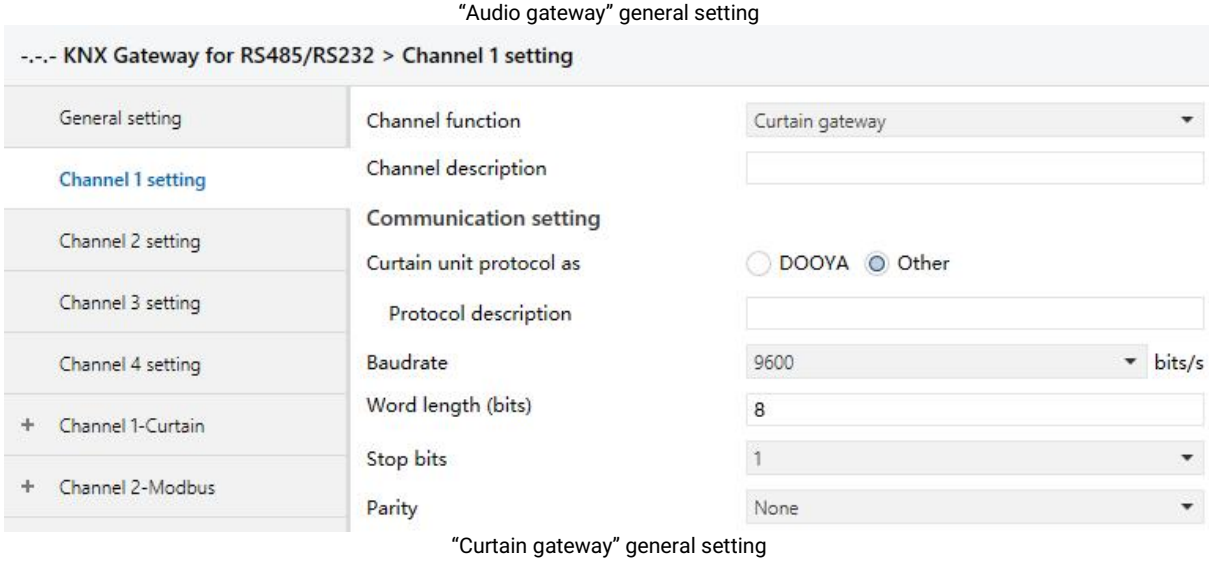
“VRV/VRF gateway” general setting



"RS485/RS232<->KNX" general setting



"Audio gateway" general setting



"Curtain gateway" general setting

Fig.5.2 "Channel x setting" parameter window

Parameter "Channel function"

This parameter is for setting channel function. Options:

Modbus<->KNX

RS485<->KNX

RS232<->KNX

VRV/VRF gateway

Audio gateway

Curtain gateway

Note: VRV/VRF gateway is not supported on Channel 2/3/4. Curtain gateway is not supported on Channel 3/4.

Parameter "Channel description"

This parameter is for setting the custom description of channel, up to 30 characters can be input.

Parameter "Gateway address [0..255]"

This parameter is visible when channel function is selected as "VRV/VRF gateway". Set the gateway address. Options: **0..255**

Communication setting

Parameter "AC unit protocol as"

This parameter is visible when channel function is selected as "VRV/VRF gateway". Set Air-conditioning unit protocol, and DTA116A621 is currently temporarily supported. Options:

DAIKIN (DTA116A621)
HITACHI (HL03B)
Mitsubishi (ABJK-CCM10)
Fujitsu (UTY-VMGX)
Gree
Toshiba (IFMB645TLE)
Other

Parameter "Curtain unit protocol as"

This parameter is visible when channel function is selected as "Curtain gateway". Set Curtain unit protocol, and DOOYA motor is currently temporarily supported. Options:

DOOYA
Other

Parameter "Audio unit protocol as"

This parameter is visible when channel function is selected as "Audio gateway". Set Audio unit protocol Options:

backaudio
Yodaar
MiYue
cnWise
Other

---Parameter "Protocol description"

This parameter is visible when the above 3 parameters is selected as "Other". Set the name of other protocol, up to 4 characters can be input.

Parameter "Baudrate"

This parameter is for setting the baud rate of communication. Options:

1200 bits/s
2400 bits/s
...
115200 bits/s

Parameter "Word length (bits)"

This parameter is for setting the word length. The default length is 8.

Parameter "Stop bits"

This parameter is for setting the stop bits of data frames. Options: **0.5 / 1 / 1.5 / 2**

Parameter "Parity"

This parameter is for setting the Parity of data frames。Options:

Even
Odd
None

Modbus setting(The following parameters are visible when channel function is selected as "Modbus<->KNX")

Parameter "Gateway type"

This parameter is for setting the Gateway type. Options:

Modbus master
Modbus slave

Note: Channel 2/3/4 only supports Modbus master.

——Parameter "Slave address (common)"

If gateway used as slave, this parameter enters slave address;

If gateway used as master, this parameter enters slave common address. If the slave does not use common address, you can configure additional addresses in the channel page.

Options: **0..247**

Parameter "Transmission order of 2 byte"

This parameter is for setting the sequence of the 2-byte-value data transmission. Options:

MSB first

LSB first

Parameter "Register address"

This parameter is for setting register address is defined based on 0 or 1. Options:

0 based

1 based

Request setting for master

——Parameter "Send delay for the next request"

This parameter is visible when gateway type is selected as "Modbus master". For setting delay of the next request. Options:

100ms

200ms

300ms

500ms

Note: The setting of the request time should be configured reasonably according to the response time of the access device.

——Parameter "Send delay for the next request cycle"

This parameter is visible when gateway type is selected as "Modbus master". For setting delay of the next request cycle. That is, the time interval to start the next round of requests, which cannot be set too short and ensure that the last round of requests is completed, otherwise the later unrequested telegram will be ignored. Options:

Minimal
1s
2s
...

10s

---Parameter "Multi read requests"

This parameter is visible when gateway type is selected as "Modbus master". Set enabled, the slave address and function code are the same, and the register address continuous or duplicate channels can be combined into a multiple read request.Up to 16 channels can be combined.

Diagnostic settings

Parameter "Diagnostic objects"

This parameter is for setting the object "Diagnostic: Slave (common)" is visible when enabled.

---Parameter "Send delay for the next request cycle"

This parameter is visible when gateway type is selected as "Modbus slave" and the previous parameter enabled.

This parameter is for setting the communication status of the sending between master and the slave. For the master, a ON telegram will be sent to the KNX bus when without receiving a response from the device. For slave, within a request timeout, a ON telegram is sent to the KNX bus when without a request from the host. Options:

10min

20min

...

120min

RS485 / RS232 setting(The following parameters are visible when channel function is selected as "RS485/RS232<->KNX")

RS485/RS232 telegram is HEX Code, support max. size is 16 bytes, and format as: 0A-0F-AB-....; or 0A 0F AB ...; or 0A,0F,BA,...;

Parameter "Send delay between RS485/RS232 telegram"

This parameter is for setting the send delay between RS485/RS232 telegram. When set the "Disable", RS485/RS232 telegram is sent without delay. Options:

Disable

50ms

...

500ms

5.3.Parameter window “Modbus<->KNX”

5.3.1. Parameter window “Datapoints setting”

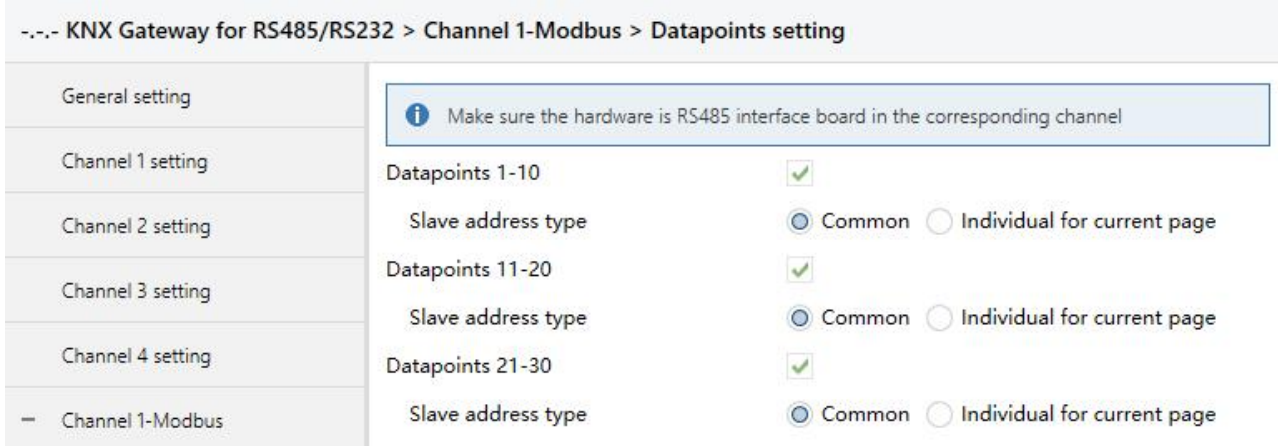


Fig.5.3.1 “Datapoints setting” parameter window

Make sure the hardware is RS485 interface board in the corresponding channel

Parameter "Datapoints 1-10/11-20/21-30/...”

This parameter is for setting the datapoint window of Modbus<->KNX. Display 10 datapoints per page when enabled.

Note: Modbus<->KNX channel 1 supports most 500 datapoints and channel 2/3/4 most 50 datapoints.

——Parameter "Slave address type"

This parameter is visible when gateway type is selected as “Modbus master” and the previous parameter enabled. Set slave address type per page. Options:

Common

Individual for current page

The following parameters are visible when select “Individual for current page”

——Parameter "Slave address"

This parameter is for setting slave address of the current page when the page uses an individual slave address. For example, the slave address of datapoint 1-10. Option: **0..247**

——Parameter "Slave description (max 30char.)"

This parameter is for setting the custom description of diagnostic object, up to 30 characters can be input.

Each 10 datapoint with a diagnostic object, such as the object “Diagnostic: Slave (Data point 1-10)”.

5.3.2. Parameter window “Datapoint x”

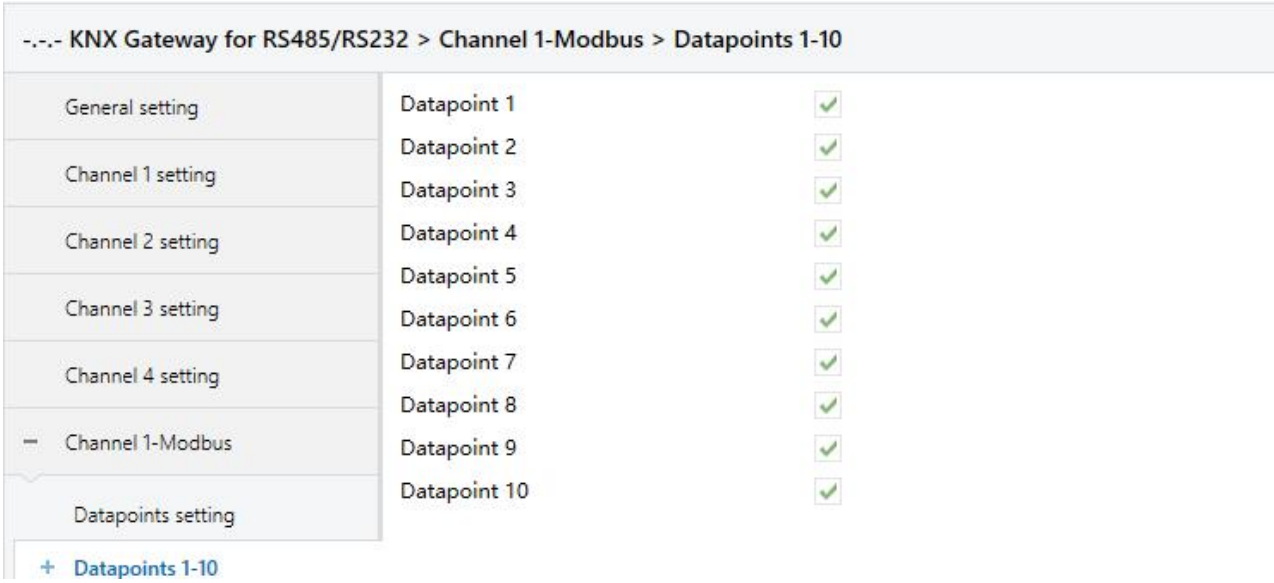
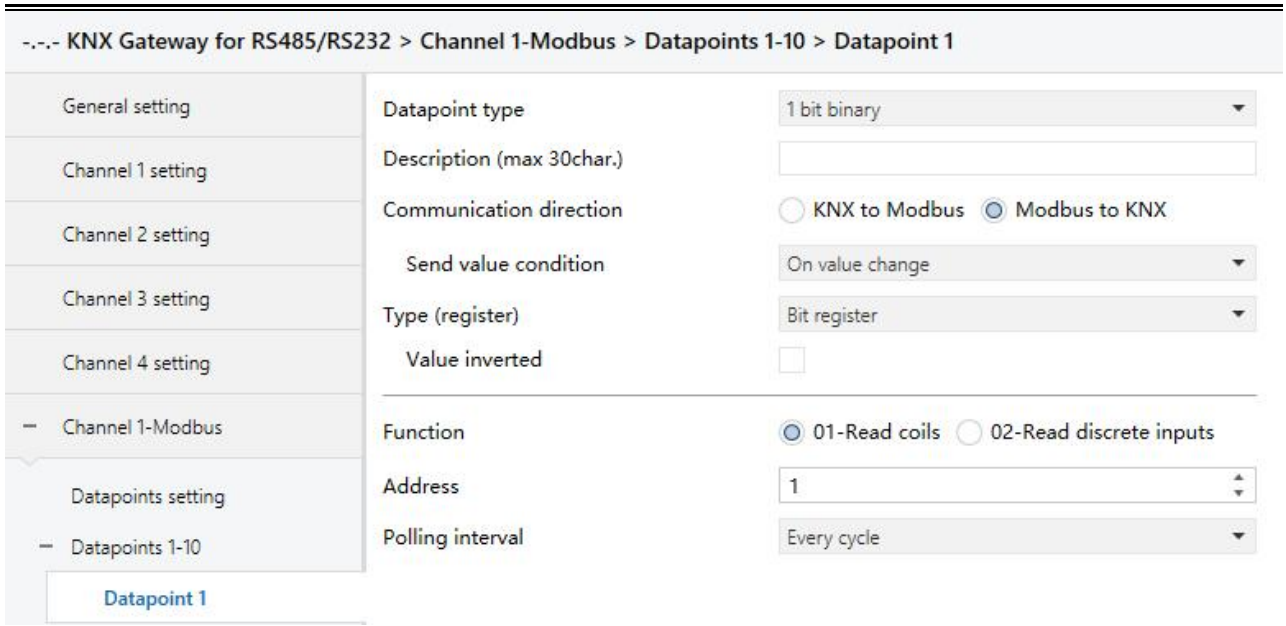


