ENGLISH

# MODBUS RTU CLIENT TO KNX TP GATEWAY

## Application's program version: 1.1

USER MANUAL
Version 1.0.3
Publication date 2024-07-18

14 rue Edouard Petit
42000 St-Etienne
Tel : +33 477 92 03 56
Email : contact@rg2i.com
www.rg2i.com

Copyright © 2024 Intesis

***Important User Information***

**Disclaimer**

The information in this document is for informational purposes only. Please inform HMS Networks of any inaccuracies or omissions found in this document. HMS Networks disclaims any responsibility or liability for any errors that may appear in this document.

HMS Networks reserves the right to modify its products in line with its policy of continuous product development. The information in this document shall therefore not be construed as a commitment on the part of HMS Networks and is subject to change without notice. HMS Networks makes no commitment to update or keep current the information in this document.

The data, examples and illustrations found in this document are included for illustrative purposes and are only intended to help improve understanding of the functionality and handling of the product. In view of the wide range of possible applications of the product, and because of the many variables and requirements associated with any particular implementation, HMS Networks cannot assume responsibility or liability for actual use based on the data, examples or illustrations included in this document nor for any damages incurred during installation of the product. Those responsible for the use of the product must acquire sufficient knowledge in order to ensure that the product is used correctly in their specific application and that the application meets all performance and safety requirements including any applicable laws, regulations, codes and standards. Further, HMS Networks will under no circumstances assume liability or responsibility for any problems that may arise as a result from the use of undocumented features or functional side effects found outside the documented scope of the product. The effects caused by any direct or indirect use of such aspects of the product are undefined and may include e.g. compatibility issues and stability issues.

# Table of Contents

**1. Description and Order Codes ..... 1**

**2. General Information ..... 2**

2.1. Intended Use of the User Manual ..... 2
2.2. General Safety Information ..... 2
2.3. Admonition Messages and Symbols ..... 3

**3. Overview ..... 4**

3.1. Inside the Package ..... 4
3.2. Main Features ..... 4
3.3. General Introduction ..... 5
3.4. Gateway Capacity ..... 5

**4. Hardware ..... 6**

4.1. Mounting ..... 6
4.2. Connection Procedure and DIP Switch Setting ..... 7
4.3. LED Indicators ..... 8
4.4. Push Button ..... 9
4.5. Technical Specifications ..... 10
4.6. Dimensions ..... 11

**5. Communication Interfaces ..... 12**

5.1. KNX Interface ..... 12
5.1.1. KNX Communication Objects ..... 12
5.1.2. KNX Points Definition ..... 13
5.2. Modbus Interface ..... 13
5.2.1. Modbus Points Definition ..... 14

**6. Configuration ..... 16**

6.1. Prerequisites ..... 16
6.2. Configuration Process with the ETS Tool ..... 16
6.3. General Configuration Tab ..... 17
6.4. Device Tab ..... 19
6.5. Intesis Modbus Device Manager DCA ..... 20
6.6. Sending the Configuration to the Gateway ..... 22

# 1. Description and Order Codes

**INKNXMBM1000200 Protocol Translator Gateway.**

Modbus RTU Client to KNX TP Gateway.

| ORDER CODE | LEGACY ORDER CODE |
|---|---|
| INKNXMBM1000200 | INKNXMBM1000100 |

# 2. General Information

## 2.1. Intended Use of the User Manual

This manual contains the main features of this Intesis gateway and the instructions for its appropriate installation, configuration, and operation.

The contents of this manual should be brought to the attention of any person who installs, configures, or operates this gateway or any associated equipment.

Keep this manual for future reference during the installation, configuration, and operation.

## 2.2. General Safety Information

**IMPORTANT**
Follow these instructions carefully. Improper work may seriously harm your health and damage the gateway and/or any other equipment connected to it.

Only technical personnel, following these instructions and the country legislation for installing electrical equipment, can install and manipulate this gateway.

Install this gateway indoors, in a restricted access location, avoiding exposure to direct solar radiation, water, high relative humidity, or dust.

Preferably, mount this gateway on a DIN rail inside a grounded metallic cabinet, following the instructions in this manual.

If mounting on a wall, firmly fix this gateway on a non-vibrating surface, following the instructions in this manual.

All wires (for communication and power supply, if needed) must only be connected to networks with indoor wiring. All communication ports are considered for indoor use and must only be connected to SELV circuits.

Disconnect all systems from power before manipulating and connecting them to the gateway.

Respect the expected polarity of power and communication cables when connecting them to the gateway.

