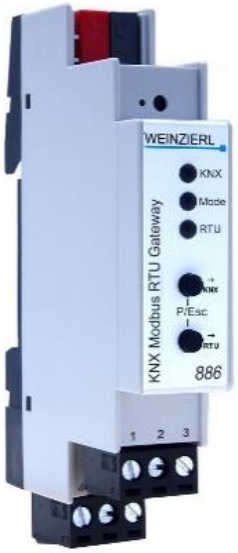
KNX Modbus Gateway with 250 data points

KNX Modbus RTU Gateway 886

Operation and installation manual



(Art. # 5256)

WEINZIERL ENGINEERING GmbH
Achatz 3-4
84508 Burgkirchen an der Alz
GERMANY

Tel.: +49 8677 / 916 36 – 0
E-Mail: info@weinzierl.de
Web: www.weinzierl.de

Content

1 Application

The KNX Modbus RTU Gateway 886 is a compact gateway between KNX TP and Modbus RTU with 250 free configurable channels.

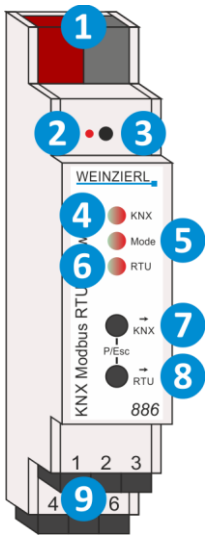
The device enables an easy integration of Modbus devices that support the RTU protocol via RS-485 and can act as Modbus master or slave. The device can be used as a master for up to 25 slave devices.

The assignment between KNX objects and Modbus registers can be configured via parameters in the ETS. No additional software is required. The KNX Bus and Modbus have a galvanic isolation from each other.

Two push buttons and three LEDs enable a local operation and a visualization of the device status.

2 Installation and connection

The KNX Modbus RTU Gateway 886 is designed for installation on a DIN rail (35 mm) with a width of 1 unit (18 mm). An installation friendly design with pluggable screw terminals helps to reduce the cost of commissioning. It features the following controls and displays:



1. KNX bus connector
2. Programming LED
3. Button for programming mode
4. LED KNX (multicolor)
5. LED Mode (multicolor)
6. LED RTU (multicolor)
7. Button KNX
8. Button RTU
9. Pluggable screw terminals

The device has galvanic isolation between Modbus and KNX. The Modbus side must be connected with 12...24 V⎓. The unchoked output voltage from the KNX power supply must not be used for this purpose.

If the bus voltage is missing, the device is without function.

2.1 KNX programming mode

The KNX programming mode is activated/deactivated either by pressing the flushed KNX programming button 3 or by simultaneously pressing the buttons 7 and 8. The operation of the programming mode at the front can be activated/deactivated in the ETS® with Prog. mode on device front.

When the programming mode is active, the programming LED 2 and the LED Mode 5 light up red.

2.2 Manual operation and status display

The LED KNX 4 lights up or flashes when KNX bus voltage is present.

The LED Mode 5 lights up or flashes when KNX bus voltage is present.

The LED RTU 6 lights up or flashes when supply voltage is present.

The synchronization of the KNX objects is triggered by pressing the button KNX 7 for a long time. This is indicated by the LED Mode 5 lighting up in orange.

Pressing the button RTU 8 for a long time triggers the synchronization of the Modbus registers. This is indicated by the LED Mode 5 lighting up in orange.

The LED KNX 4 is used to indicate the status of KNX communication. It flashes when sending and receiving telegrams.

The LED RTU 6 indicates the status of the Modbus communication. It flashes when sending and receiving telegrams.

Summary of the states of the LED Mode 5:

LED Status	Meaning
LED lights green	The unit operates in normal operating mode.
LED lights red	The programming mode is active.
LED lights orange	The programming mode is not active. Synchronization is active.
LED flashes red	The programming mode is not active. Synchronization is not active. The device is not loaded correctly, e.g. after aborting a download.
LED flashes green	The device is currently loaded by the ETS.

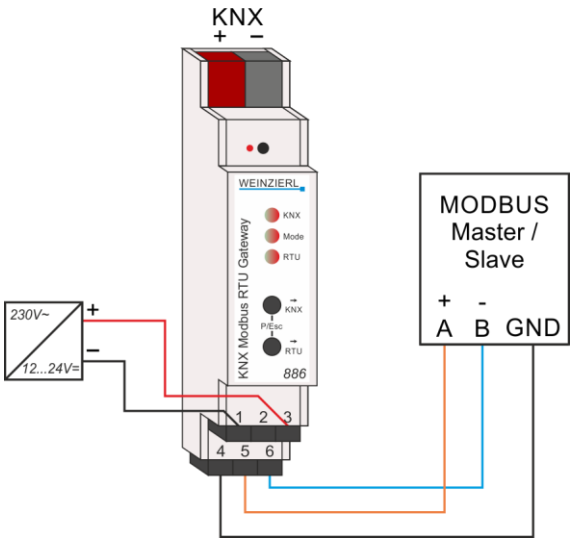
3 Reset to factory default settings

It is possible to reset the device to the factory default settings:

- Disconnect the KNX Bus connector (1) from the device.
- Press the KNX programming button (3) and keep it pressed down.
- Reconnect the KNX Bus connector (1) to the device.
- Keep the KNX programming button (3) pressed for at least another 6 seconds.
- A short flashing of all LEDs ((2)(4)(5)(6)) visualizes the successful reset of the device to factory default settings.

In the factory default settings, the device has the physical address 15.15.255 and no group addresses are connected.

4 Wiring scheme



4.1 Pluggable screw terminals

The upper terminal is used to connect the supply voltage 12 ... 24 V⎓, the lower terminal to connect the Modbus:

-	-	+
-	A	B

4.2 Pin assignment

Connection	Icon	Description
1	-	Ground connection for supply voltage
2	-	Ground connection for supply voltage
3	+	Positive connection for supply voltage 12 ... 24 V⎓
4	-	Ground connection for Modbus (connected to connection 1 and connection 2)
5	A	Data line A (+) for Modbus
6	B	Data line B (-) for Modbus
KNX	+	Positive connection for KNX bus
KNX	-	Ground connection for KNX bus

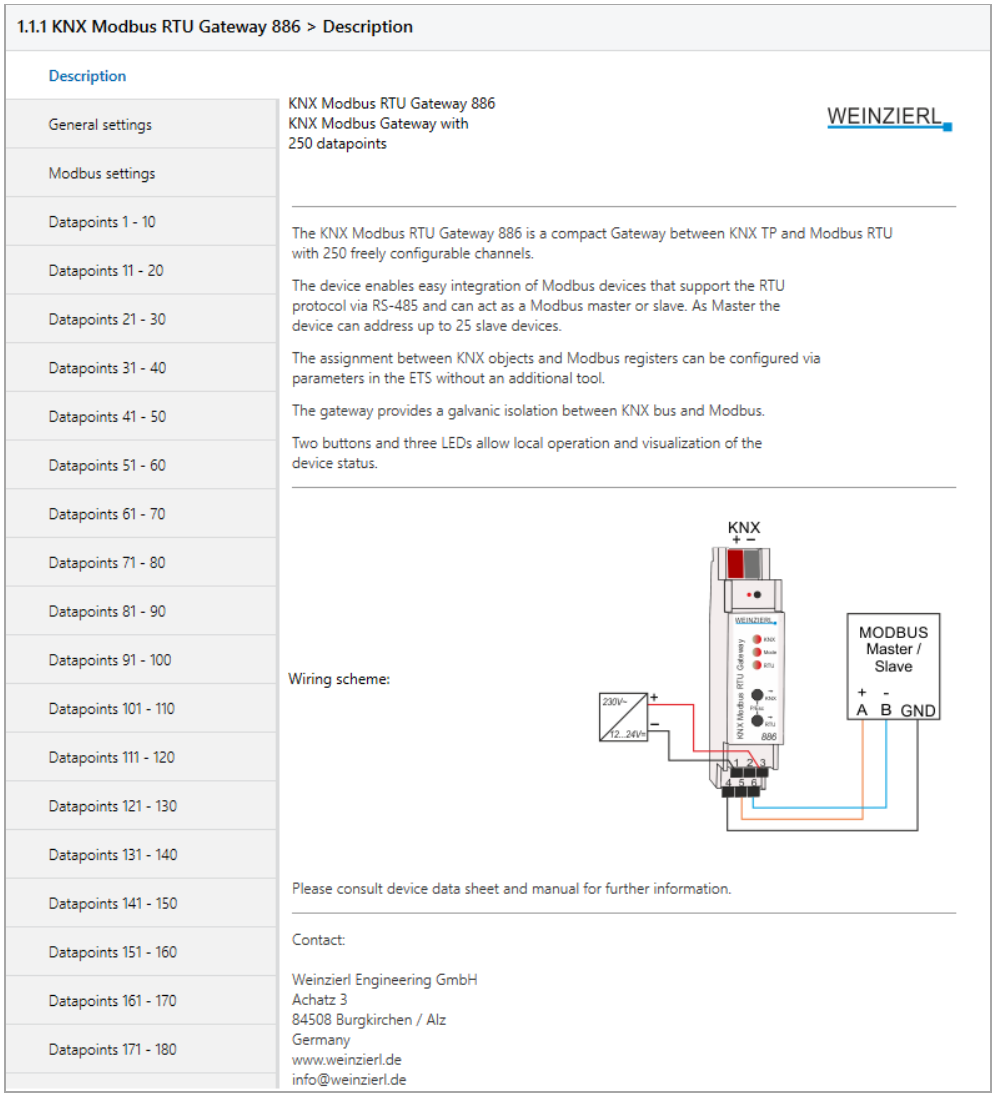
The transmission line must be terminated at the last participant of the Modbus transmission line with a 120 Ohm / 0.25 W resistor. This resistor must be inserted directly between the two signal lines before the input of the last device. No terminating resistor is installed in the device itself.

Only shielded and twisted cables should be used for Modbus.

5 ETS database

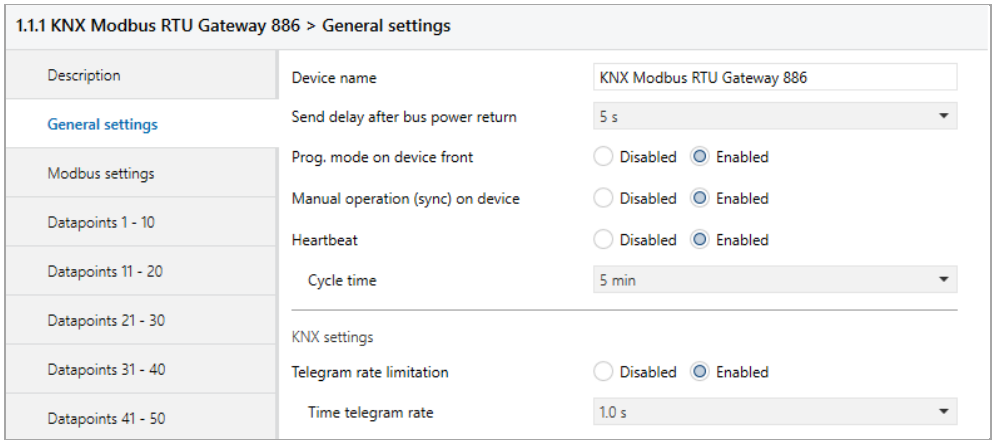
The ETS5 database (for ETS 5.7 or newer) can be downloaded from the product website of the KNX Modbus RTU Gateway 886 (www.weinzierl.de) or from the ETS online catalogue.

5.1 Description



The first page shows general information about the device.

5.2 General settings



Device name *(30 characters)*

Any arbitrary name can be assigned to the KNX Modbus RTU Gateway 886. The device name should be meaningful, e.g. “Living room ground floor”. This helps the clarity of the ETS project.

Send delay after bus power return

The send delay of telegrams after the return of the bus voltage can be set via this parameter. In this case, telegrams from the device are sent to the KNX bus in a delayed manner by the set time. This reduces the bus load when the bus voltage returns. Other functions such as receiving telegrams are not affected by this parameter.

Prog. mode on device front

In addition to the normal programming button 3 the device allows activating the programming mode on the device front without opening the switchboard cover. The programming mode can be activated and deactivated via pressing simultaneously both buttons 7 and 8.

This feature can be enabled and disabled via the parameter Prog. mode on device front. The recessed programming button 3 (next to the programming LED 2) is always enabled and is not affected by this parameter.

Manual operation (sync) on device

This parameter is used to configure manual operation on the device. The manual operation can be activated or deactivated.

Manual operation enables synchronization of all channels in the direction to KNX (button KNX 7) and in the direction to Modbus (button RTU 8).

Direction KNX:
All datapoints of the channels configured as “Modbus to KNX” send their current value on the KNX bus.

Direction Modbus:
If the gateway is operating as Modbus master, all registers of the channels configured “KNX to Modbus” are written to Modbus again.

Synchronization can be cancelled by pressing the buttons 7 and 8 simultaneously.

Heartbeat

Cyclic sending of values to the KNX bus, to indicate that the device is operational. For the Cycle time values between 1 min and 24 h are selectable.