Fig.5.3.2.1 “Datapoints 1-10” parameter window

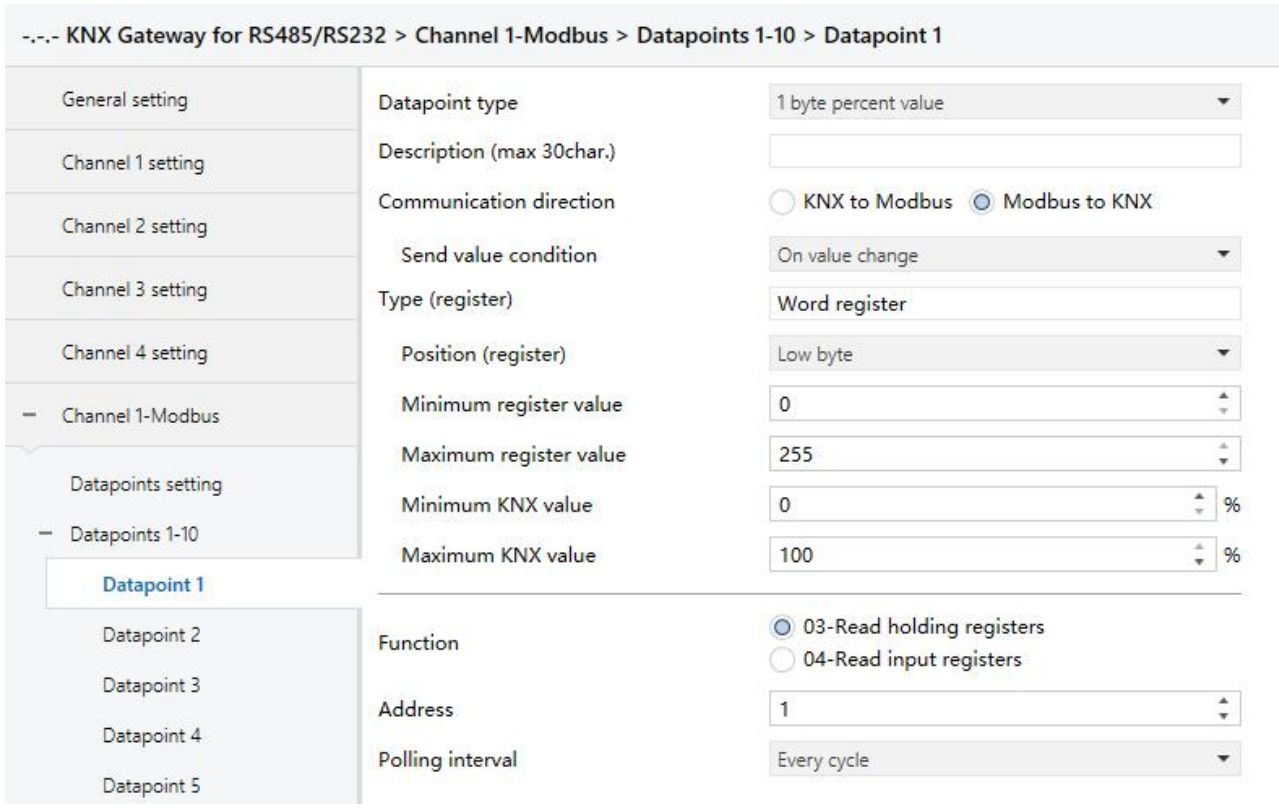
Parameter "Datapoints 1/2/3/..."

This parameter is for setting the datapoint of Modbus<->KNX. Display datapoint when enabled.

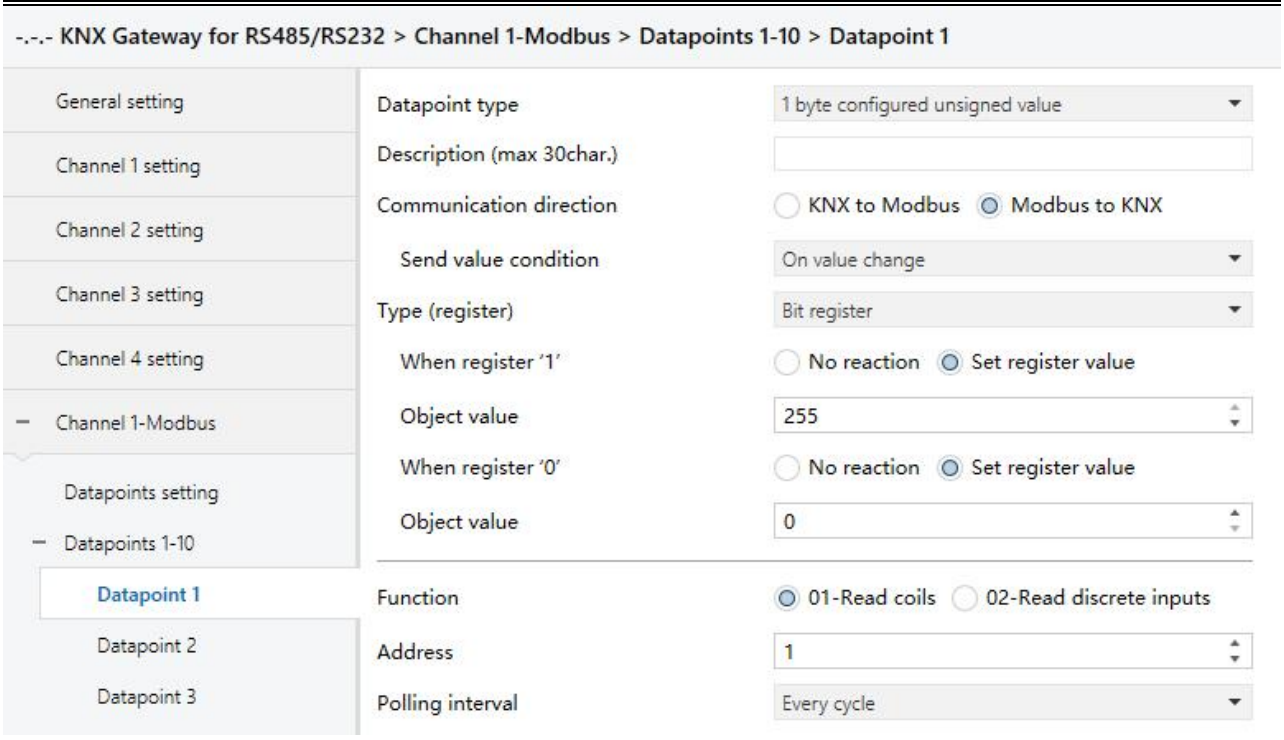
The following is the datapoint parameter window for the Modbus <-> KNX.



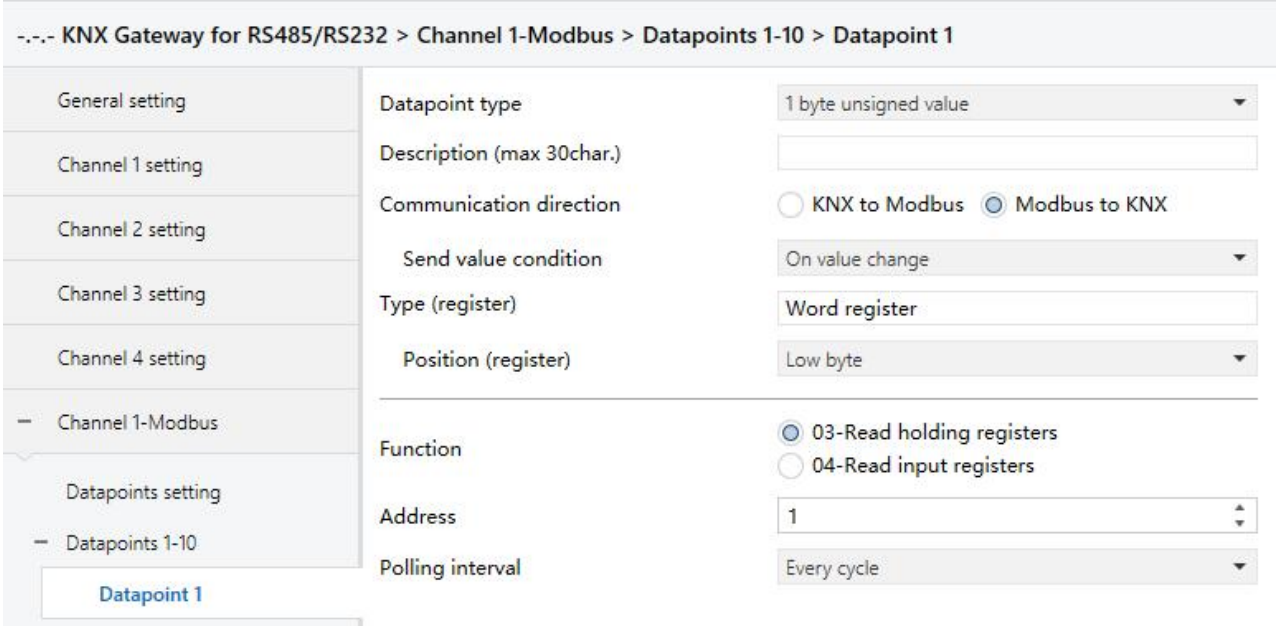
"1 bit binary" parameter



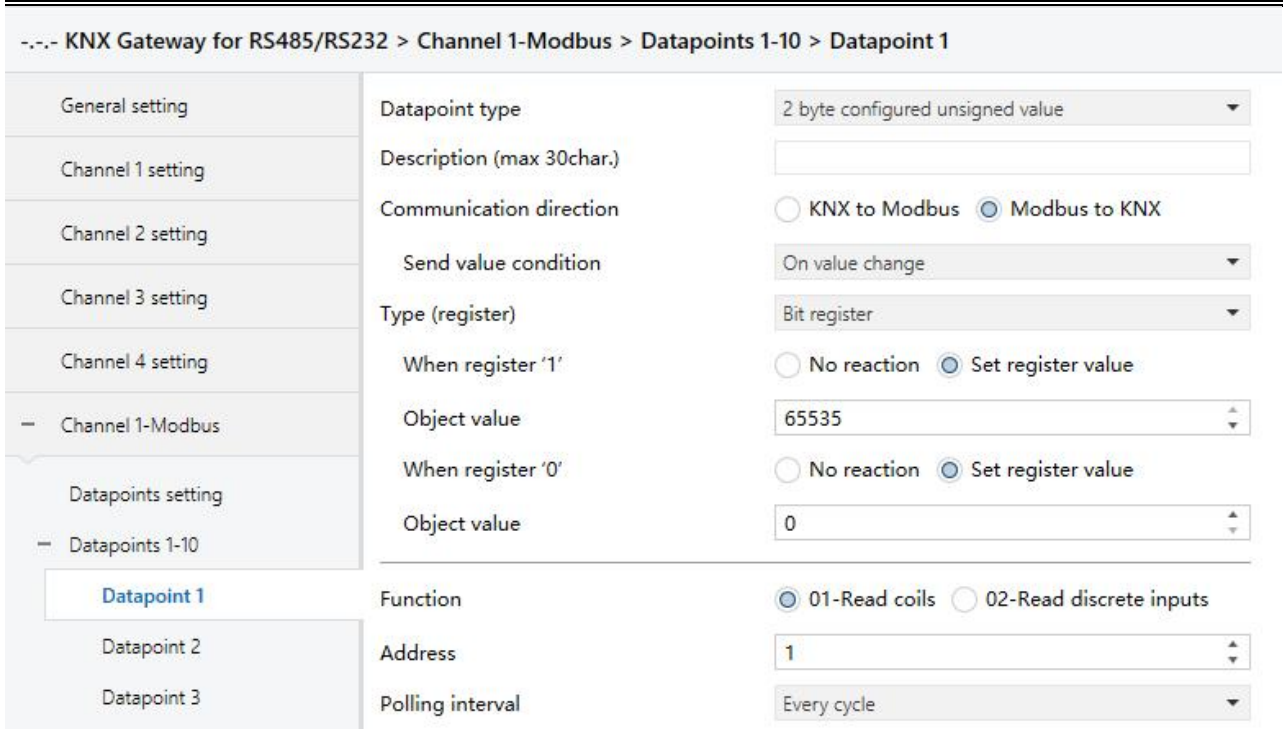
"1 byte percent value" parameter



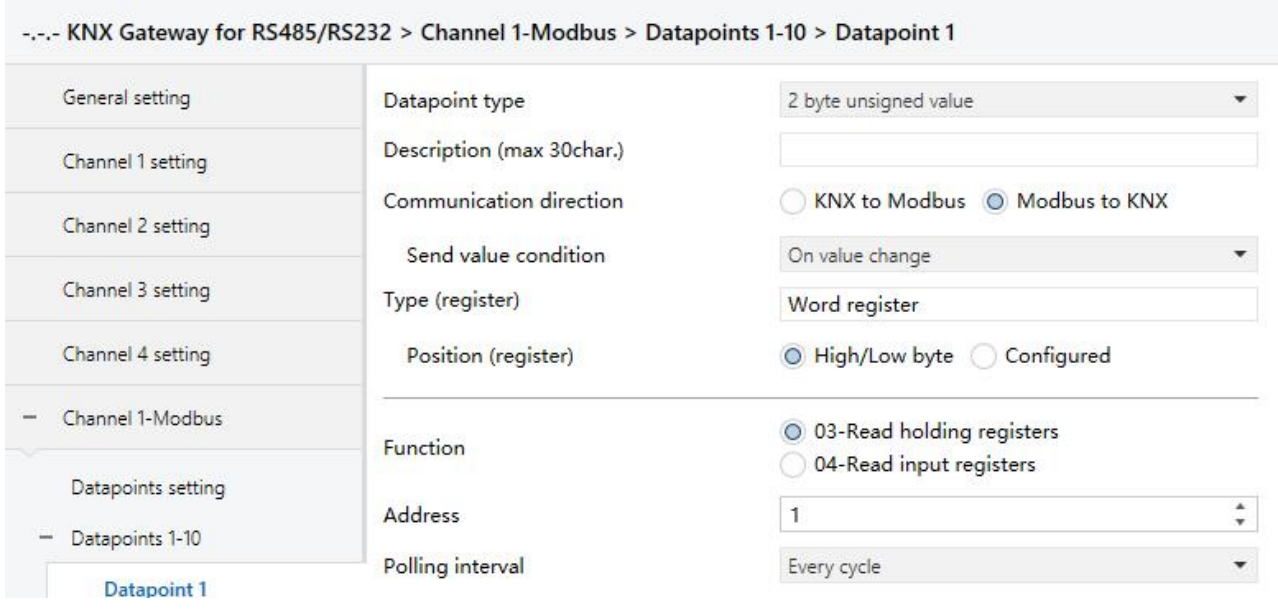
"1 byte configured unsigned value" parameter



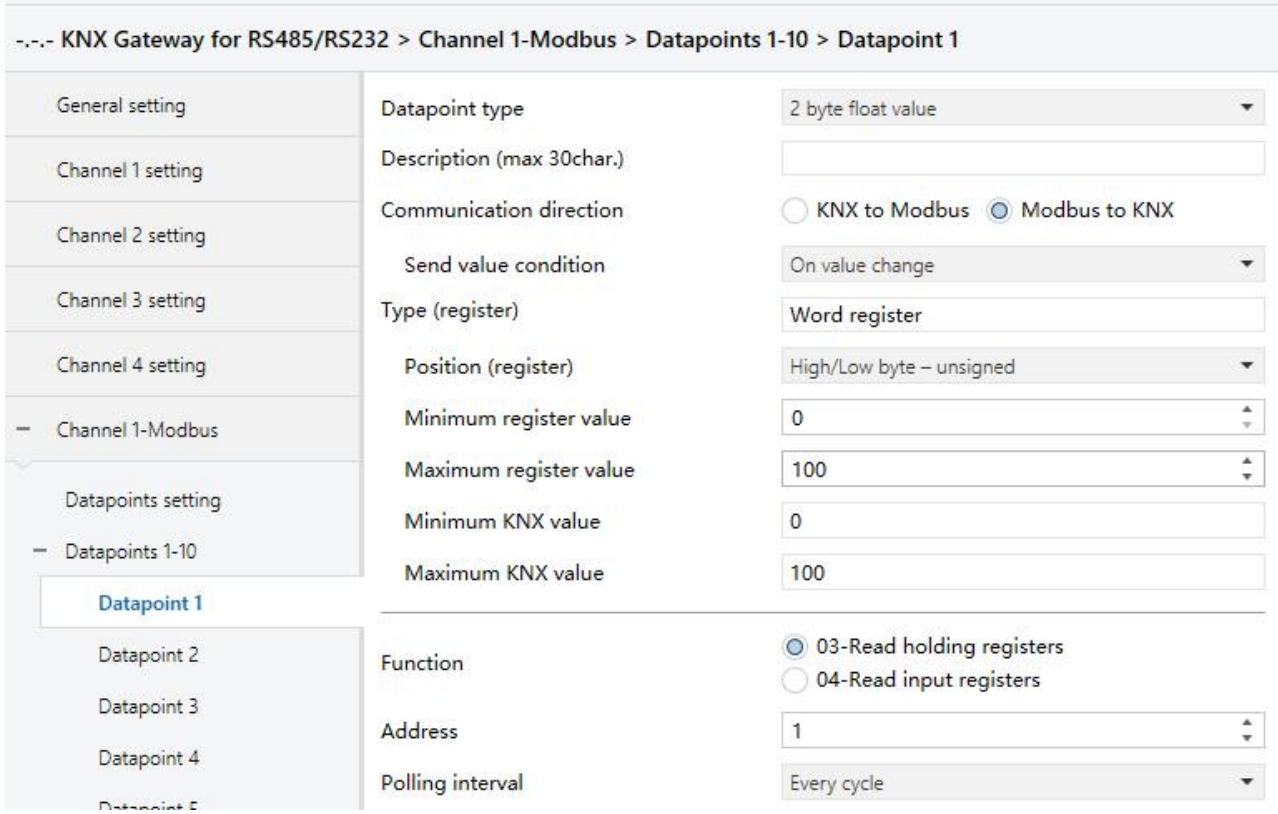
"1 byte unsigned value" parameter



"2 byte configured unsigned value" parameter



"2 byte unsigned value" parameter



"2 byte float value" parameter

Fig.5.3.2.2 "Datapoints x" parameter window

Parameter "Datapoint type"

This parameter is for setting the datapoint type. Options:

1 bit binary

1 byte percent value

1 byte configured unsigned value

1 byte unsigned value

2 byte configured unsigned value

2 byte unsigned value

2 byte float value

Parameter "Description (max 30char.)"

This parameter is for setting the custom description of datapoint, up to 30 characters can be input.

Parameter "Communication direction"

This parameter is for setting the communication direction. Options:

KNX to Modbus

Modbus to KNX

---Parameter "Send value condition"

This parameter is visible when previous parameter is selected as "Modbus to KNX". Set the conditions for the value sending. Options:

Only on read requests

On value change

Cyclic sends

Cyclic sends and on value change

---Parameter "Cycle time [1..255]min"

This parameter is visible when previous parameter is selected as "Cyclic sends...". Set the cycle time. Options: **1..255**

5.3.2.1. Datapoint type "1 bit binary"

Parameter "Type (register)"

This parameter defines the channel function and the size of the register used. Options:

Bit register

Bit in word register

Value in word register

---Parameter "Value inverted"

This parameter is visible when register type is selected as "Bit register" or "Bit in word register". Value inverted when enabled. If inverted, then inversion value of the group object corresponds to the value of the bit register.

---Parameter "Position (register)"

This parameter is visible when register type is selected as "Bit in word register". Define position in the word register. Options:

Bit 00

Bit 01

...

Bit 15

---Parameter "Bit count"

This parameter is visible when register type is selected as "Value in word register". Define bit count in the word register. Options:

16 bit

15 bit

...

01 bit

---Parameter "Offset"

This parameter is visible when register type is selected as "Value in word register". Define offset in the word register. Options:

Bit 00

Bit 01

...

Bit 15

Note: The channel will not work if it is misconfigured for the option "Value in word register".

The number of bit count and offset must not exceed 16bit, value needs to match the bit count. For example, the bit count is 1bit, so the value is only 0 and 1.

If the bit count is 08bit, the offset must not exceed 08bit.

Eg: If bit count is 08bit,Offset is 08bit, then the value actually defined is the following range (bold font):

B15 B14 B13 B12 B11 B10 B9 B8 B7 B6 B5 B4 B3 B2 B1 B0

It is seen from the example that the bit count and offset must not exceed 16bit. If exceeded, it is out of range and this channel fails.

The following parameters are visible when select "Modbus to KNX"

---Parameter "Checked value (register)"

This parameter is visible when register type is selected as "Value in word register". Define checked value in the word register via the number of bit count and offset. Options: **0..65535**

---Parameter "When register value>checked value, object"

---Parameter "When register value=checked value, object"

---Parameter "When register value<checked value, object"

These parameters are visible when register type is selected as "Value in word register". Set action of group object. When the register value is greater than/equal to/less than checked value, send a

ON/OFF telegram to the bus, or not. Options:

No reaction

Send 'ON'

Send 'OFF'

The following parameters are visible when select "KNX to Modbus"

——Parameter "When object receiving value=ON"

——Parameter "When object receiving value=OFF"

These two parameters are visible when register type is selected as "Value in word register". Set whether to send a value to the register when the object receives a ON/OFF telegram. Options:

No reaction

Set register value

——Parameter "Register value"

This parameters is visible when register type is selected as "Value in word register" and the previous parameter selected "Set register value". Set register value when the object receives a ON/OFF telegram **(Note the range of the definable values)**.

Options: **0..65535**

5.3.2.2. Datapoint type “1 byte percent value”

Parameter “Type (register)”

This parameter defines the channel function and the size of the register used.

Read-only by default **Word register**.

Parameter “Position (register)”

This parameter defines position mapped to the word register. Options:

Low byte

High byte

High/Low byte

—Parameter “Minimum register value”

—Parameter “Maximum register value”

These two parameters is for setting correspond to the register value of the KNX minimum or maximum value.

When select “Low byte” or “High byte”, options: **0..255**

When select “Low byte/High byte”, options: **0..65535**

—Parameter “Minimum KNX value”

—Parameter “Maximum KNX value”

These two parameters is for setting correspond to the KNX value of the register minimum or maximum value. Options: **0..100 (%)**

Note: Realize the mapping of the KNX percentage value to the value in the word register. The conversion is always transferred to the entire register area and there is no limit definition on the minimum and maximum of the register.

The proportional coefficient can be calculated based on the maximum/minimum values of the register and KNX.

Eg:

Value minimum (register) = 0

Value maximum (register) = 100

Value minimum (KNX) = 0

Value maximum (KNX) = 10

The proportional coefficient is 10, and the value maps as follows:

Value KNX = 10.5 -->Value Register = 105

5.3.2.3. Datapoint type “1 byte configured unsigned value”

Parameter “Type (register)”

This parameter defines the channel function and the size of the register used. Options:

Bit register

Bit in word register

Value in word register

——Parameter “Position (register)”

This parameter is visible when register type is selected as “Bit in word register”. Define position in the word register. Options:

Bit 00

Bit 01

...

Bit 15

——Parameter “Bit count”

This parameter is visible when register type is selected as “Value in word register”. Define bit count in the word register. Options:

16 bit

15 bit

...

01 bit

——Parameter “Offset”

This parameter is visible when register type is selected as “Value in word register”. Define offset in the word register. Options:

Bit 00

Bit 01

...

Bit 15

The following parameters are visible when select “Modbus to KNX”

——Parameter “When register ‘1’”

——Parameter “When register ‘0’”

These two parameters are visible when register type is selected as “Bit register” or “Bit in word register”. Set whether to send a value to the register when register value is 1/0. Options:

No reaction

Set register value

--Parameter "Object value"

This parameters is visible when register type is selected as "Bit register" or "Bit in word register" and the previous parameter selected "Set register value". Set object value when the register value is 1/0.

Options: **0..255**

--Parameter "Checked value (register)"

This parameter is visible when register type is selected as "Value in word register". Define checked value in the register via the number of bit count and offset. Options: **0..65535**

--Parameter "When register value>checked value"

--Parameter "When register value=checked value"

--Parameter "When register value<checked value"

These parameters are visible when register type is selected as "Value in word register". Set whether to send a value to the register when the register value is greater than/equal to/less than the checked value. Options:

No reaction

Set register value

--Parameter "Object value"

This parameters is visible when register type is selected as "Value in word register" and the previous parameter selected "Set register value". Set object value when the value of the register is greater than/equal to/less than the value defined via parameter. Options: **0..255**

The following parameters are visible when select "KNX to Modbus"

--Parameter "Checked value (register)"

This parameter is visible when register type is selected as "Bit in word register" or "Value in word register". Define checked value in the register via the number of bit count and offset. Options: **0..255**

--Parameter "When register value>checked value, object"

--Parameter "When register value=checked value, object"

--Parameter "When register value<checked value, object"

These parameters are visible when register type is selected as "Bit register" or "Bit in word register". Set action of group object. When the register value is greater than/equal to/less than the checked value, send a 1/0 telegram to the bus, or not. Options:

No reaction

Set register '1'

Set register '0'

---Parameter "When register value>checked value, object"

---Parameter "When register value=checked value, object"

---Parameter "When register value<checked value, object"

These parameters are visible when register type is selected as "Value in word register". Set action of group object. When the register value is greater than/equal to/less than the checked value, set whether to send a value to the register. Options:

No reaction

Set register value

---Parameter "Register value"

This parameters is visible when register type is selected as "Value in word register" and the previous parameter selected "Set register value". Set register value. Options: **0..65535**

5.3.2.4. Datapoint type "1 byte unsigned value"

Parameter "Type (register)"

This parameter defines the channel function and the size of the register used.

Read-only by default **Word register**.

Parameter "Position (register)"

This parameter defines position mapped to the word register. Options:

Low byte

High byte

Configured

---Parameter "Bit count"

This parameters is visible when selects "Configured". Define bit count in the word register. Options:

08 bit

07bit

...

01 bit

---Parameter "Offset"

This parameters is visible when selects "Configured". Define offset in the word register. Options:

Bit 00

Bit 01

...

Bit 15

5.3.2.5. Datapoint type "2 byte configured unsigned value"

Parameter "Type (register)"

This parameter defines the channel function and the size of the register used. Options:

Bit register

Bit in word register

Value in word register

---Parameter "Position (register)"

This parameter is visible when register type is selected as "Bit in word register". Define position in the word register. Options:

Bit 00

Bit 01

...

Bit 15

---Parameter "Bit count"

This parameter is visible when register type is selected as "Value in word register". Define bit count in the word register. Options:

16 bit

15 bit

...

01 bit

---Parameter "Offset"

This parameter is visible when register type is selected as "Value in word register". Define offset in the word register. Options:

Bit 00

Bit 01

...

Bit 15

The following parameters are visible when select "Modbus to KNX"

—Parameter "When register '1'"

—Parameter "When register '0'"

These two parameters are visible when register type is selected as "Bit register" or "Bit in word register". Set whether to send a value to the register when register value is 1/0. Options:

No reaction

Set register value

—Parameter "Object value"

This parameters is visible when register type is selected as "Bit register" or "Bit in word register" and the previous parameter selected "Set register value". Set object value when the register value is 1/0.

Options: **0..65535**

—Parameter "Checked value (register)"

This parameter is visible when register type is selected as "Value in word register". Define checked value in the register via the number of bit count and offset. Options: **0..65535**

—Parameter "When register value>checked value"

—Parameter "When register value=checked value"

—Parameter "When register value<checked value"

These parameters are visible when register type is selected as "Value in word register". Set whether to send a value to the register when the register value is greater than/equal to/less than the checked value. Options:

No reaction

Set register value

—Parameter "Object value"

This parameters is visible when register type is selected as "Value in word register" and the previous parameter selected "Set register value". Set object value when the value of the register is greater than/equal to/less than the value defined via parameter. Options: **0..65535**

The following parameters are visible when select "KNX to Modbus"

—Parameter "Checked value (register)"

This parameter defines checked value in the register via the number of bit count and offset.

Options: **0..65535**

—Parameter "When register value>checked value, object"

—Parameter "When register value=checked value, object"

—Parameter "When register value<checked value, object"

These parameters are visible when register type is selected as "Bit register" or "Bit in word register". Set action of group object. When register value is greater than/equal to/less than the check value, send a 1/0 telegram to the bus, or not. Options:

No reaction

Set register '1'

Set register '0'

—Parameter "When register value>checked value"

—Parameter "When register value=checked value"

—Parameter "When register value<checked value"

These parameters are visible when register type is selected as "Value in word register". Set whether to send a value to the register when the register value is greater than/equal to/less than the checked value. Options:

No reaction

Set register value

—Parameter "Object value"

This parameters is visible when register type is selected as "Value in word register" and the previous parameter selected "Set register value". Set Object value when the register value is greater than/equal to/less than the checked value. Options: **0..65535**

5.3.2.6. Datapoint type “2 byte unsigned value”

Parameter “Type (register)”

This parameter defines the channel function and the size of the register used.

Read-only by default **Word register**.

Parameter “Position (register)”

This parameter defines position mapped to the word register. Options:

Low/High byte

Configured

——Parameter “Bit count”

This parameters is visible when selects “Configured”. Define bit count in the word register. Options:

16 bit

15 bit

...

01 bit

——Parameter “Offset”

This parameters is visible when selects “Configured”. Define offset in the word register. Options:

Bit 00

Bit 01

...

Bit 15

5.3.2.7. Datapoint type “2 byte float value”

Parameter “Type (register)”

This parameter defines the channel function and the size of the register used.

Read-only by default **Word register**.

Parameter “Position (register)”

This parameter defines position mapped to the word register. Options:

Low byte – unsigned

High byte – unsigned

High/Low byte – unsigned

Low byte – 2th complement

High byte – 2th complement

High/Low byte – 2th complement

—Parameter “Minimum register value”

—Parameter “Maximum register value”

These two parameters is for setting correspond to the register value of the KNX minimum or maximum value.

When select “Low byte – unsigned” or “High byte – unsigned”, options: **0..255**

When select “High/Low byte – unsigned”, options: **0..65535**

When select “Low byte – 2th complement” or “High byte – 2th complement”, options: **-128..127**

When select “High/Low byte – 2th complement”, options: **-32768..32767**

—Parameter “Minimum KNX value”

—Parameter “Maximum KNX value”

These two parameters is for setting correspond to the KNX value of the register minimum or maximum value. Options: **-671088.00 ...670760.00**

5.3.2.8. Register function

——Parameter "Function"

This parameter is for setting the Modbus function code of this channel. Different function codes can be configured depending on KNX gateway types (Modbus master/slave), communication direction, and register types.

The following bit register configurations are visible when select "Bit register":

When select "Modbus Master" and "Modbus to KNX", or "Modbus Slave" and "KNX to Modbus", options:

01-Read coils

02-Read discrete inputs

When select "Modbus Master" and "KNX to Modbus", Read-only by default **05-Write single coil**

When select "Modbus Slave" and "Modbus to KNX", Read-only by default **05,15-Write single/multi coils**

The following word register configurations are visible when select "word register" or "Bit in word register" or "Value in word register":

When select "Modbus Master" and "Modbus to KNX", or "Modbus Slave" and "KNX to Modbus", options:

03-Read holding registers

04-Read input registers

When select "Modbus Master" and "KNX to Modbus", Read-only by default **06-Write single holding registers**

When select "Modbus Slave" and "Modbus to KNX", Read-only by default **06,16-Write single/multi holding registers**

Parameter "Address"

This parameter is for setting the address of Modbus register. Options: **0..65535**

If the address is "based 1", then it will not be configured here as 0, or an error appears and the channel function is disabled.