Group object	Type KNX	Size	Direction
GO 277 Heartbeat – Trigger	1.001	1 bit	To KNX

Telegram rate limitation

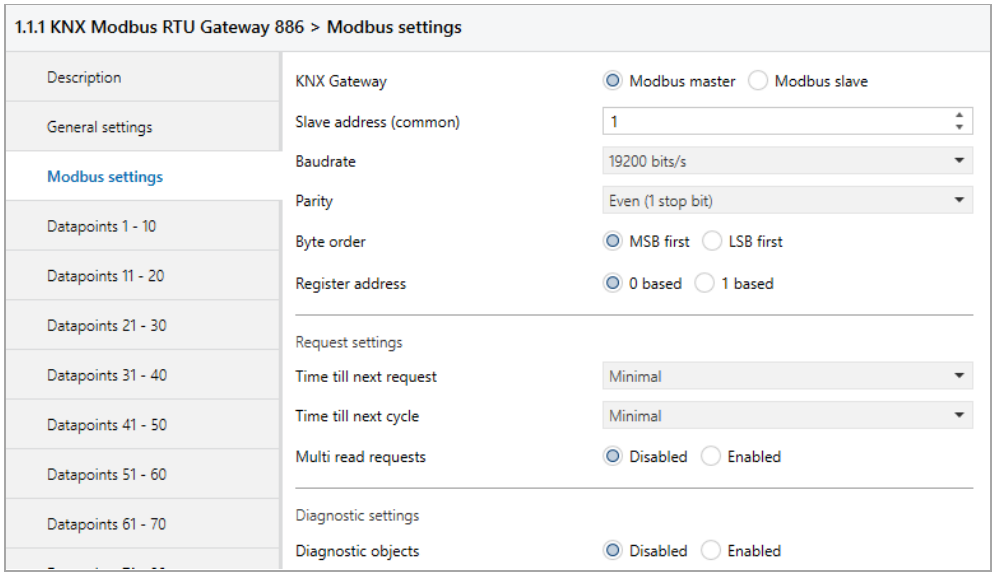
With this parameter the telegram rate limitation can be activated or deactivated.

Time telegram rate

If telegram rate limitation is activated, the time between the telegrams can be configured here. Times between 0.1 sec. and 1.0 sec. can be selected.

The telegram rate limitation only occurs when the bus load is increased.

5.3 Modbus settings



KNX Gateway

This parameter defines the role of the KNX gateway within the client/server architecture of the Modbus environment. Available are:

- Modbus master
- Modbus slave

Slave address (common)

Here the general slave address (0 ... 247) is set and used according to KNX Gateway configuration.

Modbus master:
The slave address of the Modbus communication partner is entered here. If several Slave devices are to be addressed, a specific slave address can be assigned per parameter page (e.g. datapoints 1-10).

Modbus slave:
The slave address of the KNX gateway is entered here.

Baudrate

Configures the baudrate of the Modbus communication. The following are available:

- 1200 bits/s
- 2400 bits/s
- 4800 bits/s
- 9600 bits/s
- 19200 bits/s
- 38400 bits/s
- 56000 bits/s
- 115200 bits/s

Parity

Here the Modbus frame is parameterized with regard to parity and stop bit. The following options are available:

- Even (1 stop bit)
- Odd (1 stop bit)
- None (2 stop bits)
- None (1 stop bit)

Byte order

Defines the order for the transmission of 2 byte values:

- MSB first (high byte is sent first)
- LSB first (low byte is sent first)

Register address

Here it is set on which address basis the register address is defined:

- 0 based
- 1 based

Time till next request *(only in Master mode)*

This parameter defines the minimum time for the next request. Used to slow down the master.

Time till next cycle *(only in Master Mode)*

This parameter defines the time after which a new request cycle has to begin. If the time is too short, the cycle will not be completed. This is indicated by the red flashing of the RTU LED 6.

Multi read requests *(only in Master Mode)*

When this parameter is activated, Modbus registers which lie one after the other in the channels are combined in a multi-read request.

The slave address and the function code must be identical. The register address must be continuous, but repetitions may occur. A maximum of 16 channels can be combined.

Example:

Type Slave address: Common
Channel 1 – MB to KNX – Read holding register – Address 0
Channel 2 – MB to KNX – Read holding register – Address 1
Channel 3 – MB to KNX – Read holding register – Address 2
Channel 4 – MB to KNX – Read holding register – Address 2
Channel 5 – MB to KNX – Read holding register – Address 3
Channel 6 – MB to KNX – Read holding register – Address 4
Channel 7 – MB to KNX – Read holding register – Address 4

The result is a read-multi-holding-register request for addresses 0 – 4.

Diagnostic objects

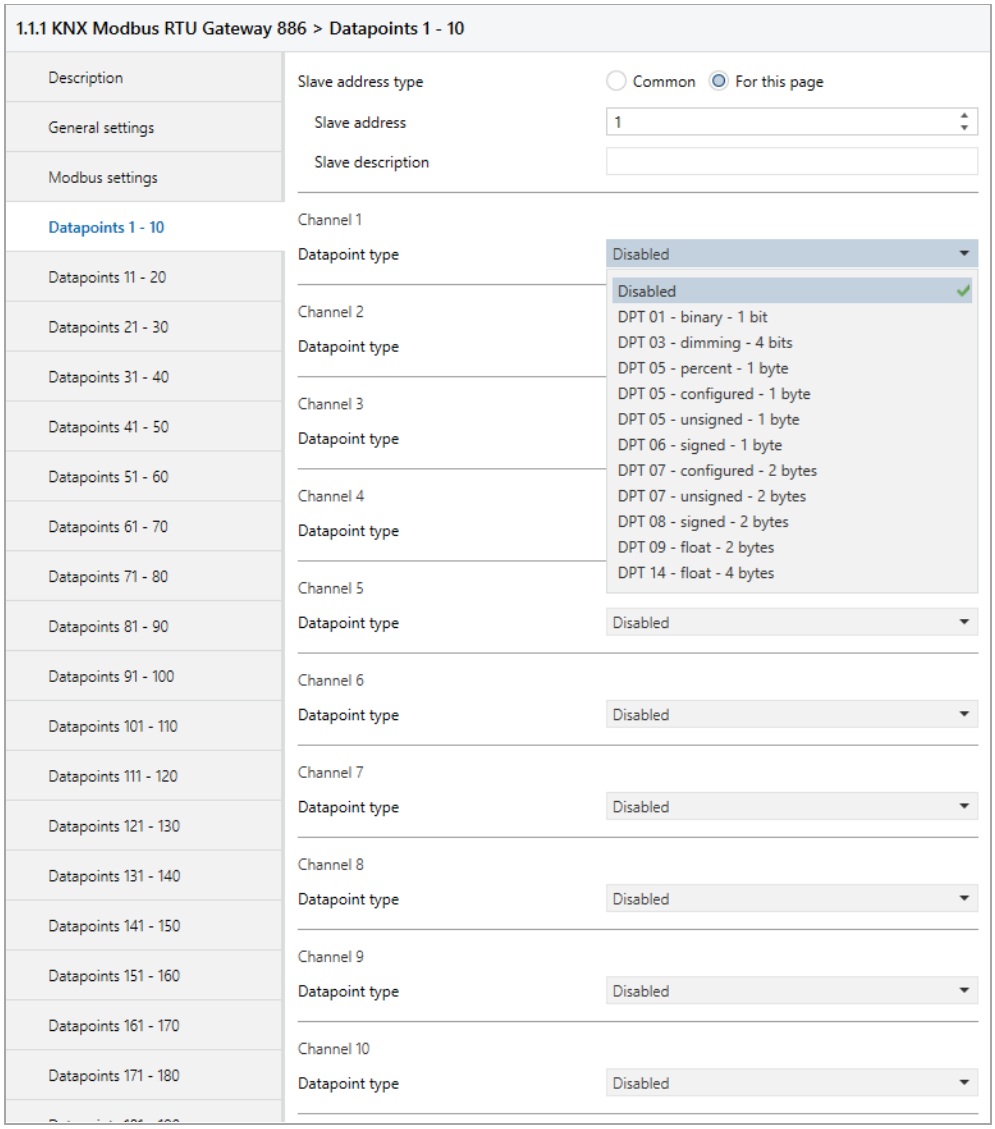
Activated, objects for diagnostic are displayed and provide information about the communication with the respective slave device.

Modbus master:
If no response is received from the slave device, “No communication – On” is sent to the KNX bus via the diagnostic object.

Modbus slave:
If no request is received from the master within the time interval, “No communication – On” is sent to the KNX bus via the diagnostic object.

Group object	Type KNX	Size	Direction
GO 276 Diagnostic: Slave (common) – No communication	1.001	1 bit	To KNX

5.4 Datapoints N – M



10 channels are combined per page.

Slave address type *(only in master mode)*

This parameter determines whether the general address or another slave address is to be used for the configured channels on this page.

Slave address *(only in master mode)*

Here the slave address (0 ... 247) of the channels of this page is set.

Slave description *(only in master mode)*

If the diagnostic is activated, the diagnostic object of this page can be named here.

Group object	Type KNX	Size	Direction
GO 251 Diagnostic: Slave (page 1) – No communication	1.001	1 bit	To KNX

Datapoint type

This parameter activates and defines the KNX interface and the function of this channel. The following options are available:

- Disabled
- DPT 01 – binary – 1 bit

Group object	Type KNX	Size	Direction
GO 1 Channel 1: … – Switch – 1 bit	1.001	1 bit	…

- DPT 03 – dimming – 4 bits

Group object	Type KNX	Size	Direction
GO 1 Channel 1: … – Dimming – 4 bits	3.007	4 bits	…

- DPT 05 – percent – 1 byte

Group object	Type KNX	Size	Direction
GO 1 Channel 1: … – Percent – 1 byte	5.001	1 byte	…

- DPT 05 – configured – 1 byte

Group object	Type KNX	Size	Direction
GO 1 Channel 1: … – Configured – 1 byte	5.010	1 byte	…

- DPT 05 – unsigned – 1 byte

Group object	Type KNX	Size	Direction
GO 1 Channel 1: … – Unsigned – 1 byte	5.010	1 byte	…

- DPT 06 – signed – 1 byte

Group object	Type KNX	Size	Direction
GO 1 Channel 1: … – Signed – 1 byte	6.010	1 byte	…

- DPT 07 – configured – 2 bytes

Group object	Type KNX	Size	Direction
GO 1 Channel 1: … – Configured – 2 bytes	7.001	2 bytes	…

- DPT 07 – unsigned – 2 bytes

Group object	Type KNX	Size	Direction
GO 1 Channel 1: … – Unsigned – 2 bytes	7.001	2 bytes	…

- DPT 08 – signed – 2 bytes

Group object	Type KNX	Size	Direction
GO 1 Channel 1: … – Signed – 2 bytes	8.001	2 bytes	…

- DPT 09 – float – 2 bytes

Group object	Type KNX	Size	Direction
GO 1 Channel 1: … – Float – 2 bytes	9.001	2 bytes	…

- DPT 14 – float – 4 bytes

Group object	Type KNX	Size	Direction
GO 1 Channel 1: … – Float – 4 bytes	14.000	4 bytes	…

Description *(30 characters)*

Any name can be assigned to the channel. However, this should be unique and meaningful, which makes it easier to work with the associated group objects later, since the name assigned is displayed there as a description. If no name is assigned, the group objects are designated as “Channel N: ...”.

Direction

The KNX specific communication direction is set here:

- KNX to Modbus *(group object is input)*

Group object	Type KNX	Size	Direction
GO 1 Channel 1: Input – …	Depending on DPT		From KNX

- Modbus to KNX *(group object is output)*

Group object	Type KNX	Size	Direction
GO 1 Channel 1: Output – …	Depending on DPT		To KNX

Send condition

If the group object is defined as an output, the time at which the object sends the value to the KNX bus is configured here. The following options are available:

- Read only
  Object sends only on read requests
- On change
  Object sends on value change
- Cyclic
  Object sends after cycle time
- Cyclic and on change
  Object transmits after cycle time and on value change

Cycle time

The time for send condition cyclic.

Type

This parameter defines the function of the channel as well as the size of the Modbus register used.

Depending on the parameter **Datapoint type N** various channel functions are possible, which will be discussed in more detail in the following chapters.

For “Value in word register” it should be noted that the channel does not work in case of incorrect configuration.

*The **Bit count** and **Offset** together must not be greater than 16.*
*The value must fit into **Bit count**, for example **Bit count** = 1 → “Value” = 0 or 1.*

Function

The Modbus function code for this channel is configured here.

Different function codes can be configured depending on **KNX gateway** (Modbus master/slave), **Direction** and **Type**.

Word register:

Modbus Master | KNX to Modbus
- Write single holding register – 06
- Write multi holding registers – 16
  (for “DPT 14” and “Double word register”)

Modbus Master | Modbus to KNX
- Read holding registers – 03
- Read input registers – 04

Modbus Slave | KNX to Modbus
- Read holding registers – 03
- Read input registers – 04

Modbus Slave | Modbus to KNX
- Write single/multi holding registers – 06, 16

Bit Register:

Modbus Master | KNX to Modbus
- Write single coil – 05

Modbus Master | Modbus to KNX
- Read coils – 01
- Read discrete inputs – 02

Modbus Slave | KNX to Modbus
- Read coils – 01
- Read discrete inputs – 02

Modbus Slave | Modbus to KNX
- Write single/multi coils – 05, 15

Address

Here the address of the Modbus register is configured. An address range from 0 ... 65535 is available.

If address 0 is configured with “1 based”, this is a static error which deactivates the channel function and is indicated by LED RTU 6 lighting up in red.

Polling interval *(only in Master mode and for Modbus to KNX)*

This defines the cyclic intervals of read requests for the respective register. The following options are available:

- Every cycle
- Every second cycle
- Every fourth cycle
- Every sixth cycle
- Every eighth cycle

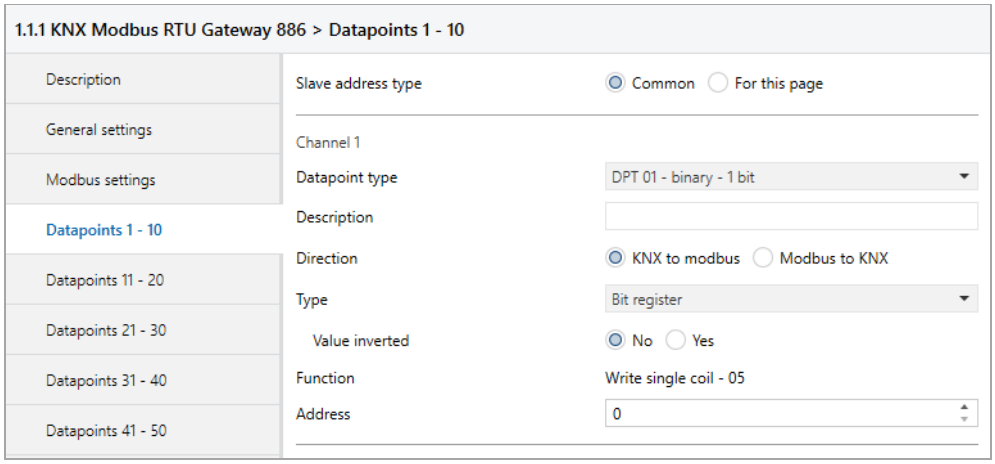
5.5 Channel function “DPT 01 – binary – 1 bit”

Type

The following types are configurable:

- Bit register
  1 bit (KNX) sets bit register (Modbus)
- Bit in word register
  1 bit (KNX) sets 1 bit in word register (Modbus)
- Value in word register
  1 bit (KNX) is mapped to value in word register (Modbus)

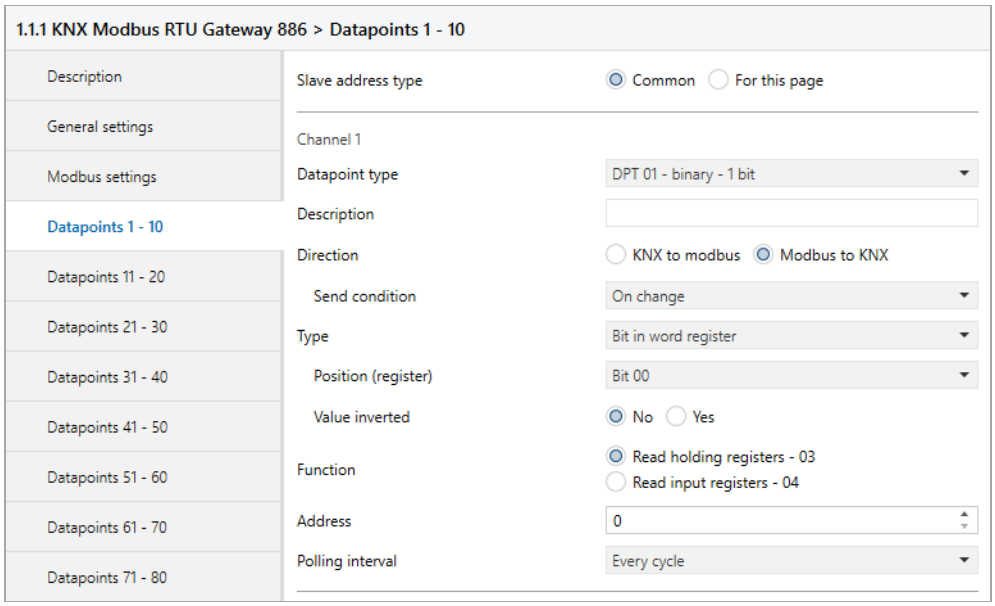
5.5.1 Type – Bit register



Value inverted

If yes, the inverted value of the group object corresponds to the value of the bit register.

5.5.2 Type – Bit in word register



Position (register)

Defines the bit in the word register.

Value inverted

If yes, the inverted value of the group object corresponds to the value of the bit in the word register.

5.5.3 Type – Value in word register

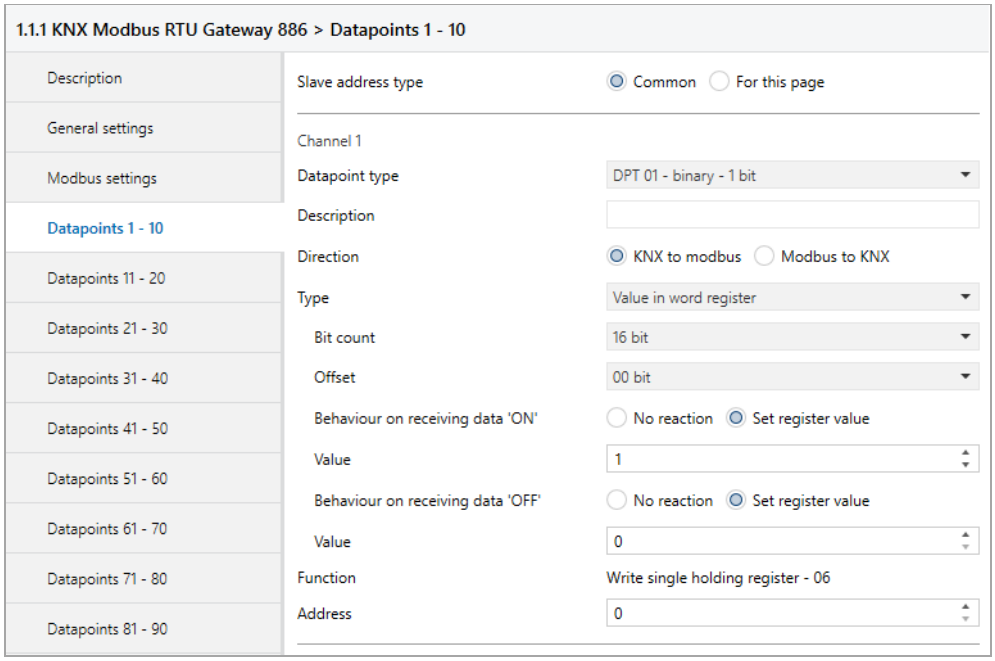
Bit count

This parameter defines the size of the value in the word register (in bits).

Offset

This parameter defines the position of the value in the word register (offset from the right in bits).

5.5.3.1 Direction – KNX to Modbus



Behaviour on receiving data ‘ON’

Here it can be configured whether a value is to be set in the register when an ‘ON’ telegram is received.

Value

The value that is set in the register when an ‘ON’ telegram is received.

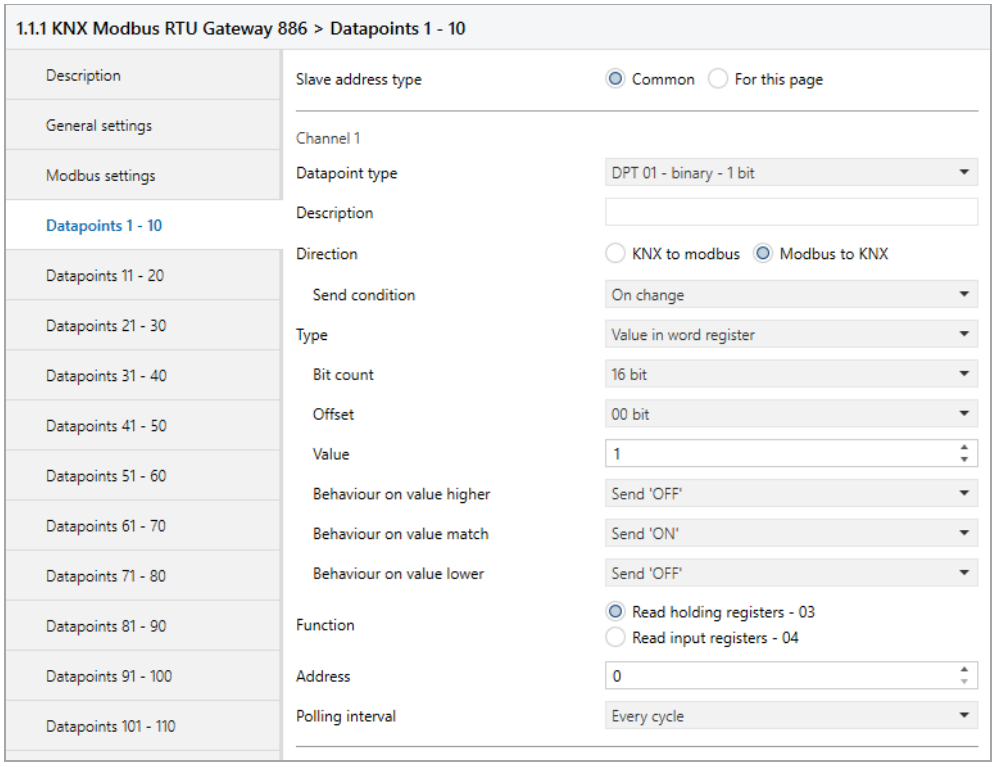
Behaviour on receiving data ‘OFF’

Here it can be configured whether a value is to be set in the register when an ‘OFF’ telegram is received.

Value

The value that is set in the register when an ‘OFF’ telegram is received.

5.5.3.2 Direction – Modbus to KNX



Value

The value for which the register is checked. Depends on **Bit count** and **Offset**.

Behaviour on value higher

This parameter defines the behaviour of the group object if the register value is greater than the parameterized value. The following options are available:

- No reaction
- Send ‘ON’
- Send ‘OFF’

Behaviour on value match

This parameter defines the behaviour of the group object in case the register value corresponds to the parameterized value. The following options are available:

- No reaction
- Send ‘ON’
- Send ‘OFF’

Behaviour on value lower

This parameter defines the behaviour of the group object in case the register value is smaller than the parameterized value. The following options are available:

- No reaction
- Send ‘ON’
- Send ‘OFF’

5.6 Channel function “DPT 03 – dimming – 4 bit”

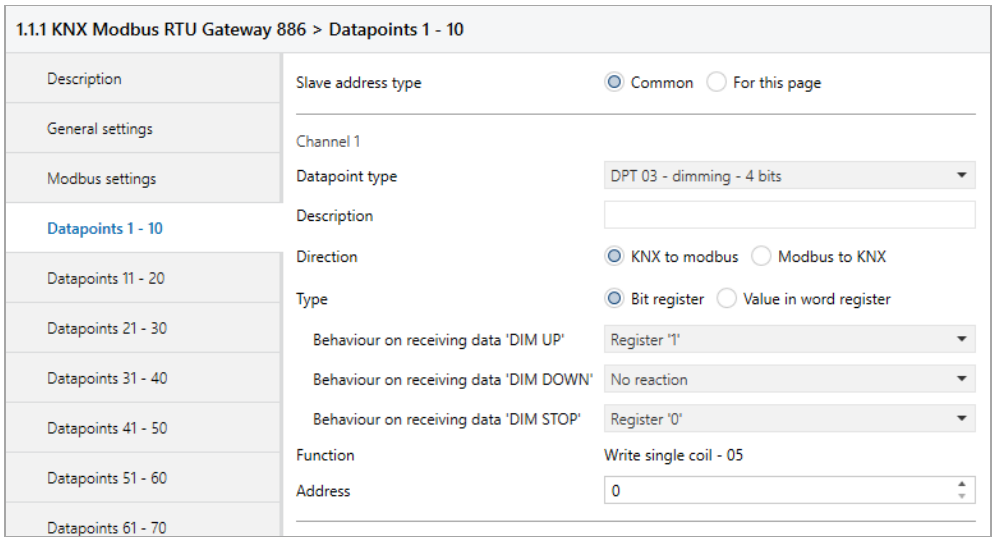
Type

The following types are configurable:

- Bit register
  4 bit dimming command (KNX) sets bit register (Modbus)
- Value in word register
  4 bit dimming command (KNX) is mapped to value in word register (Modbus)

5.6.1 Type – Bit Register

5.6.1.1 Direction – KNX to Modbus



Behaviour on receiving data ‘DIM UP’

This parameter defines the behaviour of the bit register in case a ‘DIM UP’ telegram has been received at the group object. The following options are available:

- No reaction
- Register ‘1’
- Register ‘0’

Behaviour on receiving data ‘DIM DOWN’

This parameter defines the behaviour of the bit register in case a ‘DIM DOWN’ telegram was received at the group object. The following options are available:

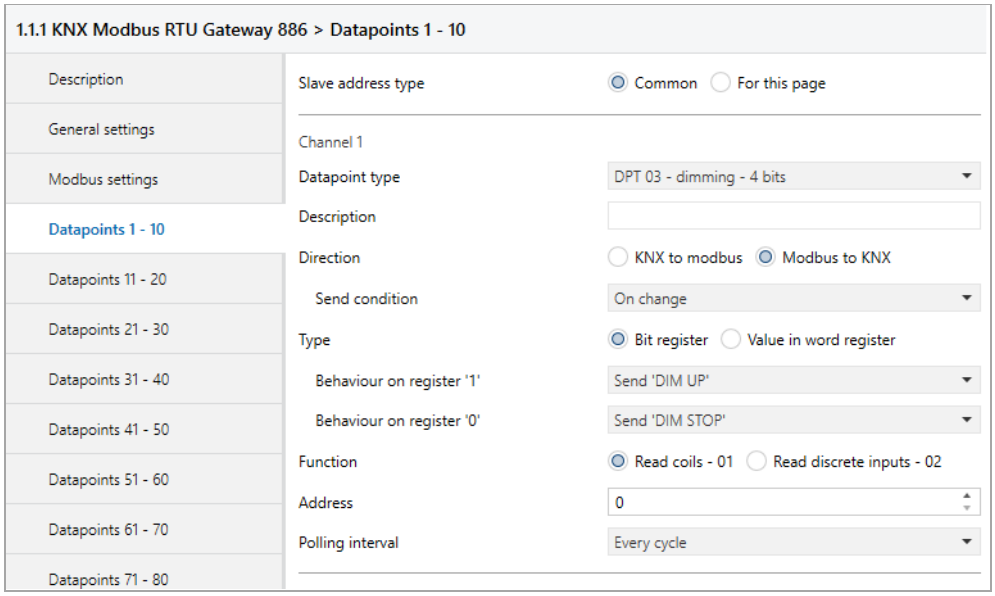
- No reaction
- Register ‘1’
- Register ‘0’

Behaviour on receiving data ‘DIM STOP’

This parameter defines the behaviour of the bit register in case a ‘DIM STOP’ telegram has been received at the group object. The following options are available:

- No reaction
- Register ‘1’
- Register ‘0’

5.6.1.2 Direction – Modbus to KNX



Behaviour on register ‘1’

This parameter defines the behaviour of the group object in case the register value corresponds to the parameterized value. The following options are available:

- No reaction
- Send ‘DIM UP’
- Send ‘DIM DOWN’
- Send ‘DIM STOP’

Behaviour on register ‘0’

This parameter defines the behaviour of the group object in case the register value corresponds to the parameterized value. The following options are available:

- No reaction
- Send ‘DIM UP’
- Send ‘DIM DOWN’
- Send ‘DIM STOP’

5.6.2 Type – Value in word register

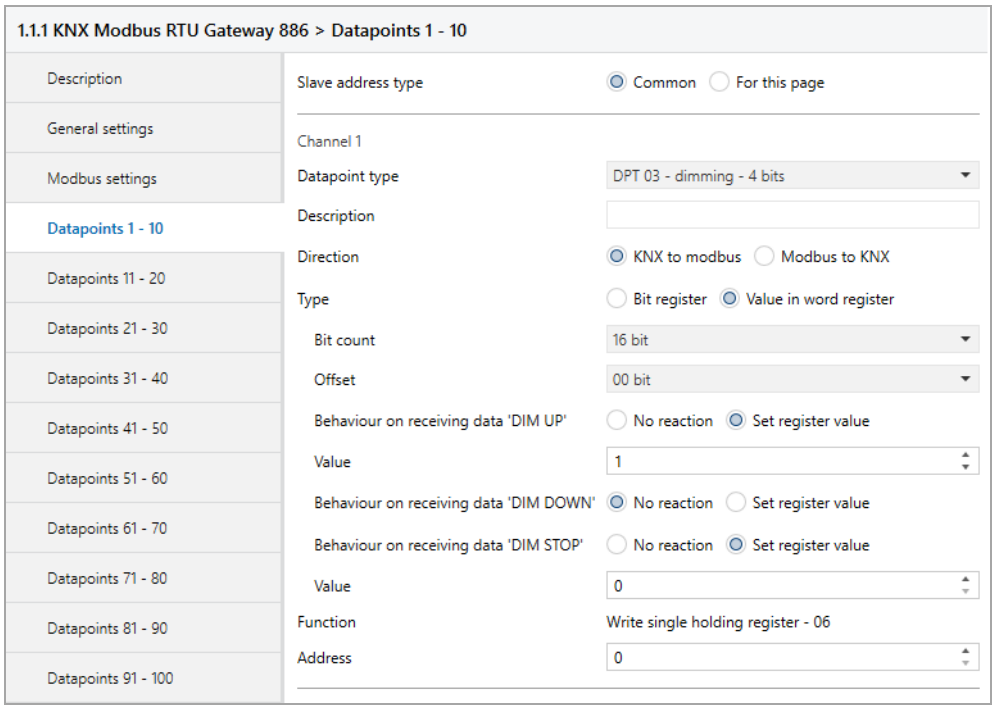
Bit count

This parameter defines the size of the value in the word register (in bits).

Offset

This parameter defines the position of the value in the word register (offset from the right in bits).

5.6.2.1 Direction – KNX to Modbus



Behaviour on receiving data ‘DIM UP’

Here it can be parameterized whether a value is to be set in the register when a ‘DIM OPEN’ telegram is received.

Value

The value that is set in the register when a ‘DIM OPEN’ telegram is received.

Behaviour on receiving data ‘DIM DOWN’

Here it can be parameterized whether a value is to be set in the register when a ‘DIM DOWN’ telegram is received.

Value

The value that is set in the register when a ‘DIMM DOWN’ telegram is received.

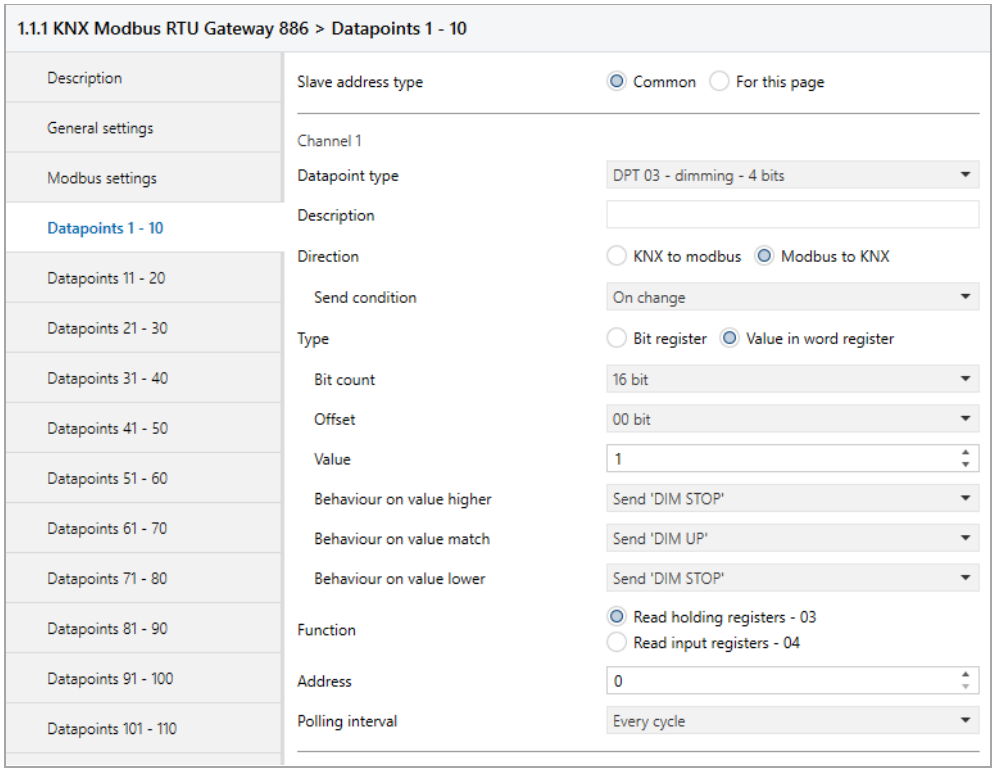
Behaviour on receiving data ‘DIM STOP’

Here it can be parameterized whether a value is to be set in the register when a ‘DIM STOP’ telegram is received.

Value

The value that is set in the register when a ‘DIM STOP’ telegram is received.

5.6.2.2 Direction – Modbus to KNX



Behaviour on value higher

This parameter defines the behaviour of the group object if the register value is greater than the parameterized value. The following options are available:

- No reaction
- Send ‘DIM UP’
- Send ‘DIM DOWN’
- Send ‘DIM STOP’

Behaviour on value match

This parameter defines the behaviour of the group object in case the register value corresponds to the parameterized value. The following options are available:

- No reaction
- Send ‘DIM UP’
- Send ‘DIM DOWN’
- Send ‘DIM STOP’

Behaviour on value lower

This parameter defines the behaviour of the group object in case the register value is smaller than the parameterized value. The following options are available:

- No reaction
- Send ‘DIM UP’
- Send ‘DIM DOWN’
- Send ‘DIM STOP’

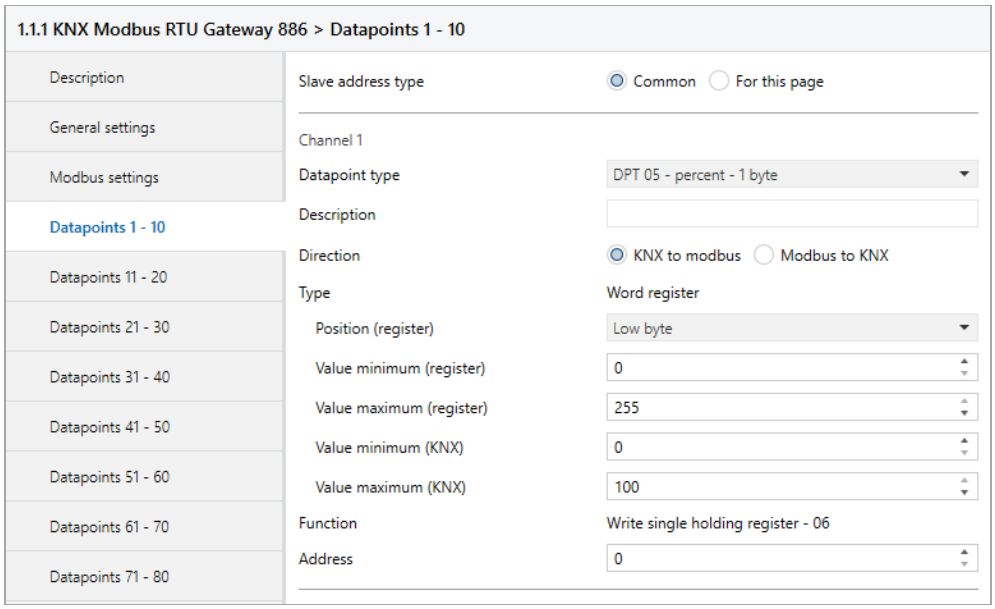
5.7 Channel function “DPT 05 – percent – 1 byte”

Type

The following type is configured:

- Word register
  1 byte percent value (KNX) is mapped to value in word register (Modbus)

5.7.1 Type – Word register



Position (register)

This parameter defines the area of the word register that is mapped. The following areas are available:

- Low byte
- High byte
- High/Low byte

Value minimum (register)

Register value which corresponds to **Value minimum (KNX)**.

Value maximum (register)

Register value which corresponds to **Value maximum (KNX)**.

Value minimum (KNX)

KNX value which corresponds to **Value minimum (register)**.

Value maximum (KNX)

KNX value, which corresponds to **Value maximum (register)**.

The conversion is always transferred to the entire register area.
***Value minimum/maximum (register)** defines no limits.*

5.8 Channel function “DPT 05 – configured – 1 byte”

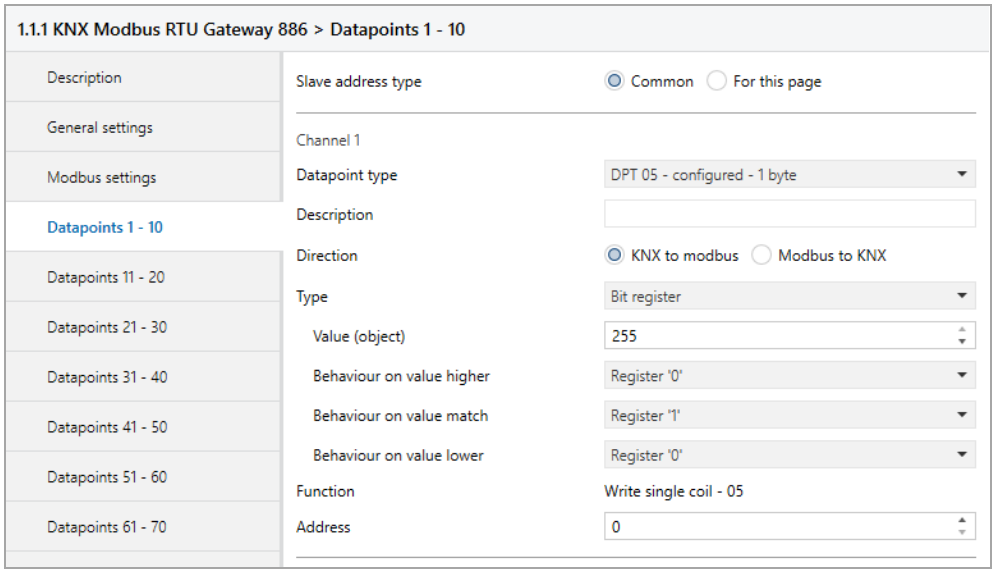
Type

The following types are configurable:

- Bit register
  1 byte configured value (KNX) sets bit register (Modbus)
- Bit in word register
  1 byte configured value (KNX) sets 1 bit in word register (Modbus)
- Value in word register
  1 byte configured value (KNX) is mapped to value in word register (Modbus)

5.8.1 Type – Bit register

5.8.1.1 Direction – KNX to Modbus



Value (object)

This parameter defines the value for which the group object (KNX) is checked.

Behaviour on value higher

This parameter defines the behaviour of the bit register in case the received value (KNX) is greater than the parameterized value. The following options are available:

- No reaction
- Register ‘1’
- Register ‘0’

Behaviour on value match

This parameter defines the behaviour of the bit register in case the received value (KNX) corresponds to the parameterized value. The following options are available:

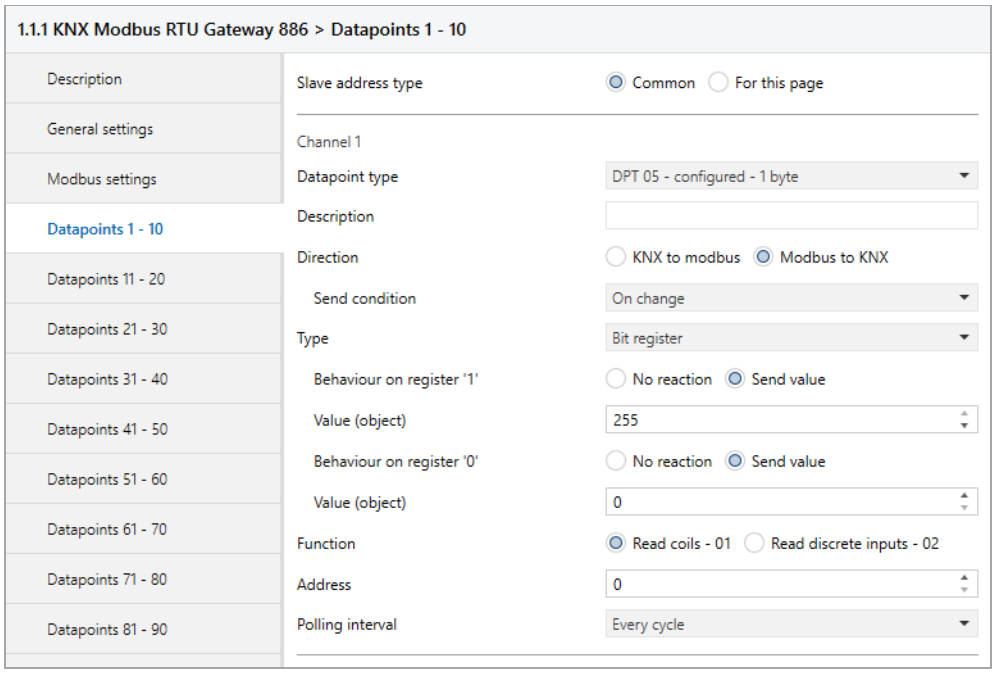
- No reaction
- Register ‘1’
- Register ‘0’

Behaviour on value lower

This parameter defines the behaviour of the bit register in case the received value (KNX) is smaller than the parameterized value. The following options are available:

- No reaction
- Register ‘1’
- Register ‘0’

5.8.1.2 Direction – Modbus to KNX



Behaviour on register ‘1’

This parameter defines the behaviour of the group object in case the register is set. The following options are available:

- No reaction
- Send value

Value (object)

The value that is sent to KNX when the register is set.