Parameter "Polling interval"

This parameter is visible when select "Modbus Master" and "Modbus to KNX". Define polling interval which each register sends read requests. Options:

Every cycle

Every second cycle

Every fourth cycle

Every sixth cycle

Every eighth cycle

5.4. Parameter window "RS485/RS232<->KNX"

5.4.1. Parameter window "Datapoints setting"

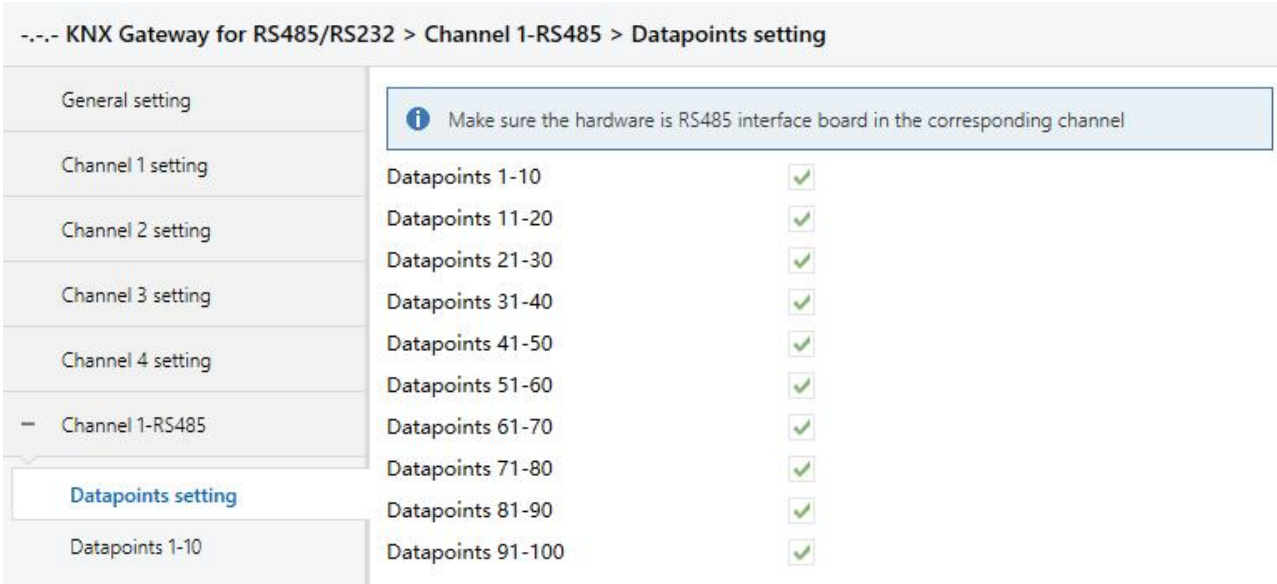


Fig.5.4.1 "Datapoints setting" parameter window

Make sure the hardware is RS485/RS232 interface board in the corresponding channel

Parameter "Datapoints 1-10/11-20/21-30/..."

This parameter is for setting the datapoint window of RS485/RS232<->KNX. Display 10 datapoints per page when enabled.

Note: RS485/RS232<->KNX channel 1 supports most 100 datapoints and channel 2/3/4 most 50 datapoints.

5.4.2. Parameter window “Datapoint x”

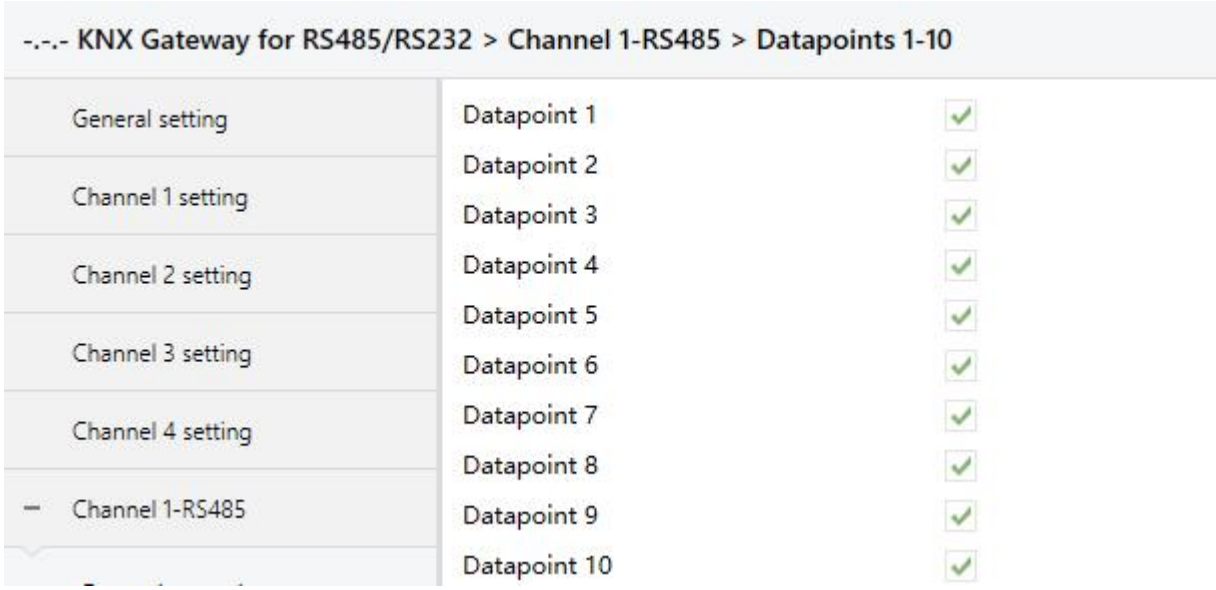


Fig.5.4.2.1 “Datapoints 1-10” parameter window

Parameter "Datapoints 1/2/3/..."

This parameter is for setting the datapoint of RS48/RS232<->KNX. Display datapoint when enabled.

The following is the datapoint parameter window for the RS48/RS232<->KNX.

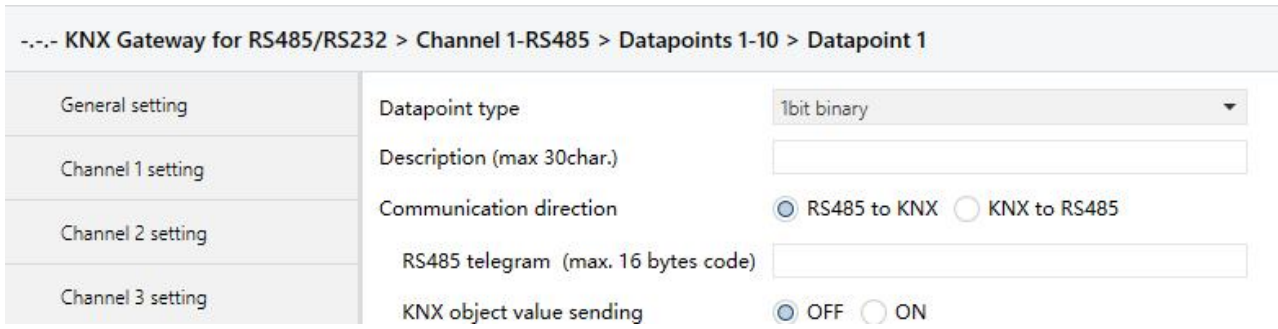


Fig.5.4.2.1 “Datapoints x” parameter window

Parameter "Datapoint type"

This parameter is for setting the datapoint type. Options:

1 bit binary

1 byte unsigned value

1byte recall scene

2 byte unsigned value

Parameter "Description (max 30char.)"

This parameter is for setting the custom description of datapoint, up to 30 characters can be input.

Parameter "Communication direction"

This parameter is for setting the communication direction. Options:

RS485/RS232 to KNX

KNX to RS485/RS232

——Parameter "RS485/RS232 telegram (max. 16 bytes code)"

This parameter is for setting the telegram of RS485/RS232, up to 16 characters can be input.

Note: RS485/RS232 telegram is HEX Code, support max. size is 16 bytes, and format as: 0A-0F-AB-....; or 0A 0F AB ...; or 0A,0F,BA,...;

——Parameter "KNX object value sending"

This parameter is visible when select "RS485/RS232 to KNX". Set the sending value of KNX object via datapoint type.

When select "1 bit binary", options:

OFF

ON

When select "1 byte unsigned value", options: **0..255**

When select "1byte recall scene", options:

Scene No.1

Scene No.2

...

Scene No.64

When select "2byte unsigned value", options: **0..65535**

——Parameter "KNX object value receiving"

This parameter is visible when select "RS485/RS232 to KNX". Set the receiving value of KNX object via datapoint type.

When select "1 bit binary", options:

OFF

ON

When select "1 byte unsigned value", options: **0..255**

When select “1byte recall scene”, options:

Scene No.1

Scene No.2

...

Scene No.64

When select “2byte unsigned value”, options: **0..65535**

5.5. Parameter window“VRV/VRF gateway”

5.5.1. Parameter window“VRV/VRF setting”

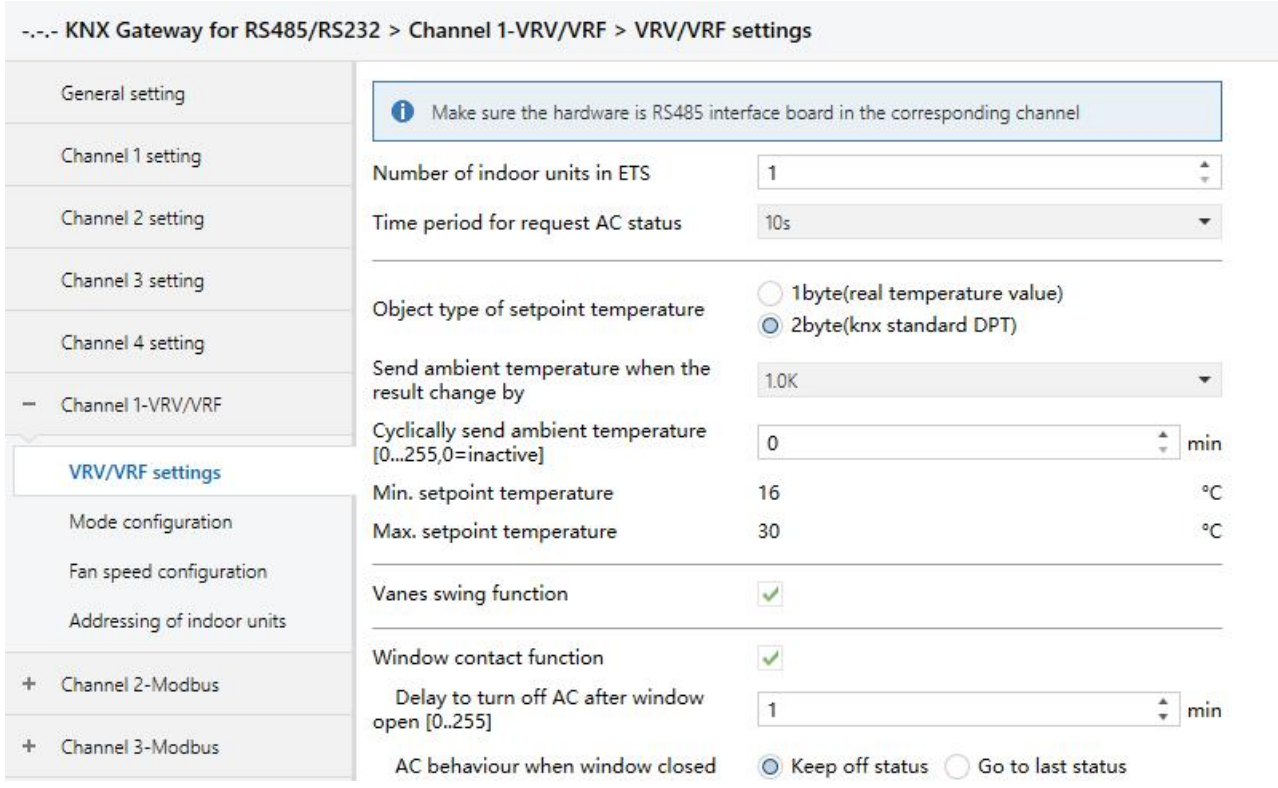


Fig.5.5.1 “VRV/VRF setting” parameter window

Make sure the hardware is RS485 interface board in the corresponding channel

Parameter "Number of indoor units in ETS"

This parameter is for setting the number of indoor units, and Air-conditioning objects and addresses will be displayed according to the amount. Options: **1..64**

Parameter "Time period for request AC status"

This parameter is for setting the time period for read request AC status, and need to read the

status when the gateway power on. Options: **Disable / 1s / 2s / 3s / 5s / 10s / 15s / 20s / 25s / 30s / 60s**

Parameter "Object type of setpoint temperature"

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

1byte(real temperature value)

2byte(knx standard DPT)

Parameter "Send ambient temperature when the result change by"

This parameter is for setting the temperature difference to send to bus. Not send when disable. Options:

Disable
0.5K
1.0K
...
10K

Note: The ambient temperature in this case is the temperature of the air condition vent.

Parameter "Cyclically send ambient temperature [0...255,0=inactive] min"

Setting the time for cyclically sending the ambient temperature value to the bus. Options: **0..255**

This period is independent and starts time counting after programming completion or reset. Transmission change has no affect on this period.

Parameter "Min. / Max. setpoint temperature"

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

Options are only read by default, and display different ranges of setpoint temperature according to the AC protocols.

Parameter "Vanes swing function"

This parameter is visible when AC protocol is selected "DAIKIN (DTA116A621)", "Toshiba" or "Other". Set whether to enable the vanes swing and stop.

Parameter "Vanes Up-Down swing function"

This parameter is visible when AC protocol is selected "Fujitsu". Set whether to enable the vanes Up-Down swing.

Parameter "Vanes Left-Right swing function"

This parameter is visible when AC protocol is selected "Fujitsu". Set whether to enable the vanes

Left-Right swing.

Parameter "Window contact function"

This parameter is for setting whether to connect window contact function.

——Parameter "Delay to turn off AC after window open [0..255]min"

This parameter is visible when Window contact function enabled. Set the delay of turning off Air-conditioning after window open. Options: **0..255**

When value=0, turn off immediately. When the window open and Air-conditioning on again, the timing is reset then off.

——Parameter "AC behaviour when window closed"

This parameter is visible when Window contact function enabled. After the window closed, setting the Air-conditioning status whether to return previous status or stay off. Options:

Keep off status

Go to last status

5.5.2. Parameter window "Mode configuration"

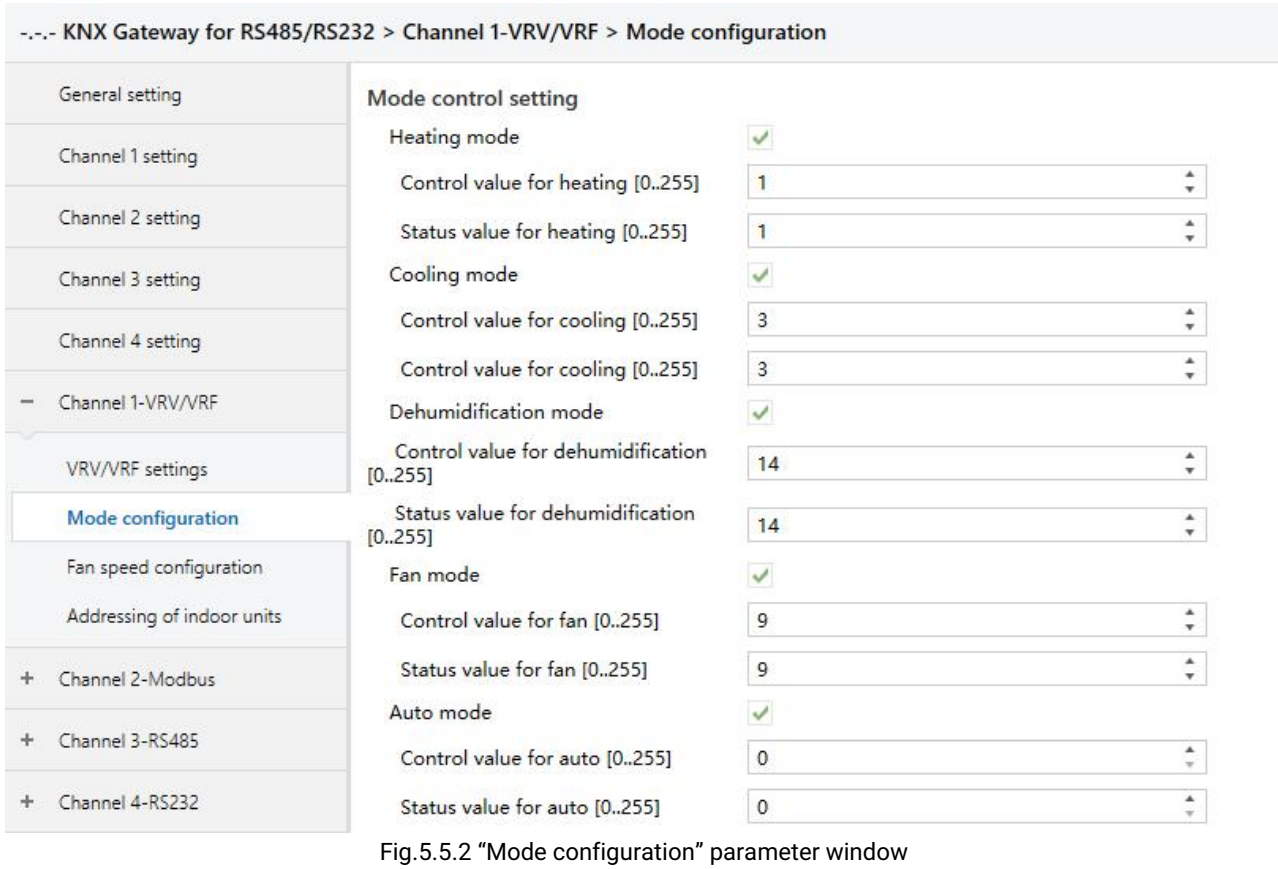


Fig.5.5.2 "Mode configuration" parameter window

Mode control setting

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

These parameters are for setting whether to enable corresponding control mode.