Behaviour on register ‘0’

This parameter defines the behaviour of the group object in case the register is not set. The following options are available:

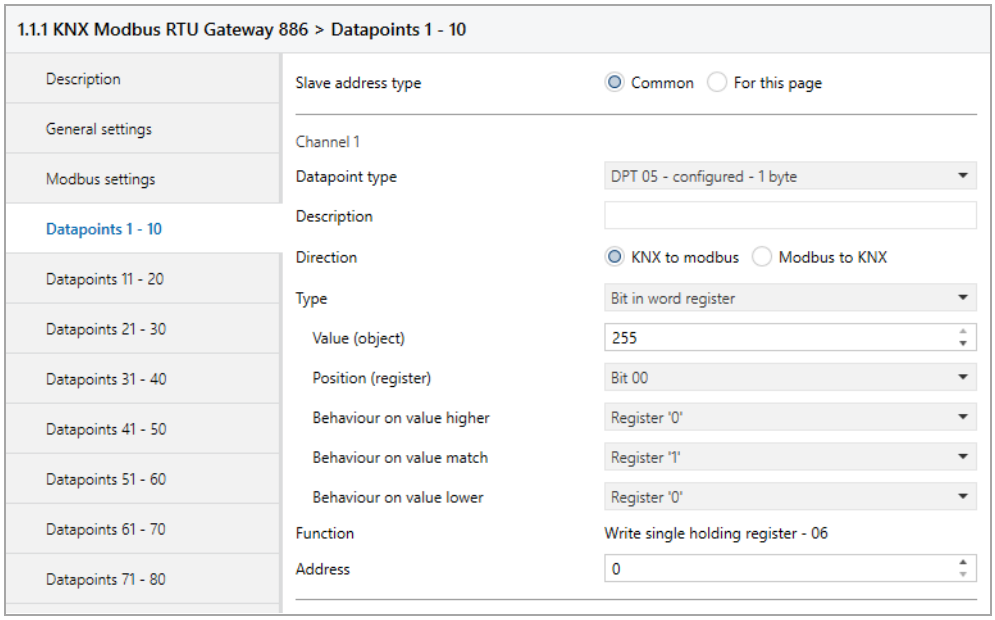
- No reaction
- Send value

Value (object)

The value that is sent to KNX when the register is not set.

5.8.2 Type – Bit in word register

5.8.2.1 Direction – KNX to Modbus



Value (object)

This parameter defines the value for which the group object (KNX) is checked.

Position (register)

Defines the bit in the word register.

Behaviour on value higher

This parameter defines the behaviour of the bit in the word register in case the received value (KNX) is greater than the parameterized value. The following options are available:

- No reaction
- Register ‘1’
- Register ‘0’

Behaviour on value match

This parameter defines the behaviour of the bit in the word register in case the received value (KNX) corresponds to the parameterized value. The following options are available:

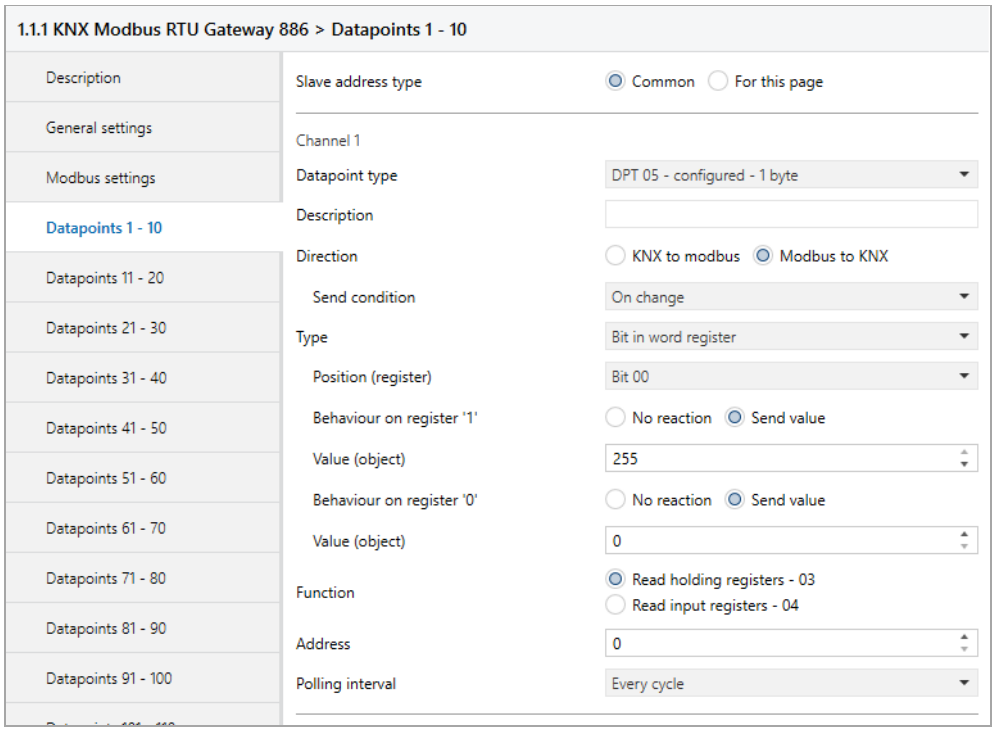
- No reaction
- Register ‘1’
- Register ‘0’

Behaviour on value lower

This parameter defines the behaviour of the bit in the word register in case the received value (KNX) is smaller than the parameterized value. The following options are available:

- No reaction
- Register ‘1’
- Register ‘0’

5.8.2.2 Direction – Modbus to KNX



Position (register)

Defines the bit in the word register.

Behaviour on register ‘1’

This parameter defines the behaviour of the group object in case the bit in the word register is set. The following options are available:

- No reaction
- Send value

Value (object)

The value that is sent to KNX when the bit in the word register is set.

Behaviour on register ‘0’

This parameter defines the behaviour of the group object in case the bit in the word register is not set. The following options are available:

- No reaction
- Send value

Value (object)

The value which is sent to KNX if the bit in the word register is not set.

5.8.3 Type – Value in word register

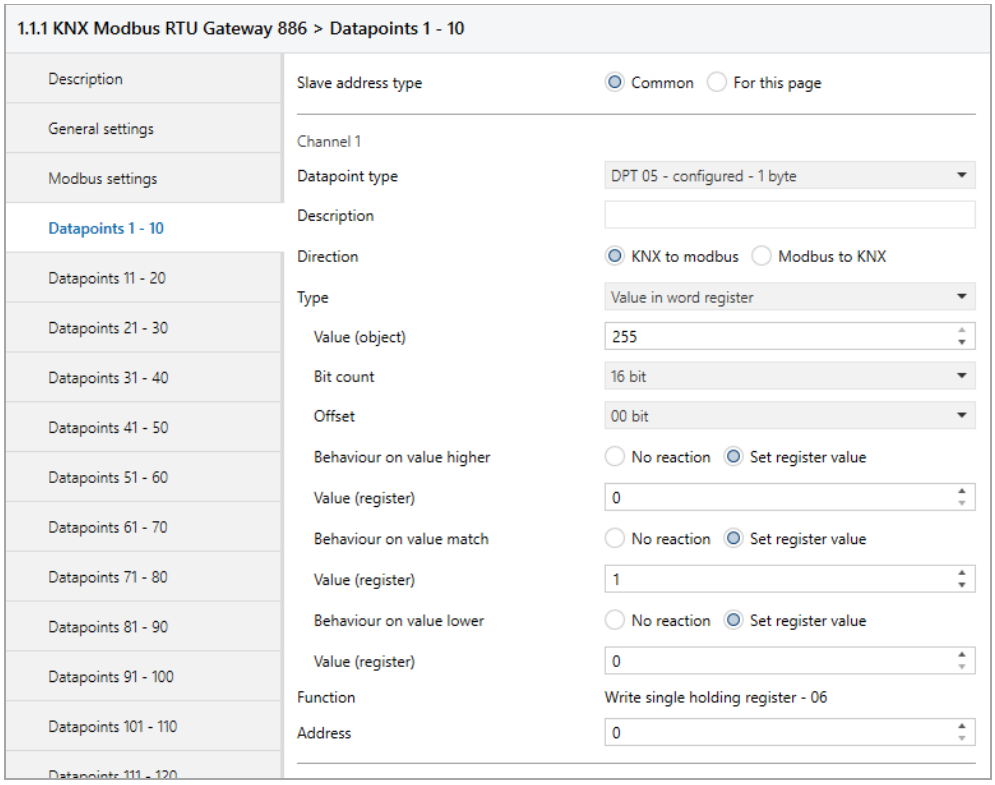
Bit count

This parameter defines the size of the value in the word register (in bits).

Offset

This parameter defines the position of the value in the word register (offset from the right in bits).

5.8.3.1 Direction – KNX to Modbus



Value (object)

This parameter defines the value for which the group object (KNX) is checked.

Behaviour on value higher

This parameter defines the behaviour of the word register in case the received value (KNX) is greater than the parameterized value. The following options are available:

- No reaction
- Set value in register

Value (register)

The value that is set in the word register.

Behaviour on value match

This parameter defines the behaviour of the word register in case the received value (KNX) corresponds to the parameterized value. The following options are available:

- No reaction
- Set value in register

Value (register)

The value that is set in the word register.

Behaviour on value lower

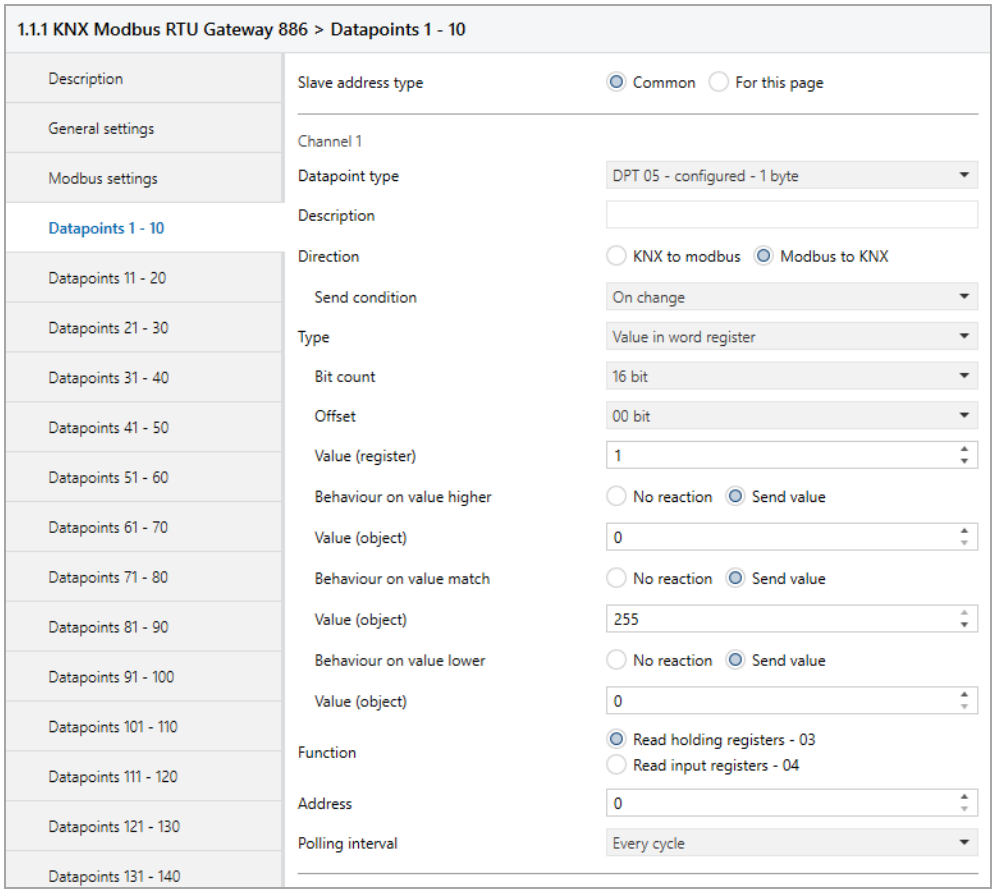
This parameter defines the behaviour of the word register in case the received value (KNX) is smaller than the parameterized value. The following options are available:

- No reaction
- Set value in register

Value (register)

The value that is set in the word register.

5.8.3.2 Direction – Modbus to KNX



Value (register)

This parameter defines the value for which the word register is checked.

Behaviour on value higher

This parameter defines the behaviour of the group object (KNX) in the event that the register value is greater than the parameterized value. The following options are available:

- No reaction
- Send value

Value (object)

The value that is sent via the group object (KNX).

Behaviour on value match

This parameter defines the behaviour of the group object (KNX) in the event that the register value corresponds to the parameterized value. The following options are available:

- No reaction
- Send value

Value (object)

The value that is sent via the group object (KNX).

Behaviour on value lower

This parameter defines the behaviour of the group object (KNX) in the event that the register value is smaller than the parameterized value. The following options are available:

- No reaction
- Send value

Value (object)

The value that is sent via the group object (KNX).

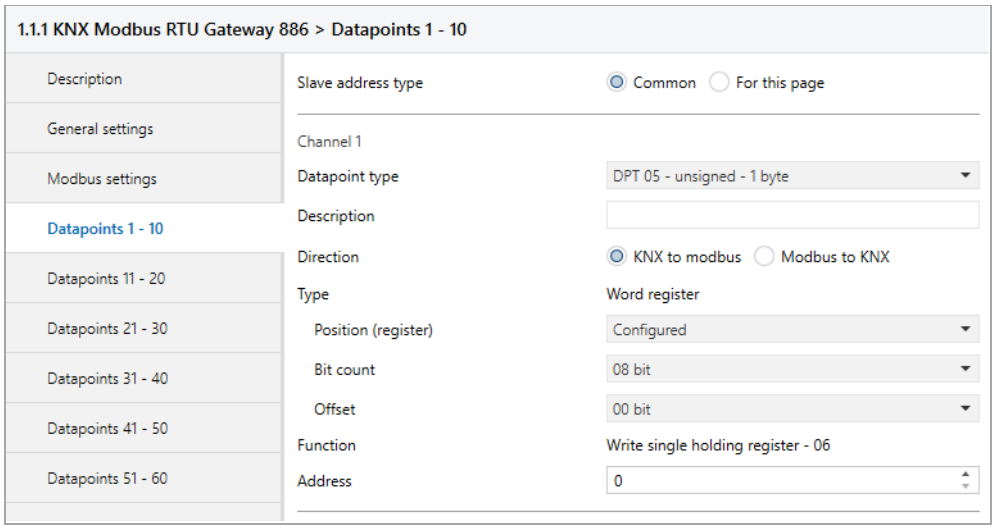
5.9 Channel function “DPT 05 – unsigned – 1 byte”

Type

The following type is configured:

- Word register
  1 byte value unsigned (KNX) is written/read to/from area in word register (Modbus)

5.9.1 Type – Word register



Position (register)

This parameter defines the area of the word register which is read/written. The following areas are available:

- Low byte
- High byte
- Configured

Bit count *(only for configured)*

This parameter defines the size of the area in the word register (in bits).

Offset *(only for configured)*

This parameter defines the position of the area in the word register (offset from the right in bits).

*The **Bit count** and **Offset** together must not be greater than 16.*
*The value must fit into **Bit count**, for example **Bit count** = 1 → "Value" = 0 or 1.*

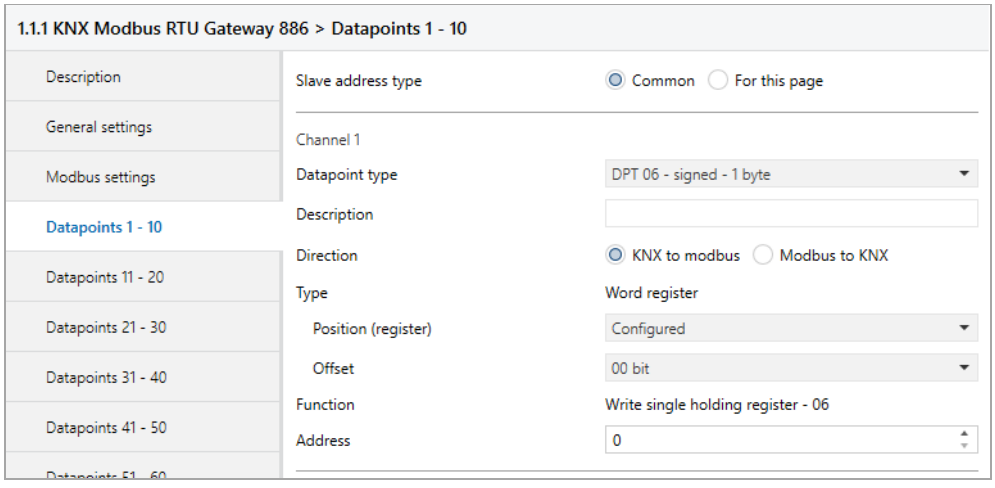
5.10 Channel function "DPT 06 – signed – 1 byte"

Type

The following type is configured:

- Word register
  1 byte value signed (KNX) is written/read to/from area in word register (Modbus)

5.10.1 Type – Word register



Position (register)

This parameter defines the area of the word register which is read/written. The following areas are available:

- Low byte
- High byte
- Configured

Offset *(only for configured)*

This parameter defines the position of the area in the word register (offset from the right in bits).

5.11 Channel function “DPT 07 – configured – 2 bytes”

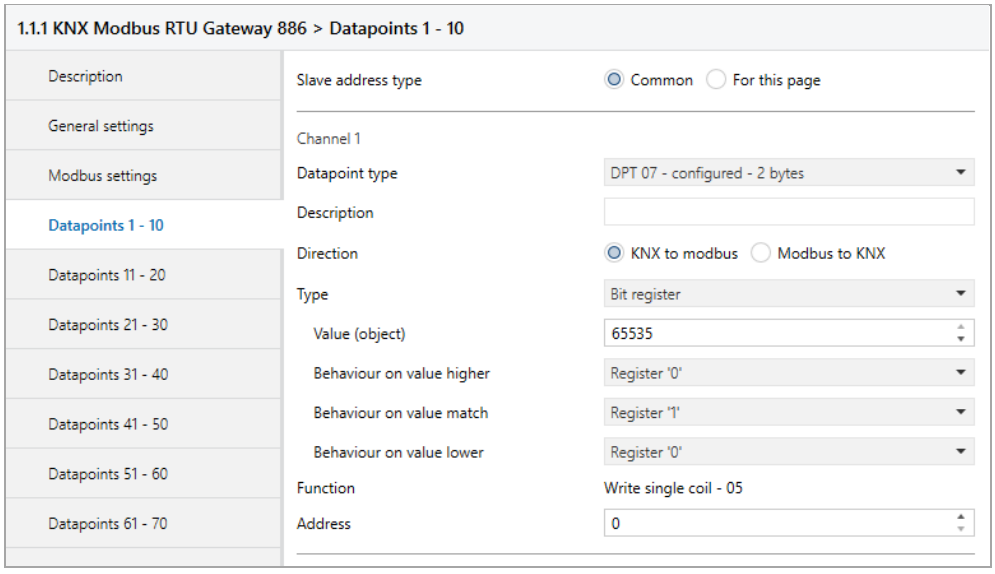
Type

The following types are configurable:

- Bit register
  2 byte value configured (KNX) sets bit register (Modbus)
- Bit in word register
  2 byte value configured (KNX) sets 1 bit in word register (Modbus)
- Value in word register
  2 byte value configured (KNX) is mapped to value in word register (Modbus)

5.11.1 Type – Bit register

5.11.1.1 Direction – KNX to Modbus



Value (object)

This parameter defines the value for which the group object (KNX) is checked.

Behaviour on value higher

This parameter defines the behaviour of the bit register in case the received value (KNX) is greater than the parameterized value. The following options are available:

- No reaction
- Register ‘1’
- Register ‘0’

Behaviour on value match

This parameter defines the behaviour of the bit register in case the received value (KNX) corresponds to the parameterized value. The following options are available:

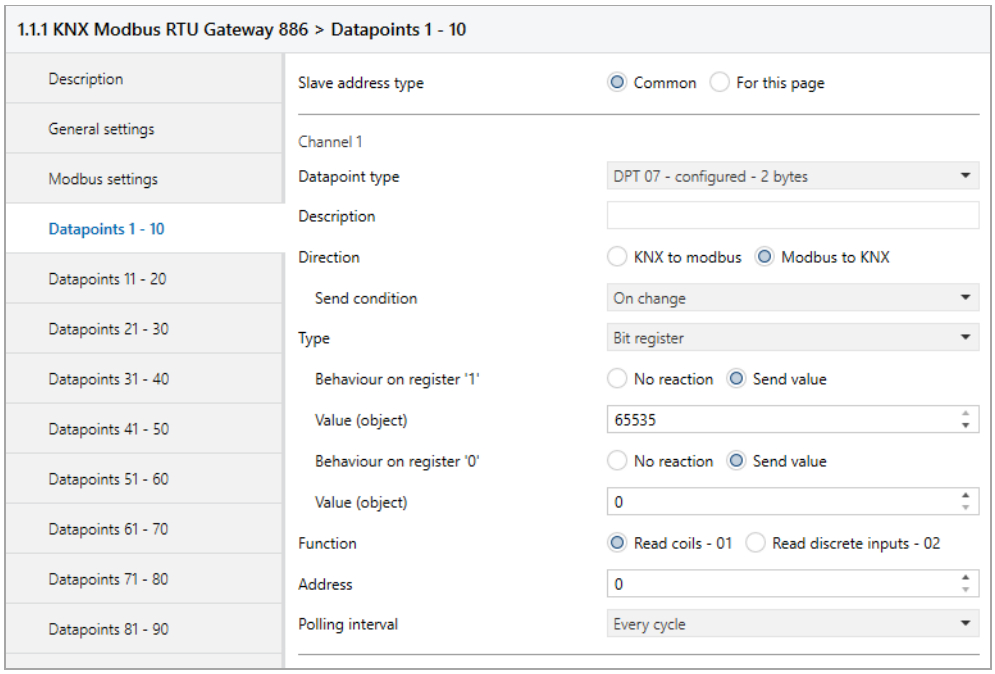
- No reaction
- Register ‘1’
- Register ‘0’

Behaviour on value lower

This parameter defines the behaviour of the bit register in case the received value (KNX) is smaller than the parameterized value. The following options are available:

- No reaction
- Register ‘1’
- Register ‘0’

5.11.1.2 Direction – Modbus to KNX



Behaviour on register ‘1’

This parameter defines the behaviour of the group object in case the register is set. The following options are available:

- No reaction
- Send value

Value (object)

The value that is sent to KNX when the register is set.

Behaviour with register ‘0’

This parameter defines the behaviour of the group object in case the register is not set. The following options are available:

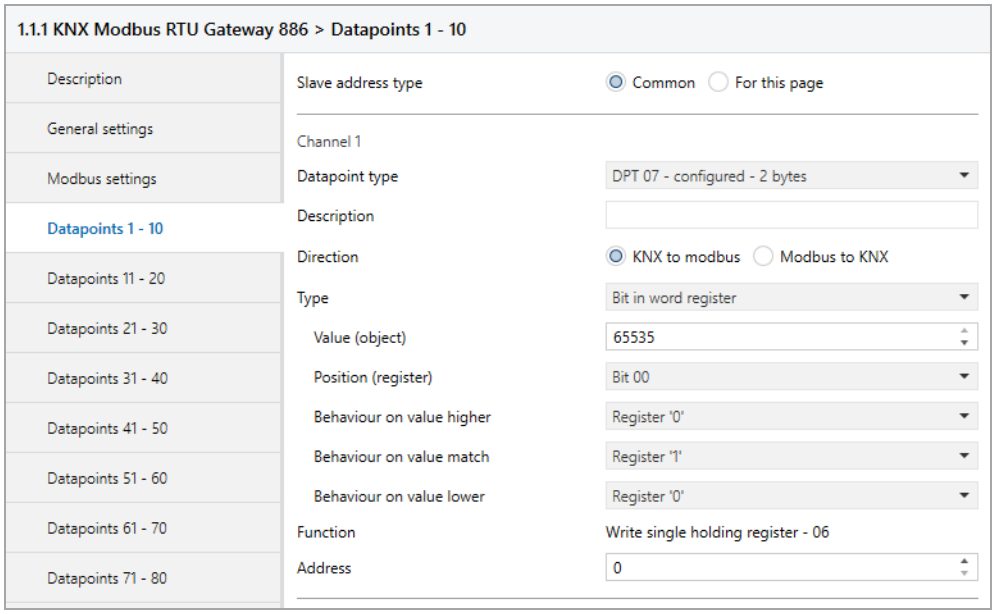
- No reaction
- Send value

Value (object)

The value that is sent to KNX when the register is not set.

5.11.2 Type – Bit in word register

5.11.2.1 Direction – KNX to Modbus



Value (object)

This parameter defines the value for which the group object (KNX) is checked.

Position (register)

Defines the bit in the word register.

Behaviour on value higher

This parameter defines the behaviour of the bit in the word register in case the received value (KNX) is greater than the parameterized value. The following options are available:

- No reaction
- Register ‘1’
- Register ‘0’

Behaviour on value match

This parameter defines the behaviour of the bit in the word register in case the received value (KNX) corresponds to the parameterized value. The following options are available:

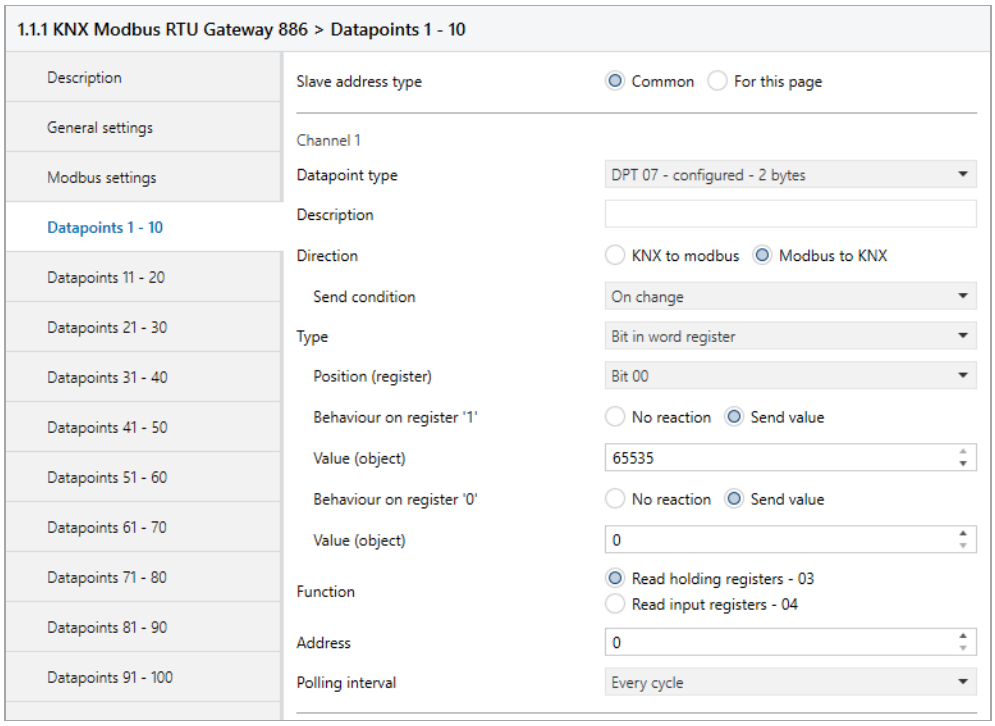
- No reaction
- Register ‘1’
- Register ‘0’

Behaviour on value lower

This parameter defines the behaviour of the bit in the word register in case the received value (KNX) is smaller than the parameterized value. The following options are available:

- No reaction
- Register ‘1’
- Register ‘0’

5.11.2.2 Direction – Modbus to KNX



Position (register)

Defines the bit in the word register.