Auto mode and and its setting parameters are visible when AC protocol is not selected "HITACHI".

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

These parameters are visible when mode enabled. Set to switch to the each mode control value.

Options: **0..255**

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

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

Options: **0..255**

5.5.3. Parameter window "Fan speed configuration"

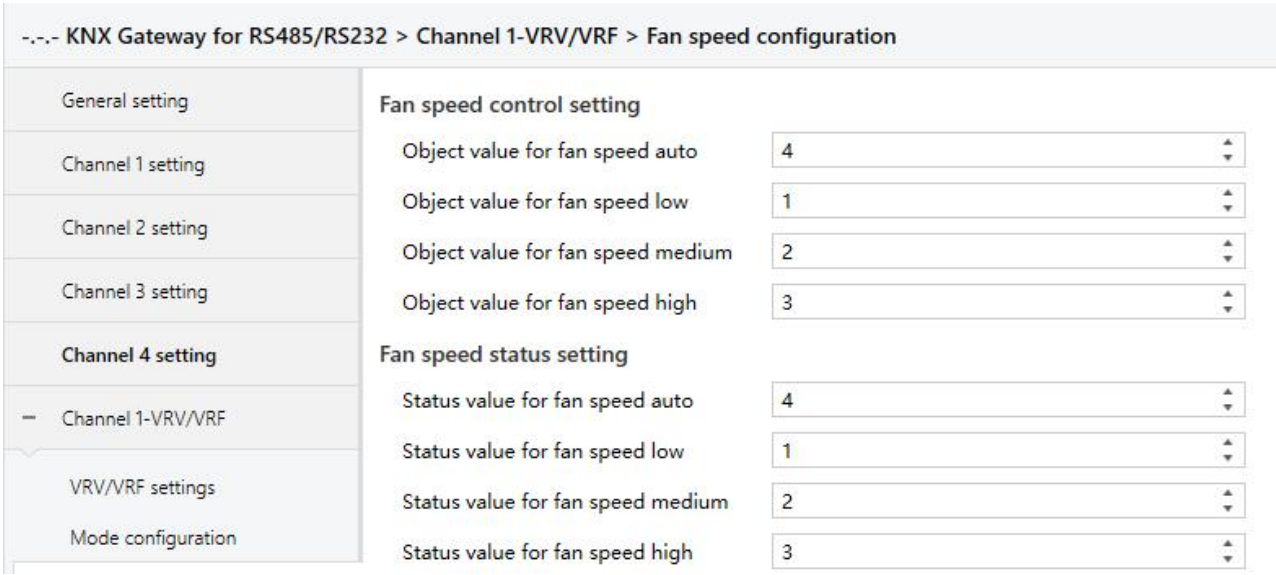


Fig.5.5.3 "Fan speed configuration" parameter window

Fan speed control setting

Parameter "Object value for fan speed auto/low/medium/high/super high"

These parameters are for setting to switch to object value sent by each fan speed. Options: **0..255**

Fan speed status setting

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

These parameters are for setting to switch to status feedback value of each fan speed. Options: **0..255**

The setting parameters of auto fan speed are visible when AC protocol is selected “DAIKIN (DTA116A621)”, “Fujitsu”, “Gree”, “Toshiba” or “Other”.

The setting parameters of super high fan speed are visible when AC protocol is selected “Mitsubishi”.

5.5.4. Parameter window “Addressing of indoor units”

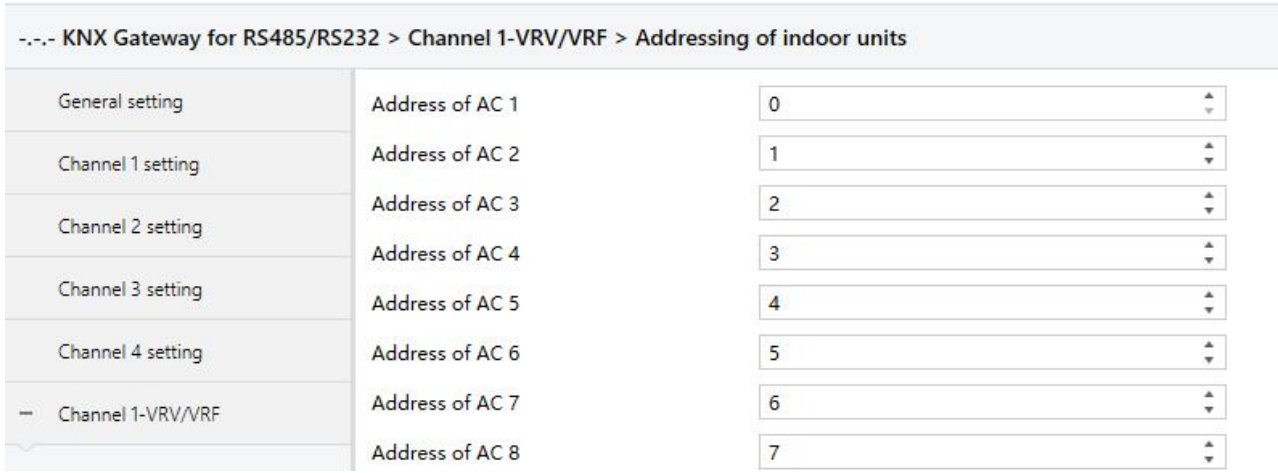


Fig.5.5.4(1) “Addressing of indoor units” parameter window

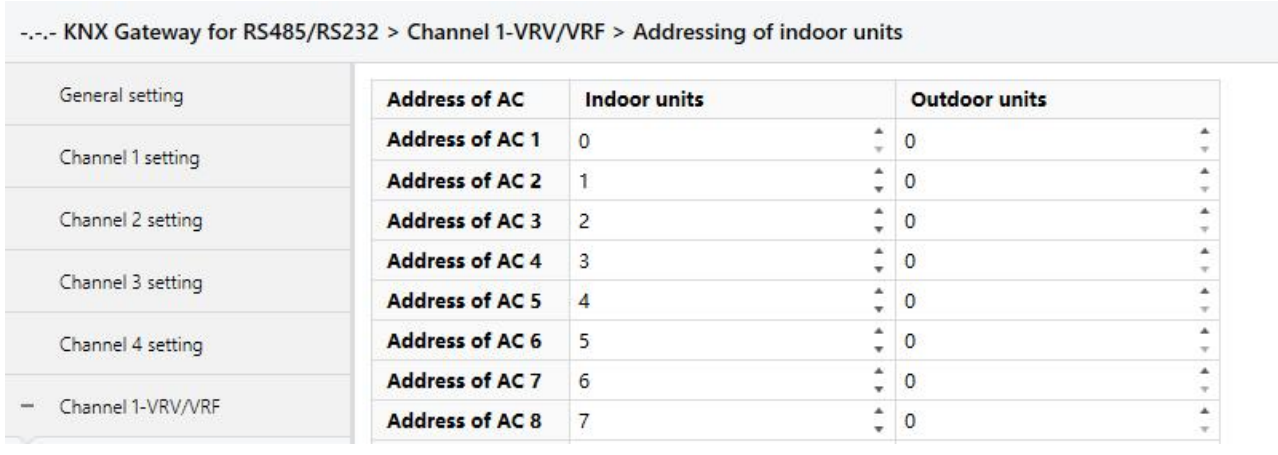


Fig.5.5.4 (2) “Addressing of indoor units” parameter window

Parameter “Address of AC x”(x=64)

This parameter displays address of Air-conditioning according to the number of indoor units. Range of options is according to the address.

Only the address of the indoor units can be set when AC protocol is not selected “HITACHI”, shown as Fig.5.5.4 (1).

The address of the indoor and outdoor units can be set when AC protocol is selected “HITACHI”, shown as Fig.5.5.4 (2).

Please note the address settings of DAIKIN:

AC1~AC16 corresponds to the address of the indoor unit 0~15, AC17~AC32 corresponds to the address of the indoor unit 0~15, AC33~AC48 corresponds to the address of the indoor unit 0~15, AC49~AC64 corresponds to the address of the indoor unit 0~15, and the parameters are used to set the slave address.

For example:

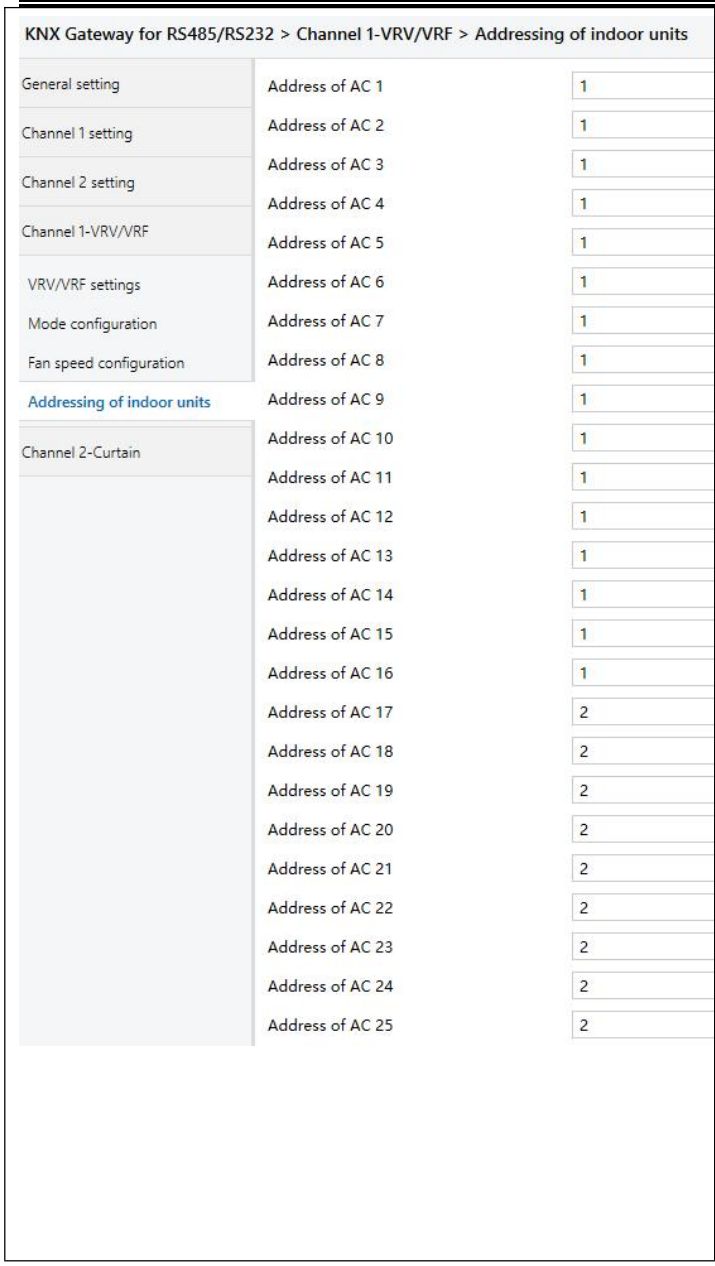
If parameter options of AC1~AC16 are set to 2, the slave address corresponding to AC1~AC16 is 2. The encoding of address of indoor units according to DAIKIN protocol is x-xx 

, 2-01 indicates the slave address is 2 and indoor unit address is 01.

Example of parameters configuration:

The first 16 AC addresses indicate that the slave address is 1 and the indoor unit addresses are 0~15, the other parameters setting is the similar to this.

Parameter setting	The corresponding register mapping table in the protocol (The first address is 4 in the registration address indicates the indoor unit control, 3 is the indoor unit status, and it is control as following figure)



3-2-2.Indoor Unit control

Address	Address of Indoor unit
42001-42003	1-00
42004-42006	1-01
42007-42009	1-02
42010-42012	1-03
42013-42015	1-04
42016-42018	1-05
42019-42021	1-06
42022-42024	1-07
42025-42027	1-08
42028-42030	1-09
42031-42033	1-10
42034-42036	1-11
42037-42039	1-12
42040-42042	1-13
42043-42045	1-14
42046-42048	1-15
42049-42051	2-00
42052-42054	2-01
42055-42057	2-02
42058-42060	2-03
42061-42063	2-04
42064-42066	2-05
42067-42069	2-06
42070-42072	2-07
42073-42075	2-08
42076-42078	2-09

5.6.Parameter window “Curtain”

5.6.1. Parameter window “Curtain setting”

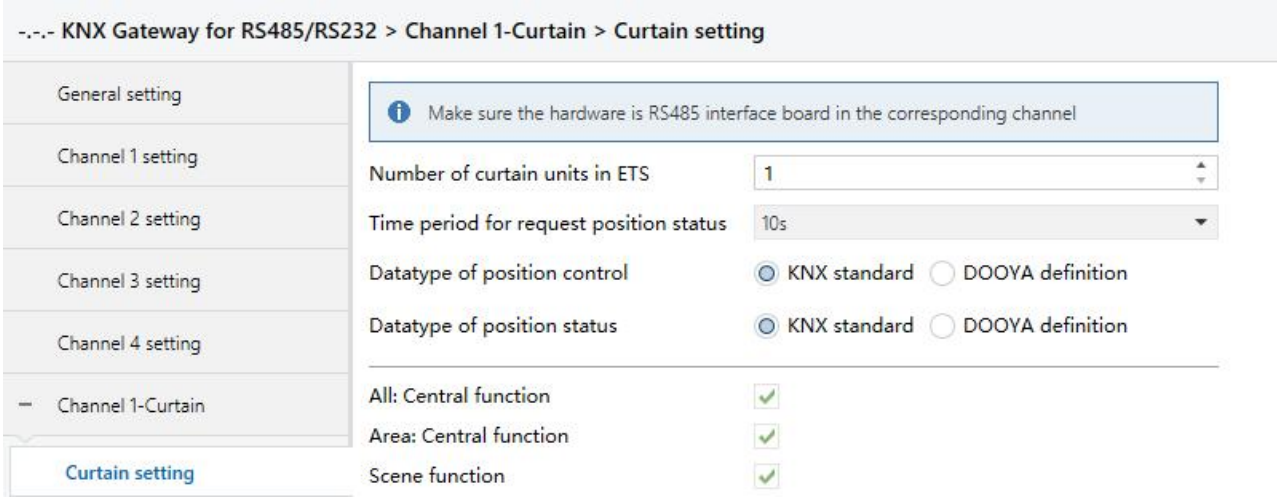


Fig.5.6.1 “Curtain gateway” parameter window

Make sure the hardware is RS485 interface board in the corresponding channel

Parameter "Number of curtain units in ETS"

This parameter is for setting the number of curtain units , Up to 16 motor controls can be supported per channel. Options: **1..16**

Parameter "Time period for request position status "

This parameter is visible when selects “Curtain position” or “Venetian blind position and slat”. Set whether to enable position status can be read, and the read cycle. Options:

Disable

1s

2s

...