Behaviour on register ‘1’

This parameter defines the behaviour of the group object in case the bit in the word register is set. The following options are available:

- No reaction
- Send value

Value (object)

The value that is sent to KNX when the bit in the word register is set.

Behaviour with register ‘0’

This parameter defines the behaviour of the group object in case the bit in the word register is not set. The following options are available:

- No reaction
- Send value

Value (object)

The value which is sent to KNX if the bit in the word register is not set.

5.11.3 Type – Value in word register

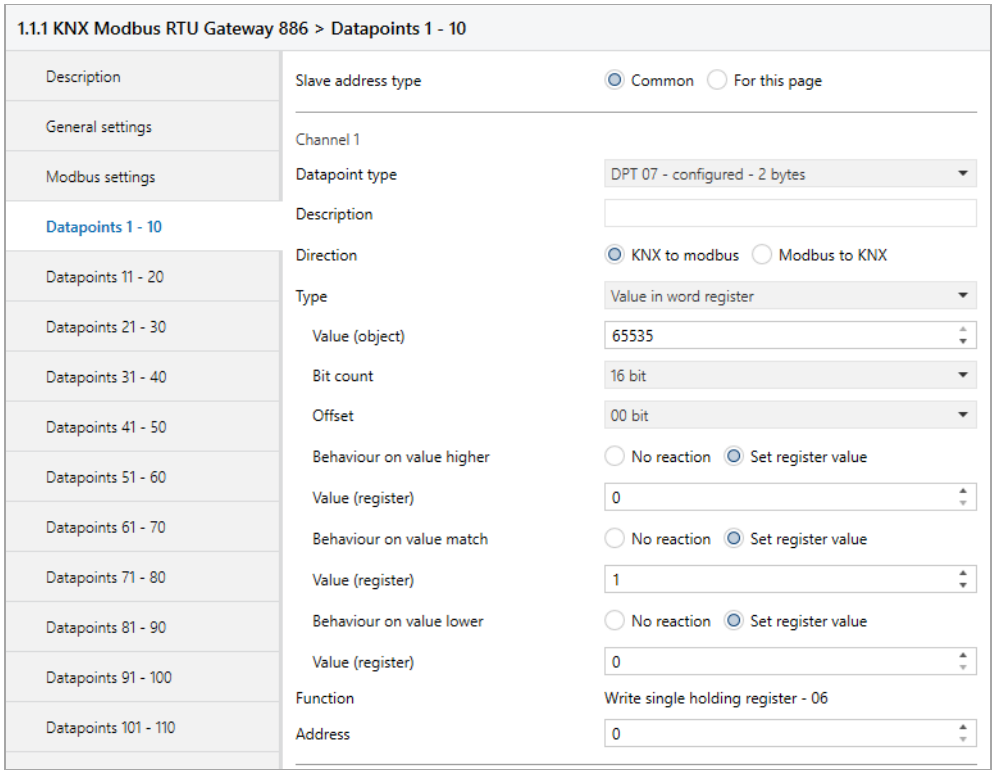
Bit count

This parameter defines the size of the value in the word register (in bits).

Offset

This parameter defines the position of the value in the word register (offset from the right in bits).

5.11.3.1 Direction – KNX to Modbus



Value (object)

This parameter defines the value for which the group object (KNX) is checked.

Behaviour on value higher

This parameter defines the behaviour of the word register in case the received value (KNX) is greater than the parameterized value. The following options are available:

- No reaction
- Set value in register

Value (register)

The value that is set in the word register.

Behaviour on value match

This parameter defines the behaviour of the word register in case the received value (KNX) corresponds to the parameterized value. The following options are available:

- No reaction
- Set value in register

Value (register)

The value that is set in the word register.

Behaviour on value lower

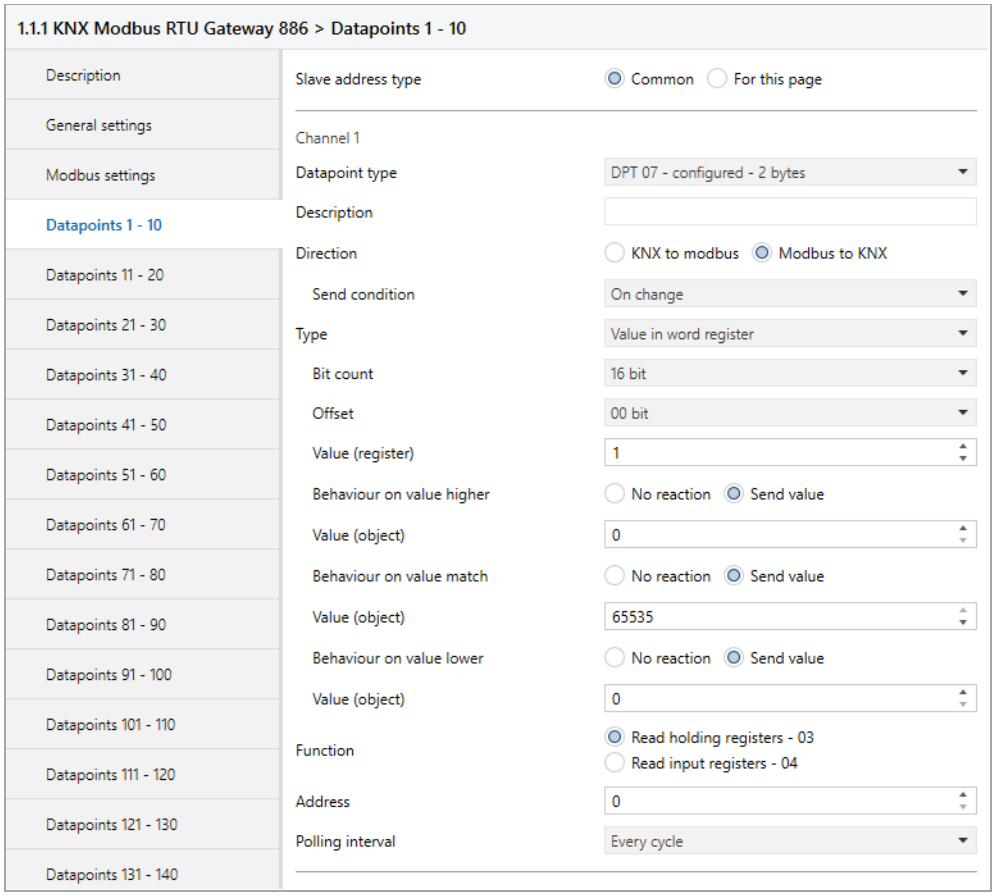
This parameter defines the behaviour of the word register in case the received value (KNX) is smaller than the parameterized value. The following options are available:

- No reaction
- Set value in register

Value (register)

The value that is set in the word register.

5.11.3.2 Direction – Modbus to KNX



Value (register)

This parameter defines the value for which the word register is checked.

Behaviour on value higher

This parameter defines the behaviour of the group object (KNX) in the event that the register value is greater than the parameterized value. The following options are available:

- No reaction
- Send value

Value (object)

The value that is sent via the group object (KNX).

Behaviour on value match

This parameter defines the behaviour of the group object (KNX) in the event that the register value corresponds to the parameterized value. The following options are available:

- No reaction
- Send value

Value (object)

The value that is sent via the group object (KNX).

Behaviour on value lower

This parameter defines the behaviour of the group object (KNX) in the event that the register value is smaller than the parameterized value. The following options are available:

- No reaction
- Send value

Value (object)

The value that is sent via the group object (KNX).

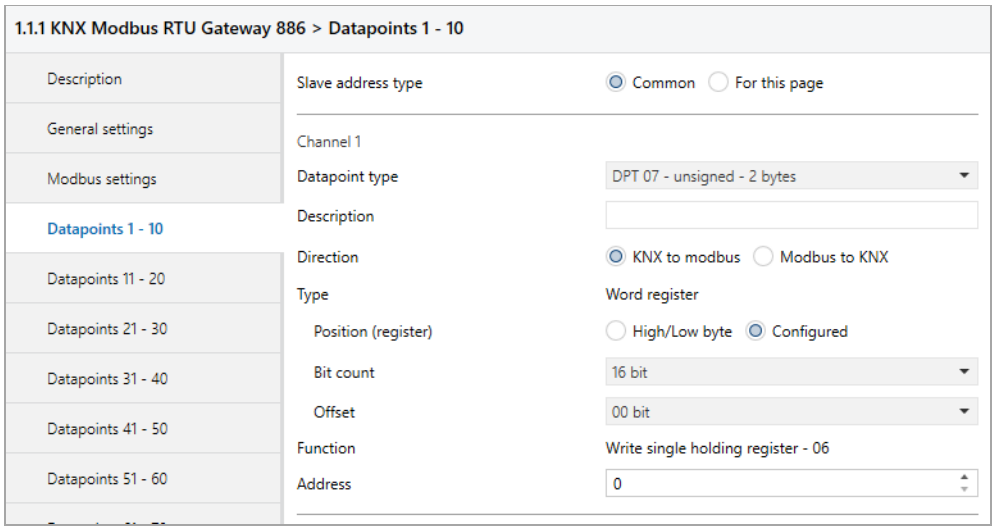
5.12 Channel function “DPT 07 – unsigned – 2 bytes”

Type

The following type is configured:

- Word register
  2 byte value unsigned (KNX) is written/read to/from area in word register (Modbus)

5.12.1 Type – Word register



Position (register)

This parameter defines the area of the word register which is read/written. The following areas are available:

- High/Low byte
- Configured

Bit count *(only for configured)*

This parameter defines the size of the area in the word register (in bits).

Offset *(only for configured)*

This parameter defines the position of the area in the word register (offset from the right in bits).

*The **Bit count** and **Offset** together must not be greater than 16.*
*The value must fit into **Bit count**, for example **Bit count** = 1 → "Value" = 0 or 1.*

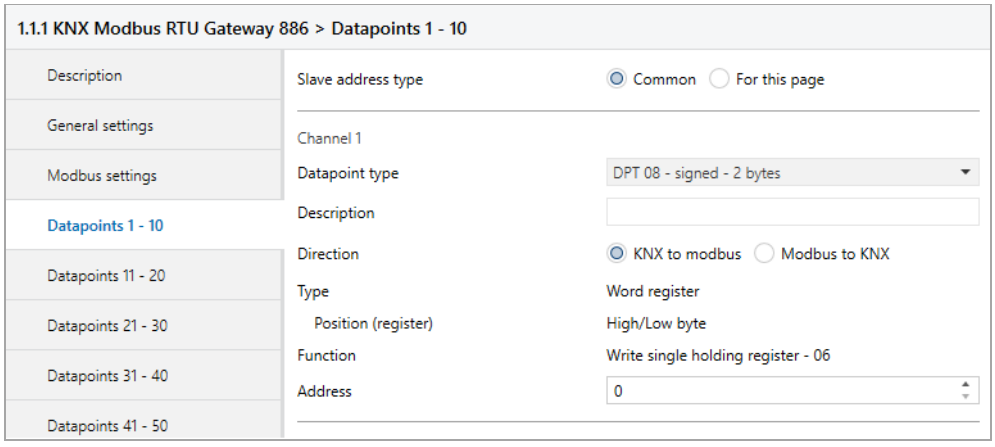
5.13 Channel function "DPT 08 – signed – 2 bytes"

Type

The following type is configured:

- Word register
  2 byte value signed (KNX) is written/read to/from area in word register (Modbus)

5.13.1 Type – Word register



Position (register)

This parameter defines the area of the word register which is read/written. The following area is configured:

- High/Low byte

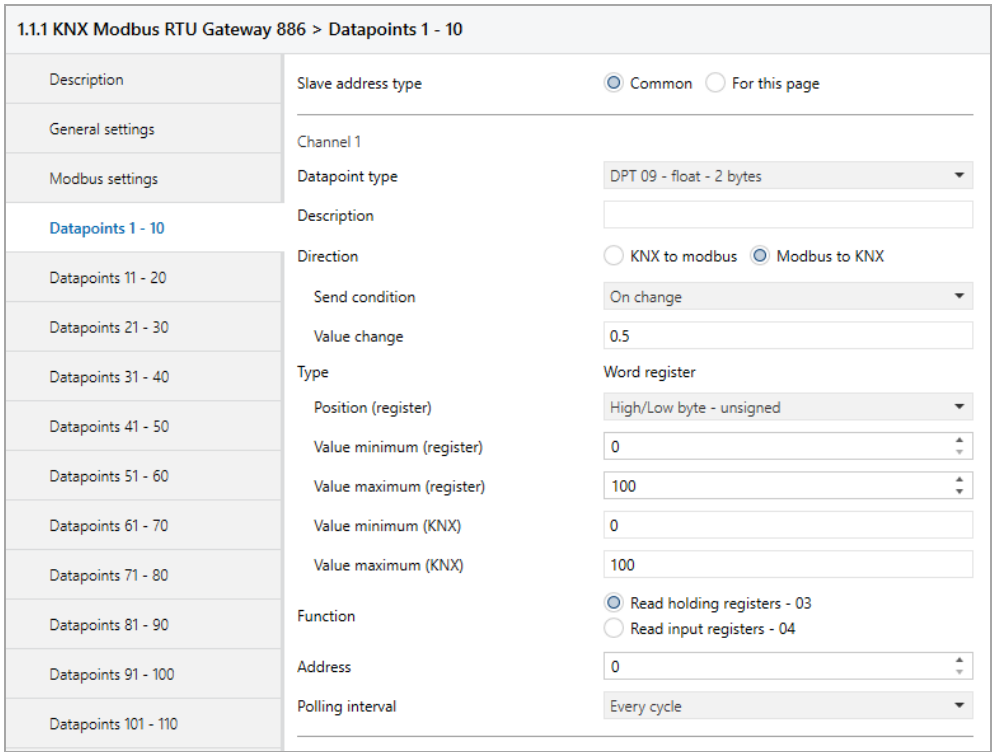
5.14 Channel Function “DPT 09 – float – 2 bytes”

Type

The following type is configured:

- Word register
  2 byte float value (KNX) is mapped to area in word register (Modbus)

5.14.1 Type – Word register



Position (register)

This parameter defines the area of the word register that is mapped. The following areas are available:

- Low byte – unsigned
- High byte – unsigned
- High/Low byte – unsigned
- Low byte – 2th complement
- High byte – 2th complement
- High/Low byte – 2th complement

Value minimum (register)

Register value which corresponds to **Value minimum (KNX)**.