60s

Parameter "Datatype of position control"

Parameter "Datatype of position status"

These two parameters are visible when selects “Curtain position” or “Venetian blind position and slat”. Set data type of position control/status value. Options:

KNX standard

DOOYA definition

KNX standard: 0%--Top/open position, 100%--Bottom/close position;

DOOYA definition: 100%--Top/open position, 0%--Bottom/close position.

Parameter "All: Central function"

This parameter is for setting whether to enable broadcast control to control all motor connected in channel, and support Up/Down/Stop/Position.

Parameter "Area: Central function"

This parameter is for setting whether to enable area control to control a area motor connected in channel, and support Up/Down/Stop/Position. Up to support 8 groups of this function objects.

Parameter "Scene function"

This parameter is for setting whether to enable scene function, up to support preset 16 scenes. Each scene can connect the motor and control commands created(Open/Close/Stop or Position, decided by curtain type).

5.6.2. Parameter window "Scene setting"

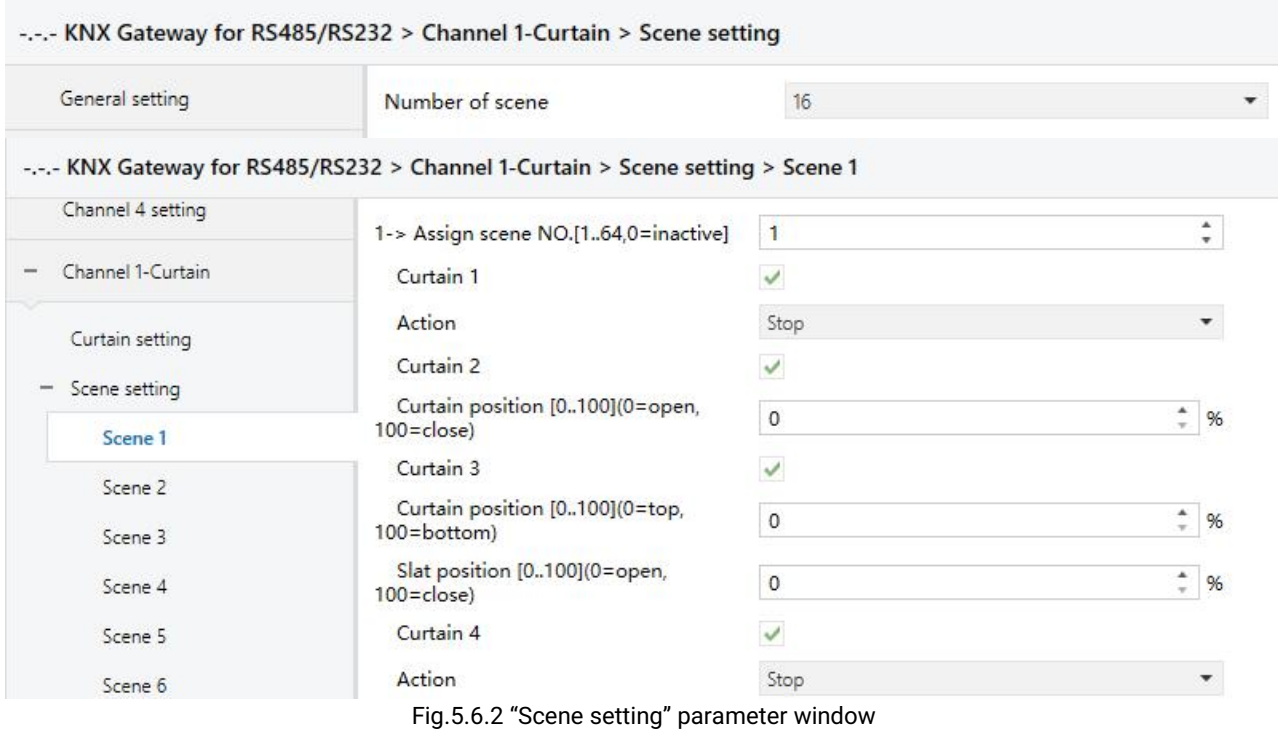


Fig.5.6.2 "Scene setting" parameter window

This window is visible when scene function enabled.

Parameter "Number of scene"

This parameter is for setting the number of scene, each channel up to support 16 scenes. Options: **1..16**

Parameter "x -> Assign scene NO.[1..64,0=inactive]"(x=16)

This parameter is for setting the scene NO. of curtain or venetian blind position. Options: **0..64, 0=inactive**

Parameter "Curtain x"(x=16)

This parameter is for setting motor connected to each preset scene according to the number of curtain units. It will apply to the selected curtain when recall the scene, while no reaction if unselect.

---Parameter "Curtain position 0..100%(0%=open, 100%=close)"

This parameter is visible when select "Curtain position". Set curtain position. Options: **0..100**

---Parameter "Curtain position 0..100%(0%=top, 100%=bottom)"

This parameter is visible when select "Venetian blind position and slat". Set venetian blind position.

Options: **0..100**

---Parameter "Slat position 0..100%(0%=open, 100%=close)"

This parameter is visible when select "Venetian blind position and slat". Set slat position.

Options: **0..100**

---Parameter "Action"

This parameter is visible when select "Curtain step/move". Set curtain action. Options:

Stop

Open

Close

5.6.3. Parameter window “Area central”

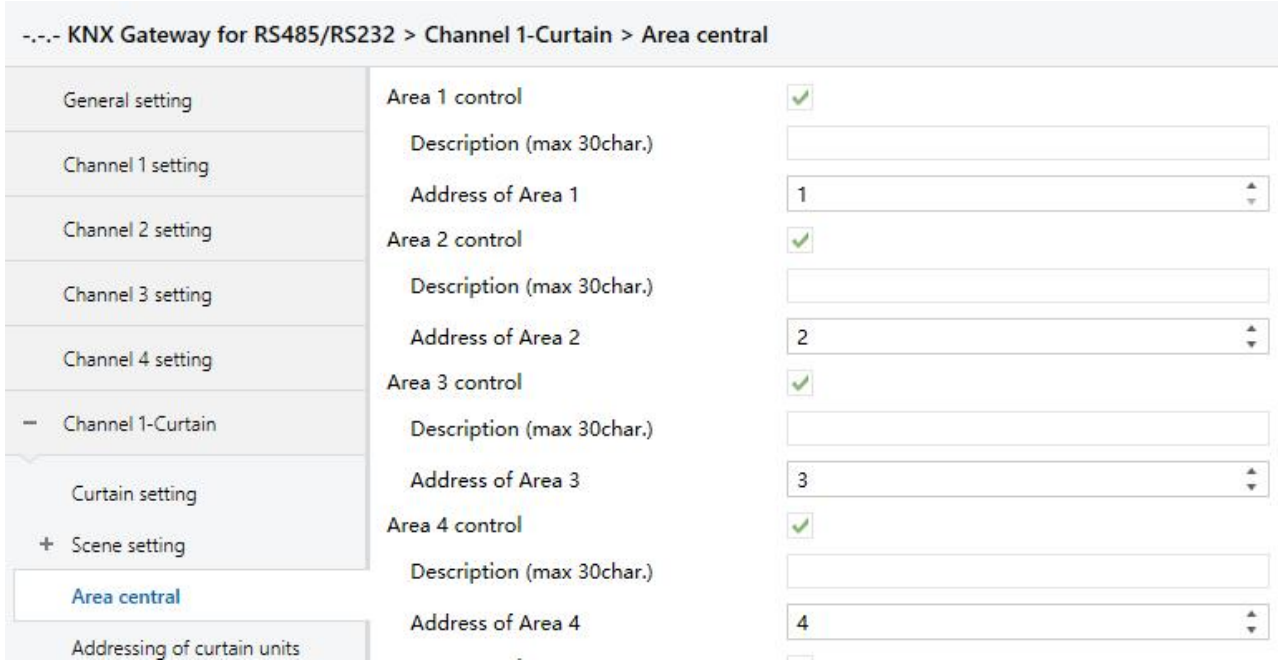


Fig.5.6.3 “Area central” parameter window

This window is visible when area control enabled.

Parameter "Area x control"(x=8)

This parameter is for setting whether to enable area control, corresponding objects and parameters are visible after selection.

——Parameter "Description (max 30char.)"

This parameter is for setting the custom description of corresponding area objects, up to 30 characters can be input.

——Parameter "Address of Area x"(x=8)

This parameter is for setting address of corresponding area. Options: **1..254**

5.6.4. Parameter window “Addressing of curtain units”

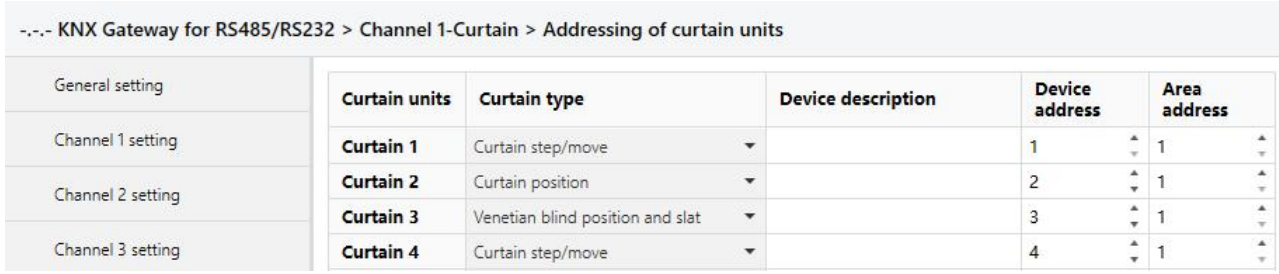


Fig.5.6.4 “Area central” parameter window

Up to set 16 curtain units, and display corresponding parameters according to the number of curtain units.

Parameter "Curtain type"

This parameter is for setting curtain type. Options:

Curtain step/move

Curtain position

Venetian blind position and slat

Parameter "Device description"

This parameter is for setting the custom description of corresponding curtain objects, up to 30 characters can be input.

Parameter "Device address"

This parameter displays the number of address according to curtain units. Set the address of device.

Options: **1..254**

Parameter "Area address"

This parameter displays the number of address according to curtain units. Set the address of area.

Options: **1..254**

5.7.Parameter window “Audio”

5.7.1. Parameter window “Audio setting”

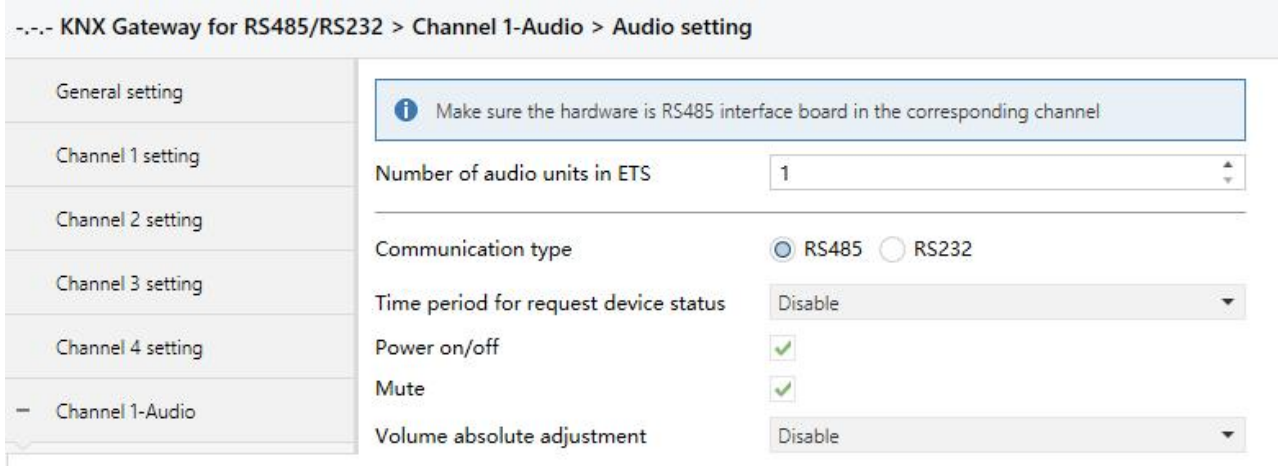


Fig.5.7.1 “Audio setting” parameter window

Make sure the hardware is RS485 interface board in the corresponding channel

Make sure the hardware is RS232 interface board in the corresponding channel

Parameter "Number of audio units in ETS"

This parameter is for setting the number of audio units , Up to 16 devices can be supported per channel. Options: **1..16**

Parameter "Communication type"

This parameter is for setting the communication type. Options:

RS485
RS232

参数"Time period for request device status "

This parameter is not visible when protocol selects “backaudio”, “Yodaar” or “cnWise”. Set whether to enable device status can be read, and the read cycle. Options:

Disable
1s
2s
...
60s

Parameter "Power on/off "

This parameter is not visible when protocol selects “MiYue”. Enable power on/off function of audio after selection, and visible corresponding objects.

Parameter "Mute"

This parameter is not visible when protocol selects "MiYue". Enable mute function of audio after selection, and visible corresponding objects.

Parameter "Volume absolute adjustment"

This parameter is for setting whether to enable volume adjustment function, and setting the datatype. Options:

Disable
Percentage (DPT_5.001)
Percentage (DPT_5.004)

——Parameter "Max. volume value [10..100]%"

This parameter is visible when previous parameter enabled. Set maximum volume value.

Options: **10..100**

5.7.2. Parameter window “Play mode configuration”

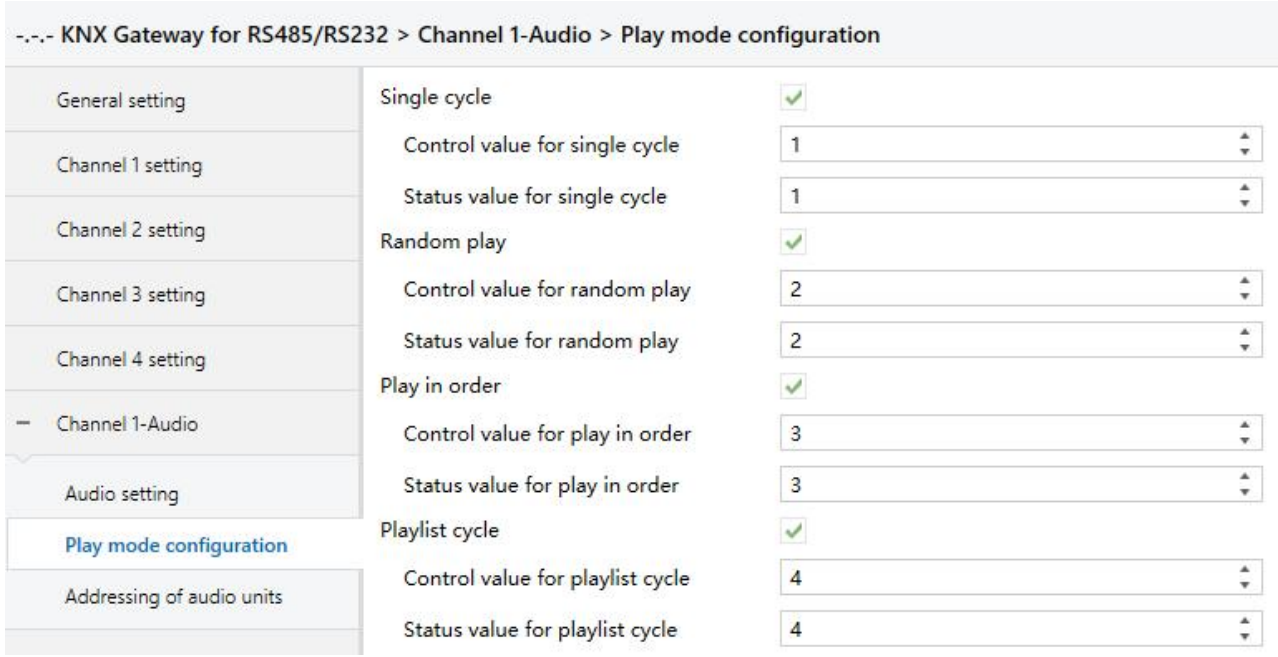


Fig.5.7.2 “Play mode configuration” parameter window

This window is not visible when protocol selects “backaudio” or “Yodaar”.

Parameter “Single cycle”

Parameter “Random play”

Parameter “Play in order”

Parameter “Playlist cycle”

These parameters for setting whether to enable each play mode, including single cycle, random play, play in order and playlist cycle.

——Parameter “Control value for single cycle”

——Parameter “Control value for random play”

——Parameter “Control value for play in order”

——Parameter “Control value for playlist cycle”

These parameters for setting control value of each play mode. Options: **0..255**

——Parameter “Status value for single cycle”

——Parameter “Status value for random play”

——Parameter “Status value for play in order”

——Parameter “Status value for playlist cycle”

These parameters for setting status value of each play mode. Options: **0..255**

5.7.3. Parameter window “Addressing of audio units”

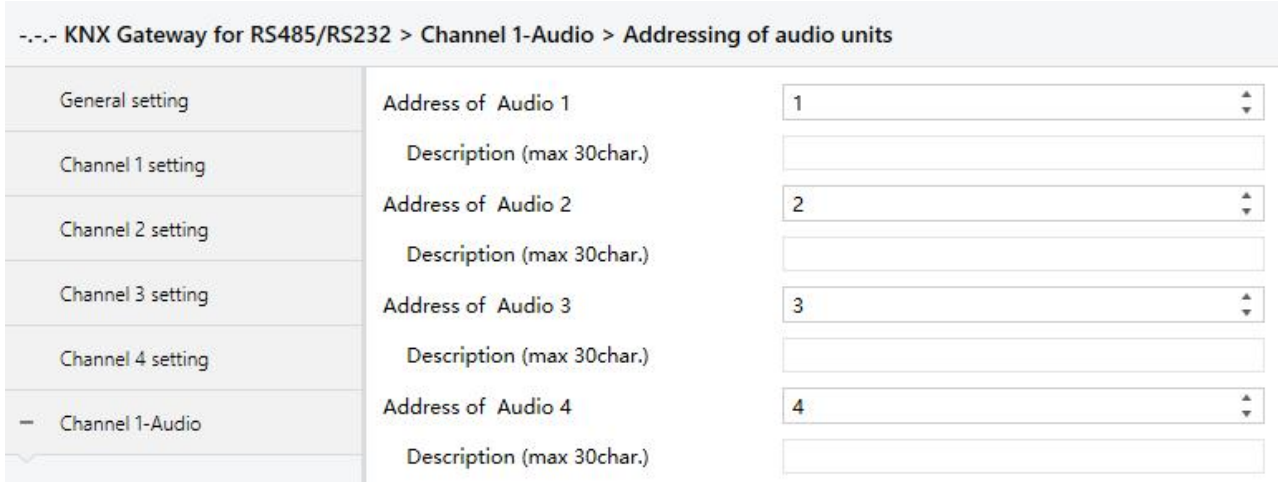


Fig.5.7.2 “Addressing of audio units” parameter window

Parameter "Address of Audio x "(x=1..16)

This parameter displays the number of address according to audio units. Set the address of device.

Options: **0..255**

——Parameter "Description (max 30char.)"

This parameter is for setting the custom description of corresponding audio objects, up to 30 characters can be input.

Chapter 6 Communication Object Description

The communication object is the medium through which the device communicates with other devices on the bus, that is, only the communication object can perform bus communication.

The function of each communication object of each function block is described in detail below.

Note: "C" in the property bar of the table below represents the communication function of the communication object;

"W" represents the value of the communication object can be rewritten by the bus;

"R" represents the value of the communication object can be read through the bus;

"T" stands for communication object with transmission function;

"U" means that the value of the communication object can be updated.

6.1. "General setting" communication object

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

Fig.6.1 "General" communication object

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

Table 6.1 "General setting " communication object

6.2. Communication object of channel

6.2.1. "Modbus<->KNX" communication object

Num	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	I	Data Type	Priority
2	CH1-Datapoint 1 : Output	1bit binary value			1 bit	C	R	-	T	-	-	switch	Low
3	CH1-Datapoint 2 : Output	1byte percent value			1 byte	C	R	-	T	-	-	percentage (0..100%)	Low
4	CH1-Datapoint 3 : Output	1byte configured value			1 byte	C	R	-	T	-	-	counter pulses (0..255)	Low
5	CH1-Datapoint 4 : Output	1byte unsigned value			1 byte	C	R	-	T	-	-	counter pulses (0..255)	Low
6	CH1-Datapoint 5 : Output	2byte configured value			2 bytes	C	R	-	T	-	-	pulses	Low
7	CH1-Datapoint 6 : Output	2byte unsigned value			2 bytes	C	R	-	T	-	-	pulses	Low
8	CH1-Datapoint 7 : Output	2byte float value			2 bytes	C	R	-	T	-	-	2-byte float value	Low
9	CH1-Datapoint 8 : Output	4byte float value			4 bytes	C	R	-	T	-	-	4-byte float value	Low

2	CH1-Datapoint 1 : Input	1bit binary value	1 bit	C	-	W	-	-	-	switch	Low
3	CH1-Datapoint 2 : Input	1byte percent value	1 byte	C	-	W	-	-	-	percentage (0..100%)	Low
4	CH1-Datapoint 3 : Input	1byte configured value	1 byte	C	-	W	-	-	-	counter pulses (0..255)	Low
5	CH1-Datapoint 4 : Input	1byte unsigned value	1 byte	C	-	W	-	-	-	counter pulses (0..255)	Low
6	CH1-Datapoint 5 : Input	2byte configured value	2 bytes	C	-	W	-	-	-	pulses	Low
7	CH1-Datapoint 6 : Input	2byte unsigned value	2 bytes	C	-	W	-	-	-	pulses	Low
8	CH1-Datapoint 7 : Input	2byte float value	2 bytes	C	-	W	-	-	-	2-byte float value	Low
9	CH1-Datapoint 8 : Input	4byte float value	4 bytes	C	-	W	-	-	-	4-byte float value	Low
502	CH1-Diagnostic: Slave (Datapoint 1-10)	No communication	1 bit	C	R	-	T	-	-	switch	Low
552	CH1-Diagnostic: Slave (common)	No communication	1 bit	C	R	-	T	-	-	switch	Low

Fig.6.2.1 “Modbus<->KNX” communication object

NO.	Object Function	Name	Data Type	Flag	DPT
2	**1bit binary value** **1byte percent value** **1byte configured value** **1byte unsigned value** **2byte configured value** **2byte unsigned value** **2byte float value**	**CH1-{{Datapoint 1}}: Output** **CH1-{{Datapoint 1}}: Input**	**1bit** **1byte** **2byte**	**C, R, W, T, U** **C, W**	**1.001 switch** **5.001 percentage(0..100%)** **5.010 counter pulses** **7.001 pulses** **9.001 temperature**
This communication object is used to converter between KNX value and register value. The range of value is determined by the selected data type. Flag and name is determined by communication direction: KNX to Modbus (Input)/Modbus to KNX (Output) The name in parentheses changes with the parameter “Description (max 30 char.)”. If description is empty, display “...Datapoint x...” by default.					
502	**No communication**	**CH1-Diagnostic: {{Slave (Datapoint 1-10)}}**	**1bit**	**C, R, T**	**1.001 switch**
This communication object is visible when select “Modbus master” and slave address set as “Individual for current page”, used for diagnostic. Send ON telegram to the bus via this object if the master not receives a response from the salve. The name in parentheses changes with the parameter “Slave description (max 30char.)”. If description is empty, display “Slave (Datapoint 1-10)” by default.					
522	**No communication**	**CH1-Diagnostic: Slave (common)**	**1bit**	**C, R, T**	**1.001 switch**
For the master: Send ON telegram to the bus via this object if the master not receives a response from the salve. For the slave: Send ON telegram to the bus via this object if not receives a request from the master during the request time.					

Table 6.2.1 “Modbus<->KNX” communication object

6.2.2. "RS485/RS485<->KNX" communication object

Number	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
2	CH1-Datapoint 1: Output	1bit binary value			1 bit	C	R	-	T	-	switch	Low
2	CH1-Datapoint 1: Input	1bit binary value			1 bit	C	-	W	-	-	switch	Low
2	CH1-Datapoint 1: Output	1byte unsigned value			1 byte	C	R	-	T	-	counter pulses (0..255)	Low
2	CH1-Datapoint 1: Input	1byte unsigned value			1 byte	C	-	W	-	-	counter pulses (0..255)	Low
2	CH1-Datapoint 1: Output	1byte scene No.			1 byte	C	R	-	T	-	scene number	Low
2	CH1-Datapoint 1: Input	1byte scene No.			1 byte	C	-	W	-	-	scene number	Low
2	CH1-Datapoint 1: Output	2byte unsigned value			2 bytes	C	R	-	T	-	pulses	Low
2	CH1-Datapoint 1: Input	2byte unsigned value			2 bytes	C	-	W	-	-	pulses	Low

Fig.6.2.2 "RS485/RS232<->KNX" communication object

NO.	Object Function	Name	Data Type	Flag	DPT
2	**1bit binary value** **1byte unsigned value** **1byte scene No.** **2byte unsigned value**	**CH1-{{Datapoint 1}}: Output** **CH1-{{Datapoint 1}}: Input**	**1bit** **1byte** **2byte**	**C, R, T** **C, W**	**1.001 switch** **5.010 counter pulses** **17.001 scene number** **7.001 pulses**
This communication object is used to converter between KNX value and RS485/232 value. The range of value is determined by the selected data type. Flag and name is determined by communication direction: KNX to RSxxx (Input)/RSxxx to KNX (Output) The name in parentheses changes with the parameter "Description (max 30 char.)". If description is empty, display "...Datapoint x..." by default.					

Table 6.2.2 "RS485/232<->KNX" communication object

6.2.3. "VRV/VRF gateway" communication object

Numl	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
2	CH1-AC 1 Control	Power on/off			1 bit	C	-	W	-	U	switch	Low
3	CH1-AC 1 Control	Current setpoint adjustment			2 bytes	C	-	W	-	U	temperature (°C)	Low
5	CH1-AC 1 Control	Control mode			1 byte	C	-	W	-	U	HVAC control mode	Low
6	CH1-AC 1 Control	Fan speed			1 byte	C	-	W	-	U	percentage (0..100%)	Low
7	CH1-AC 1 Control	Vanes swing (1-swing,0-stop)			1 bit	C	-	W	-	U	start/stop	Low
11	CH1-AC 1 Control	Window contact			1 bit	C	-	W	-	U	window/door	Low
13	CH1-AC 1 Status	Power on/off			1 bit	C	R	-	T	-	switch	Low
14	CH1-AC 1 Status	Current temperature setpoint			2 bytes	C	R	-	T	-	temperature (°C)	Low
15	CH1-AC 1 Status	Ambient reference temperature			2 bytes	C	R	-	T	-	temperature (°C)	Low
16	CH1-AC 1 Status	Control mode			1 byte	C	R	-	T	-	HVAC control mode	Low
17	CH1-AC 1 Status	Fan speed			1 byte	C	R	-	T	-	percentage (0..100%)	Low
18	CH1-AC 1 Status	Vanes swing (1-swing,0-stop)			1 bit	C	R	-	T	-	start/stop	Low
22	CH1-AC 1 Status	Communication error			1 bit	C	R	-	T	-	alarm	Low

Fig.6.2.3 "VRV/VRF gateway" communication object

NO.	Object Function	Name	Data Type	Flag	DPT
2	**Power on/off**	**CH1-AC 1 Control**	**1bit**	**C, W, U**	**1.001 switch**
This communication object is used to control power on/off status of Air-conditioning via bus. Telegram value: 1-On/0-Off					
3	**Current setpoint adjustment**	**CH1-AC 1 Control**	**1byte** **2byte**	**C, W, U**	**5.010 counter pluses(0..255)** **9.001 temperature**
This communication object is used to adjust the setpoint value of current temperature.The range of value is determined by the data type of setpoint temperature: 0..255/0..65535					
5	**Control mode**	**CH1-AC 1 Control**	**1byte**	**C, W, U**	**20.105 HVAC control mode**
This communication object is used to receive the control telegram of each mode of Air-conditioning from bus.					
6	**Fan speed**	**CH1-AC 1 Control**	**1byte**	**C, W, U**	**5.001 percentage**
This communication object is used to control Air-conditioning fan speed via bus.					
7	**Vanes swing (1-swing,0-stop)**	**CH1-AC 1 Control**	**1bit**	**C, W, U**	**1.010 start/stop**
7	**Vanes Up-Down swing (1-swing,0-stop)**	**CH1-AC 1 Control**	**1bit**	**C, W, U**	**1.010 start/stop**
9	**Vanes Left-Right swing (1-swing,0-stop)**	**CH1-AC 1 Control**	**1bit**	**C, W, U**	**1.010 start/stop**
The object "Vanes swing (1-swing,0-stop)" is visible when AC protocol is selected "DAIKIN (DTA116A621)", "Toshiba" or "Other", and vanes swing is enabled. The object "Vanes Up-Down swing (1-swing,0-stop)" or "Vanes Left-Right swing (1-swing,0-stop)" is visible when AC protocol is selected "Fujitsu", and vanes Up-Down swing or vanes Left-Right swing is enabled.					

These communication objects are used to control Air-conditioning vanes swing via bus. Telegram value: 0-stop/1-swing					
11	**Window contact**	**CH1-AC 1 Control**	**1bit**	**C, W, U**	**5.010 counter pluses(0..255)**
This communication object is used to receive the window contact status.					
13	**Power on/off**	**CH1-AC 1 Status**	**1bit**	**C,R,T**	**1.001 switch**
This communication object is used to send the power on/off telegram of Air-conditioning. Telegram value: 1-On/0-Off					
14	**Current setpoint adjustment**	**CH1-AC 1 Status**	**1byte** **2byte**	**C,R,T**	**5.010 counter pluses(0..255)** **9.001 temperature**
The communication object is used to send the actual setpoint temperature value of AC units to the bus. The range of value is determined by the data type of setpoint temperature: 0..255 / 0..65535					
16	**Control mode**	**CH1-AC 1 Status**	**1byte**	**C,R,T**	**20.105 HVAC control mode**
This communication object is used to send the control telegram of each mode of Air-conditioning to bus.					
17	**Fan speed**	**CH1-AC 1 Status**	**1byte**	**C,R,T**	**5.001 percentage**
The communication object is used to read fan speed status of Air-conditioning.					
18	**Vanes swing (1-swing,0-stop)**	**CH1-AC 1 Status**	**1bit**	**C,R,T**	**1.010 start/stop**
18	**Vanes Up-Down swing (1-swing,0-stop)**	**CH1-AC 1 Status**	**1bit**	**C,R,T**	**1.010 start/stop**
20	**Vanes Left-Right swing (1-swing,0-stop)**	**CH1-AC 1 Status**	**1bit**	**C,R,T**	**1.010 start/stop**
The object “Vanes swing (1-swing,0-stop)” is visible when AC protocol is selected “DAIKIN (DTA116A621)”, “Toshiba” or “Other”, and vanes swing is enabled. The object “Vanes Up-Down swing (1-swing,0-stop)” or “Vanes Left-Right swing (1-swing,0-stop)” is visible when AC protocol is selected “Fujitsu”, and vanes Up-Down swing or vanes Left-Right swing is enabled. These communication objects are used to read vanes swing status of Air-conditioning. Telegram value: 0-stop/1-swing					
22	**Communication error**	**CH1-AC 1 Status**	**1bit**	**C,R,T**	**1.005 alarm**
The communication object is used to read communication error status between Air-conditioning and gateway.					