Value maximum (register)

Register value which corresponds to **Value maximum (KNX)**.

Value minimum (KNX)

KNX value which corresponds to **Value minimum (register)**.

Value maximum (KNX)

KNX value, which corresponds to **Value maximum (register)**.

The conversion is always transferred to the entire register area.
***Value minimum/maximum (register)** defines no limits.*

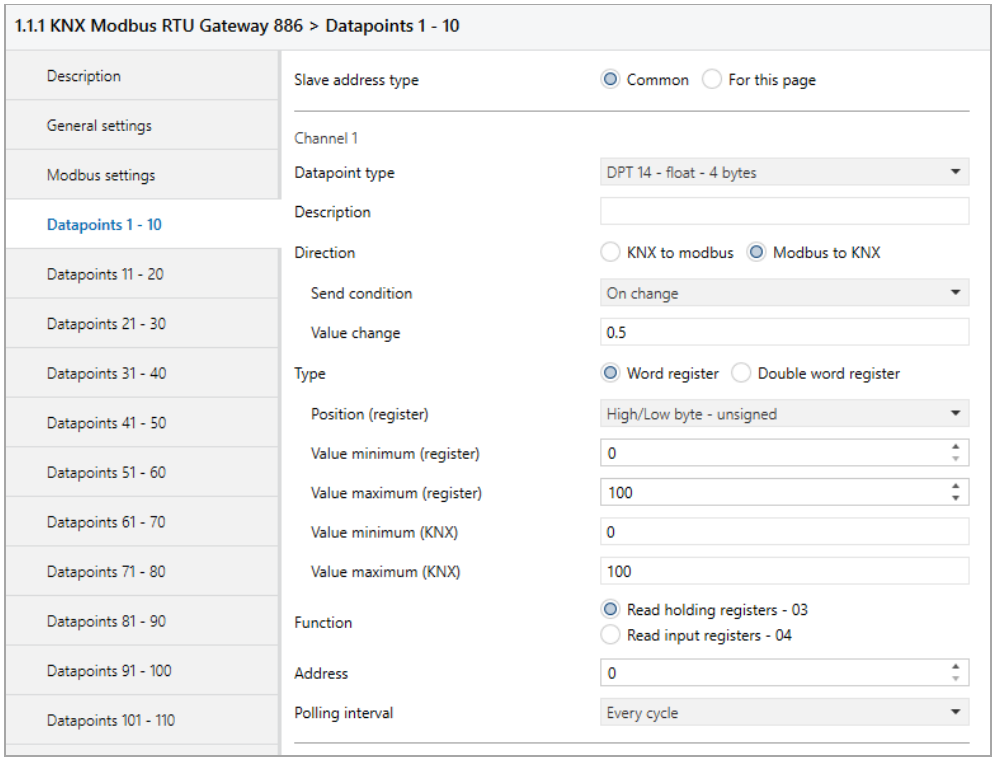
5.15 Channel Function “DPT 14 – float – 4 bytes”

Type

The following types are configurable:

- Word register
  4 byte float value (KNX) is mapped to area in word register (Modbus)
- Double word register
  4 byte float value (KNX) is mapped to two word registers (Modbus)

5.15.1 Type – Word register



Position (register)

This parameter defines the area of the word register that is mapped. The following areas are available:

- Low byte – unsigned
- High byte – unsigned
- High/Low byte – unsigned
- Low byte – 2th complement
- High byte – 2th complement
- High/Low byte – 2th complement

Value minimum (register)

Register value which corresponds to **Value minimum (KNX)**.

Value maximum (register)

Register value which corresponds to **Value maximum (KNX)**.

Value minimum (KNX)

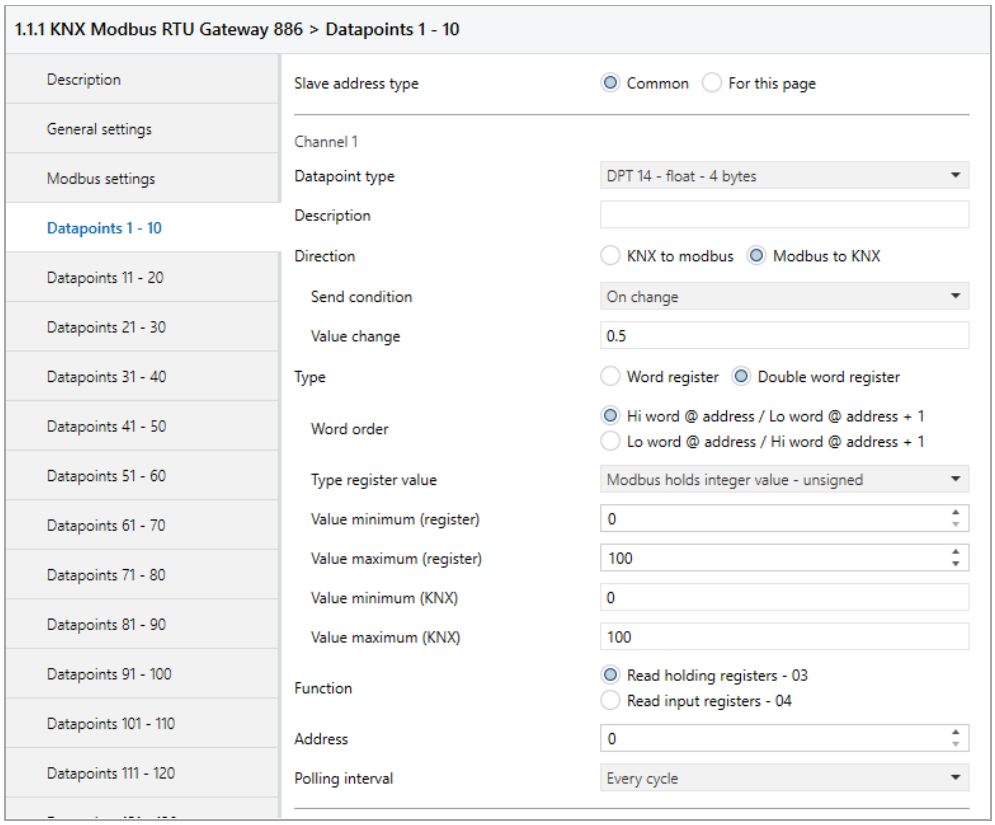
KNX value which corresponds to **Value minimum (register)**.

Value maximum (KNX)

KNX value, which corresponds to **Value maximum (register)**.

The conversion is always transferred to the entire register area.
***Value minimum/maximum (register)** defines no limits.*

5.15.2 Type – Double word register



Word order

This parameter defines the byte order in which the value of the group object (KNX) is distributed to the two word registers (Modbus). The following options are available:

- Hi word @ Address / Lo word @ Address + 1
- Lo word @ Address / Hi word @ Address + 1

Type Register value

This parameter defines how the float value is mapped to Modbus. The following options are available:

- Modbus contains integer value – unsigned
- Modbus contains integer value – 2th complement
- Modbus contains float value (IEEE)

Value minimum (register)

Register value which corresponds to **Value minimum (KNX)**.

Value maximum (register)

Register value which corresponds to **Value maximum (KNX)**.

Value minimum (KNX)

KNX value which corresponds to **Value minimum (register)**.

Value maximum (KNX)

KNX value, which corresponds to **Value maximum (register)**.

The conversion is always transferred to the entire register area.
***Value minimum/maximum (register)** defines no limits.*

Function *(for "KNX to Modbus" and "Double word register")*

This parameter defines the transmission mode of the double word register. The following options are available:

- Write multi holding registers – 16
  Both word registers in one request
- Write single holding register – 06
  For each word register one request

As "Modbus master", with "Modbus to KNX" and "Double word register", multi read requests should be activated in order to read both word registers in one request.

Address *(for "Double word register")*

Double word registers use the register address specified here as well as this register address + 1.

5.16 General information

5.16.1 Scaling

The scaling factor can be defined with the respective minimum/maximum values.

Example:

Value minimum (register) = 0
Value maximum (register) = 100
Value minimum (KNX) = 0
Value maximum (KNX) = 10

The result is scaling * 10 of the KNX value:
Value KNX = 10.5 → Value Register = 105

The conversion is always transferred to the entire register area.
***Value minimum/maximum (register)** defines no limits.*

5.16.2 2th complement

The 2th complement is used in Modbus registers to represent negative values. Thus, for example, a range of -32768...32767 can be displayed on a word register.

5.16.3 Modbus communication

If the KNX Gateway (master mode) does not receive a response from the slave within 1 second, the request is repeated twice. If these are not successful, all channels of this parameter page are skipped.

If the slave takes longer than 1 second to process the data, it can send an acknowledge telegram which restarts the time interval at the master.

5.16.4 Modbus specification

In Modbus there are various types of register address specification.

Variant 1:

Register type	Access	Size	Address range
Coil	RW	1 bit	00001 – 09999
Discrete inputs	R	1 bit	10001 – 19999
Input register	R	2 bytes	30001 – 39999
Holding register	RW	2 bytes	40001 – 49999

The address range additionally defines the register type.
The address range is 1 based.

Example – Coil:

00005 is the fifth coil.

Register address = 1 based
Function = Read coils – 01
Function = Write single coil – 05
Function = Write multi coils – 15
Address = 5

Example – Discrete input:

10001 is the first discrete input.

Register address = 1 based
Function = Read discrete inputs – 02
Address = 1

Example – Input register:

30002 is the second input register.

Register address = 1 based
Function = Read input register – 04
Address = 2

Example – Holding register:

40004 is the fourth holding register.

Register address = 1 based
Function = Read holding register – 03
Function = Write single holding register – 06
Function = Write multi holding register – 16
Address = 4

Variant 2:

Register type	Access	Size	Address range
Coil	RW	1 bit	0x0001 – 0x9999
Discrete inputs	R	1 bit	1x0001 – 1x9999
Input register	R	2 bytes	3x0001 – 3x9999
Holding register	RW	2 bytes	4x0001 – 4x9999

The address range additionally defines the register type.
The address range is 1 based.

Example – Coil:

0x0005 is the fifth coil.

Register address = 1 based
Function = Read coils – 01
Function = Write single coil – 05
Function = Write multi coils – 15
Address = 5

Example – Discrete input:

1x0001 is the first discrete input.

Register address = 1 based
Function = Read discrete inputs – 02
Address = 1

Example – Input register:

3x0002 is the second input register.

Register address = 1 based
Function = Read input register – 04
Address = 2

Example – Holding register:

4x0004 is the fourth holding register.

Register address = 1 based
Function = Read holding register – 03
Function = Write single holding register – 06
Function = Write multi holding register – 16
Address = 4

Variant 3:

Register type	Access	Size	Address range
Coil	RW	1 bit	0 – 65535
Discrete inputs	R	1 bit	0 – 65535
Input register	R	2 bytes	0 – 65535
Holding register	RW	2 bytes	0 – 65535

The address range defines the address that is actually sent.
The address range is 0 based.

Example – Coil:

5 is the sixth coil.

Register address = 0 based
Function = Read coils – 01
Function = Write single coil – 05
Function = Write multi coils – 15
Address = 5

Example – Discrete input:

0 is the first discrete input.

Register address = 0 based
Function = Read discrete inputs – 02
Address = 0

Example – Input register:

2 is the third input register.

Register address = 0 based
Function = Read input register – 04
Address = 2

Example – Holding register:

4 is the fifth holding register.

Register address = 0 based
Function = Read holding register – 03
Function = Write single holding register – 06
Function = Write multi holding register – 16
Address = 4

Variant 4:

Register type	Access	Size	Address range
Coil	RW	1 bit	1 – 65535
Discrete inputs	R	1 bit	1 – 65535
Input register	R	2 bytes	1 – 65535
Holding register	RW	2 bytes	1 – 65535

The address range is 1 based.

Example – Coil:

5 is the fifth coil.

Register address = 1 based
Function = Read coils – 01
Function = Write single coil – 05
Function = Write multi coils – 15
Address = 5

Example – Discrete input:

1 is the first discrete input.

Register address = 1 based
Function = Read discrete inputs – 02
Address = 1

Example – Input register:

2 is the second input register.

Register address = 1 based
Function = Read input register – 04
Address = 2

Example – Holding register:

4 is the fourth holding register.

Register address = 1 based
Function = Read holding register – 03
Function = Write single holding register – 06
Function = Write multi holding register – 16
Address = 4

WARNING

- The device must be mounted and commissioned by an authorized electrician.
- The prevailing safety rules must be heeded.
- The device must not be opened.
- For planning and construction of electric installations, the relevant guidelines, regulations and standards of the respective country are to be considered.

Product database for ETS 5/6
www.weinzierl.de/en/products/886/ets6

Data sheet
www.weinzierl.de/en/products/886/datasheet

CE Declaration
www.weinzierl.de/en/products/886/ce-declaration

WEINZIERL ENGINEERING GmbH
Achatz 3-4
84508 Burgkirchen an der Alz
GERMANY

Tel.: +49 8677 / 916 36 – 0
E-Mail: info@weinzierl.de
Web: www.weinzierl.de

2023-03-29