Table 6.2.3 “VRV/VRF gateway” communication object

6.2.4. “Curtain” communication object

Number	Name	Object Function	Descri	Group A	Length	C	R	W	T	U	Data Type	Priority
2	CH1-Curtain 1: Control	Close/Open			1 bit	C	-	W	-	-	open/close	Low
3	CH1-Curtain 1: Control	Stop			1 bit	C	-	W	-	-	step	Low
98	CH1-Area 1: Central	Up/Down			1 bit	C	-	W	-	-	up/down	Low
99	CH1-Area 1: Central	Stop/step			1 bit	C	-	W	-	-	step	Low
100	CH1-Area 1: Central	Curtain position (0..100%)			1 byte	C	-	W	-	-	percentage (0..100...	Low
101	CH1-Area 1: Central	Slat position (0..100%)			1 byte	C	-	W	-	-	percentage (0..100...	Low
130	CH1-All: Central	Up/Down			1 bit	C	-	W	-	-	up/down	Low
131	CH1-All: Central	Stop/step			1 bit	C	-	W	-	-	step	Low
132	CH1-All: Central	Curtain position (0..100%)			1 byte	C	-	W	-	-	percentage (0..100...	Low
133	CH1-All: Central	Slat position (0..100%)			1 byte	C	-	W	-	-	percentage (0..100...	Low
134	CH1-Scene	Scene/save			1 byte	C	-	W	-	-	scene control	Low

Curtain step/move

Number	Name	Object Function	Descri	Group A	Length	C	R	W	T	U	Data Type	Priority
2	CH1-Curtain 1: Control	Close/Open			1 bit	C	-	W	-	-	open/close	Low
3	CH1-Curtain 1: Control	Stop			1 bit	C	-	W	-	-	step	Low
4	CH1-Curtain 1: Control	Curtain position (0..100%)			1 byte	C	-	W	-	-	percentage (0..100...	Low
6	CH1-Curtain 1: Status	Curtain position (0..100%)			1 byte	C	R	-	T	-	percentage (0..100...	Low
98	CH1-Area 1: Central	Up/Down			1 bit	C	-	W	-	-	up/down	Low
99	CH1-Area 1: Central	Stop/step			1 bit	C	-	W	-	-	step	Low
100	CH1-Area 1: Central	Curtain position (0..100%)			1 byte	C	-	W	-	-	percentage (0..100...	Low
101	CH1-Area 1: Central	Slat position (0..100%)			1 byte	C	-	W	-	-	percentage (0..100...	Low
130	CH1-All: Central	Up/Down			1 bit	C	-	W	-	-	up/down	Low
131	CH1-All: Central	Stop/step			1 bit	C	-	W	-	-	step	Low
132	CH1-All: Central	Curtain position (0..100%)			1 byte	C	-	W	-	-	percentage (0..100...	Low
133	CH1-All: Central	Slat position (0..100%)			1 byte	C	-	W	-	-	percentage (0..100...	Low
134	CH1-Scene	Scene/save			1 byte	C	-	W	-	-	scene control	Low

Curtain position

Number	Name	Object Function	Descri	Group A	Length	C	R	W	T	U	Data Type	Priority
2	CH1-Curtain 1: Control	Up/Down			1 bit	C	-	W	-	-	up/down	Low
3	CH1-Curtain 1: Control	Stop/step			1 bit	C	-	W	-	-	step	Low
4	CH1-Curtain 1: Control	Curtain position (0..100%)			1 byte	C	-	W	-	-	percentage (0..100...	Low
5	CH1-Curtain 1: Control	Slat position (0..100%)			1 byte	C	-	W	-	-	percentage (0..100...	Low
6	CH1-Curtain 1: Status	Curtain position (0..100%)			1 byte	C	R	-	T	-	percentage (0..100...	Low
7	CH1-Curtain 1: Status	Slat position (0..100%)			1 byte	C	R	-	T	-	percentage (0..100...	Low
98	CH1-Area 1: Central	Up/Down			1 bit	C	-	W	-	-	up/down	Low
99	CH1-Area 1: Central	Stop/step			1 bit	C	-	W	-	-	step	Low
100	CH1-Area 1: Central	Curtain position (0..100%)			1 byte	C	-	W	-	-	percentage (0..100...	Low
101	CH1-Area 1: Central	Slat position (0..100%)			1 byte	C	-	W	-	-	percentage (0..100...	Low
130	CH1-All: Central	Up/Down			1 bit	C	-	W	-	-	up/down	Low
131	CH1-All: Central	Stop/step			1 bit	C	-	W	-	-	step	Low
132	CH1-All: Central	Curtain position (0..100%)			1 byte	C	-	W	-	-	percentage (0..100...	Low
133	CH1-All: Central	Slat position (0..100%)			1 byte	C	-	W	-	-	percentage (0..100...	Low
134	CH1-Scene	Scene/save			1 byte	C	-	W	-	-	scene control	Low

Venetian blind position and slat

Fig.6.2.4 “Curtain” communication object

NO.	Object Function	Name	Data Type	Flag	DPT
2	**Close/Open**	**CH1-{{Curtain 1}}: Control**	**1bit**	**C,W**	**1.009 open/close**
3	**Stop**	**CH1-{{Curtain 1}}: Control**	**1bit**	**C,W**	**1.007 step**

Motor control: Open/Close/Stop. Visible when select “Curtain step/move”, is suitable for opening and closing curtains. The object description is as follows:

Obj.2: The communication object is used to receive the open/close telegram from the bus. Telegram value:

1——Close the curtain

0——Open the curtain

Obj.3: The communication object is used to receive a telegram for stopping the curtain movement from the bus. Telegram value:

1——Stop

The name in parentheses changes with the parameter “Device description”. If description is empty, display “...Curtain x...” by default.

2	**Close/Open**	**CH1-{{Curtain 1}}: Control**	**1bit**	**C,W**	**1.009 open/close**
3	**Stop**	**CH1-{{Curtain 1}}: Control**	**1bit**	**C,W**	**1.007 step**
4	**Curtain position (0..100%)**	**CH1-{{Curtain 1}}: Control**	**1byte**	**C,W**	**5.001 percentage**
6	**Curtain position (0..100%)**	**CH1-{{Curtain 1}}: Status**	**1byte**	**C,R,T**	**5.001 percentage**

Motor control: Open/Close/Stop/Position. Visible when select “Curtain position”, is suitable for roller blind without slat. The object description is as follows:

Obj.2: The communication object is used to receive a telegram value from the bus to control the opening /closing of the roller blind. Telegram value:

1——Close the curtain

0——Open the curtain

Obj.3: The communication object is used to receive a telegram for stopping the curtain movement from the bus. Telegram value:

1——Stop

Obj.4: The communication object is used to receive a telegram controlling the position of the curtain from the bus. Telegram value: 0..100%

Obj.6: The communication object is used to send a telegram controlling the position of the curtain to the bus. Telegram value: 0..100%

2	**Up/Down**	**CH1-{{Curtain 1}}: Control**	**1bit**	**C,W**	**1.008 up/down**
3	**Stop/step**	**CH1-{{Curtain 1}}: Control**	**1bit**	**C,W**	**1.007 step**
4	**Curtain position (0..100%)**	**CH1-{{Curtain 1}}: Control**	**1byte**	**C,W**	**5.001 percentage**
5	**Slat position (0..100%)**	**CH1-{{Curtain 1}}: Control**	**1byte**	**C,W**	**5.001 percentage**
6	**Curtain position (0..100%)**	**CH1-{{Curtain 1}}: Status**	**1byte**	**C,R,T**	**5.001 percentage**

7	**Slat position (0..100%)**	**CH1-{{Curtain 1}}: Status**	**1byte**	**C,R,T**	**5.001 percentage**

Motor control: Up/Down/Stop/Position. Visible when select “Venetian blind position and slat”, is suitable for venetian blind with slat. The object description is as follows:

Obj.3: The communication object is used to receive a telegram from the bus to stop the curtain movement or adjust the slat angle.Telegram value:

1——Stop/Slat adj. Down

0——Stop/Slat adj. Up

Obj.5: The communication object is used to receive a telegram controlling the angle position of the slat from bus. Telegram value: 0. 100%

Obj.7: The communication object is used to send a telegram controlling the angle position of the slat to bus. Telegram value: 0. 100%

Obj.2，Obj.4 and Obj.6 are the same as above.

98	**Up/Down**	**CH1-{{Area 1}}: Central**	**1bit**	**C,W**	**1.008 up/down**
99	**Stop/step**	**CH1-{{Area 1}}: Central**	**1bit**	**C,W**	**1.007 step**
100	**Curtain position (0..100%)**	**CH1-{{Area 1}}: Central**	**1byte**	**C,W**	**5.001 percentage**
101	**Slat position (0..100%)**	**CH1-{{Area 1}}: Central**	**1byte**	**C,W**	**5.001 percentage**

Area control: Up/Down/Stop/Position. The object description is as follows:

Obj.98: The communication object is used to receive a telegram value from the bus to control the opening /closing of the blind. Telegram value:

1——Move down

0——Move up

Obj.99: The communication object is used to receive a telegram for stopping the curtain movement from the bus. Telegram value:

1——Stop

Obj.100: The communication object is used to send a telegram controlling the position of the curtain to the bus. Telegram value: 0..100%

Obj.101: The communication object is used to send a telegram controlling the angle position of the slat to bus. Telegram value: 0. 100%

The name in parentheses changes with the parameter “Description (max 30char.)”. If description is empty, display “...Area x...” by default.

130	**Up/Down**	**CH1-All: Central**	**1bit**	**C,W**	**1.008 up/down**
131	**Stop/step**	**CH1-All: Central**	**1bit**	**C,W**	**1.007 step**
132	**Curtain position (0..100%)**	**CH1-All: Central**	**1byte**	**C,W**	**5.001 percentage**
133	**Slat position (0..100%)**	**CH1-All: Central**	**1byte**	**C,W**	**5.001 percentage**

Broadcast control: Up/Down/Stop/Position. The object description is as follows:

Obj.130: The communication object is used to receive a telegram value from the bus to control the opening /closing of the blind. Telegram value:

1——Move down

0——Move up

Obj.131: The communication object is used to receive a telegram for stopping the curtain movement from the bus. Telegram value:

1——Stop

Obj.132: The communication object is used to send a telegram controlling the position of the curtain to the bus. Telegram value: 0..100%

Obj.133: The communication object is used to send a telegram controlling the angle position of the slat to bus. Telegram value: 0. 100%

134	**Scene/save**	**CH1-Scene**	**1byte**	**C,W**	**18.001 scene control**
The communication object is used to receive a telegram of scene recall from bus. Telegram value: 0..63					

Table 6.2.4 “Curtain” communication object

6.2.5. “Audio” communication object

Number	Name	Object Function	Description	Group Address	Length	C	R	W	T	U	Data Type	Priority
2	CH1-Audio 1: Control	Power on/off			1 bit	C	-	W	-	U	switch	Low
3	CH1-Audio 1: Control	Play=1/Pause=0			1 bit	C	-	W	-	U	start/stop	Low
4	CH1-Audio 1: Control	Next track=1/Previous track=0			1 bit	C	-	W	-	-	step	Low
5	CH1-Audio 1: Control	Mute			1 bit	C	-	W	-	U	enable	Low
6	CH1-Audio 1: Control	Volume+=1/Volume-=0			1 bit	C	-	W	-	-	step	Low
7	CH1-Audio 1: Control	Absolute volume			1 byte	C	-	W	-	U	percentage (0..100%)	Low
8	CH1-Audio 1: Control	Play mode			1 byte	C	-	W	-	U	counter pulses (0..255)	Low
9	CH1-Audio 1: Status	Power on/off			1 bit	C	R	-	T	-	switch	Low
10	CH1-Audio 1: Status	Play=1/Pause=0			1 bit	C	R	-	T	-	start/stop	Low
11	CH1-Audio 1: Status	Mute			1 bit	C	R	-	T	-	enable	Low
12	CH1-Audio 1: Status	Absolute volume			1 byte	C	R	-	T	-	percentage (0..100%)	Low
13	CH1-Audio 1: Status	Play mode			1 byte	C	R	-	T	-	counter pulses (0..255)	Low

Fig.6.2.5 “Audio” communication object

NO.	Object Function	Name	Data Type	Flag	DPT
2	**Power on/off**	**CH1-{{Audio 1}}: Control**	**1bit**	**C,W,U**	**1.001 switch**
The communication object is used to receive the background music on/off controlling telegram from the bus, to control the power of the background music module. Telegram value: 1——On 0——Off The name in parentheses changes with the parameter “Description (max 30 char.)”. If description is empty, display “...Audio x...” by default.					
3	**Play=1/Pause=0**	**CH1-{{Audio 1}}: Control**	**1bit**	**C,W,U**	**1.010 start/stop**
The communication object is used to play/stop the music in the background music module. Telegram value: 1——Play music 0——Pause playing music					
4	**Next track=1/Previous track=0**	**CH1-{{Audio 1}}: Control**	**1bit**	**C,W**	**1.007 step**

The communication object is used to switch the playing song of the background music module to the previous song/the next song. Telegram value:

1——Play the next song

0——Play the previous song

5	**Mute**	**CH1-{{Audio 1}}: Control**	**1bit**	**C,W,U**	**1.003 enable**
6	**Volume+=1/Volume-=0**	**CH1-{{Audio 1}}: Control**	**1bit**	**C,W**	**1.007 step**

When 1bit, support volume adjustment and mute function. The object description is as follows:

Obj.5: The communication object is used to receive a telegram value from the bus to control mute. Telegram value:

1——Mute

0——Exit mute

Obj.6: The communication object is used to adjust the volume of the background music module. Telegram value:

1——Increase volume

0——Decrease volume

7	**Absolute volume**	**CH1-{{Audio 1}}: Control**	**1byte**	**C,W,U**	**5.001 percentage** **5.004 percentage**

When 1byte, only support volume adjustment. The object description is as follows:

Obj.6: The communication object is used to adjust the volume of the background music module. The range of value is determined by the selected data type: 0..100 / 0..255

8	**Play mode**	**CH1-{{Audio 1}}: Control**	**1byte**	**C,W,U**	**5.010 counter pulses**

The communication object is used to receive the play mode control telegram of the background music from bus. Telegram of different mode preset by parameter.

9	**Power on/off**	**CH1-{{Audio 1}}: Status**	**1bit**	**C,R,T**	**1.001 switch**

The communication object is used to send the on/off status telegram of background music to the bus. Telegram value:

1——On

0——Off

10	**Play=1/Pause=0**	**CH1-{{Audio 1}}: Status**	**1bit**	**C,R,T**	**1.010 start/stop**

The communication object is used to send the play status telegram of background music to the bus. Telegram value:

1——Play music

0——Pause playing music

11	**Mute**	**CH1-{{Audio 1}}: Status**	**1bit**	**C,R,T**	**1.003 enable**

When 1bit, this communication object is used to send the mute status to the bus. Telegram value:

1——Mute

0——Exit mute

12	**Absolute volume**	**CH1-{{Audio 1}}: Status**	**1byte**	**C,R,T**	**5.001 percentage** **5.004 percentage**

When 1byte, this communication object is used to send the volume status to the bus of background music. The range of value is determined by the selected data type: 0..100 / 0..255					
13	**Play mode**	**CH1-{{Audio 1}}: Status**	**1byte**	**C,R,T**	**5.010 counter pulses**
The communication object is used to send the play mode status telegram of the background music to bus. Telegram of different mode preset by parameters.					

Table 6.2.5 “Audio” communication object