This Intesis gateway is designed for installation in an enclosure. When the device is mounted outside an enclosure, precautions should be taken to avoid electrostatic discharges to the unit in environments with static levels above 4 kV. When working in an enclosure (e.g., making adjustments, setting switches, etc.), typical anti-static precautions should be observed before touching the unit.

Binary inputs, if present, are potential-free contact. Do not connect any voltage.

## 2.3. Admonition Messages and Symbols

**DANGER**
Instructions that must be followed to avoid an imminently hazardous situation that, if not avoided, will result in death or severe injury.

**WARNING**
Instructions that must be followed to avoid a potentially hazardous situation that, if not avoided, could result in death or severe injury.

**CAUTION**
Instruction that must be followed to avoid a potentially hazardous situation that, if not avoided, could result in minor or moderate injury.

**IMPORTANT**
Instruction that must be followed to avoid a risk of reduced functionality and/or damage to the equipment or to avoid a network security risk.

**NOTE**
Additional information which may facilitate installation and/or operation.

**TIP**
Helpful advice and suggestions.

**NOTICE**
Remarkable Information.

# 3. Overview

## 3.1. Inside the Package

Items included:

- Intesis INKNXMBM1000200 protocol translator gateway
- KSN label
- Installation guide

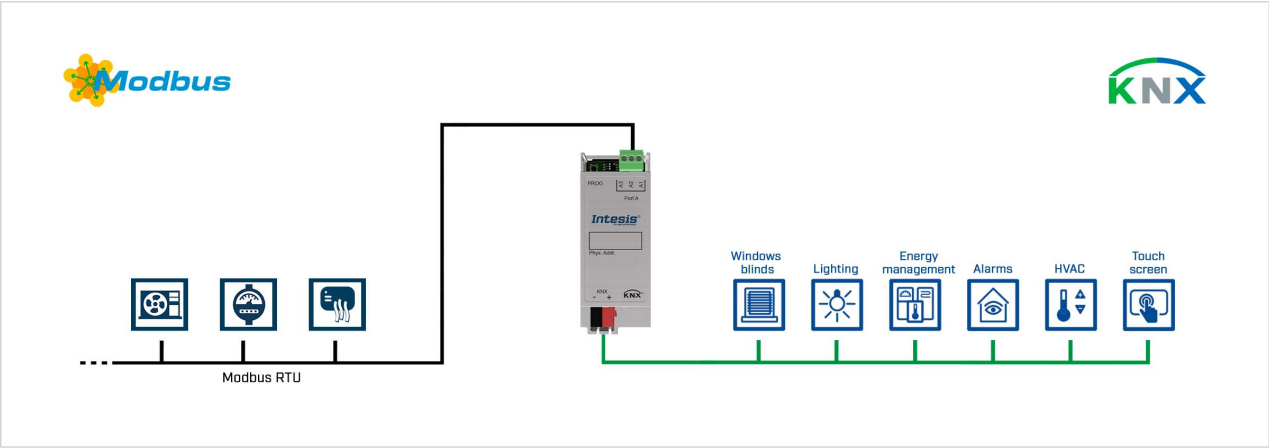
## 3.2. Main Features

- Accredited with the main certifications for electronic equipment, including the official KNX certification. To know more, check the Technical Documentation section on the product website.
- Uses KNX standard datapoints (DPTs). This guarantees a smooth integration in any KNX device or system.
- Programmable with the KNX official tool ETS.
- Flexible configuration of devices and signals.
- Export and import device templates through a DCA for quick integration.
- Every signal allows arithmetic and logical operations such as multiplication, division, equal to, greater than, etc., to adapt them to the Modbus registers.
- Reduced dimensions (HxWxD): 92x36x32 mm.
- Up to 100 Modbus registers and 205 KNX communication objects.
- Up to 100 devices (when installing more than 32 devices, you will need a repeater. This is an inherent limitation of the Modbus protocol).
- Two visible LEDs indicate the local operating status.

## 3.3. General Introduction

This document describes how to integrate Modbus RTU server devices into a KNX TP installation using the Intesis Modbus RTU client to KNX TP gateway.

*Figure 1. Integration of Modbus devices into a KNX TP installation*




This gateway allows a KNX TP system to access the signals and resources of Modbus server devices. To do so, the gateway is continuously polling the Modbus server devices and associates the Modbus signals to the KNX communication objects; this way, the KNX TP system perceives all the Modbus devices as if they were one more KNX device of the system and vice versa: the Modbus devices perceive the KNX system as if it was one more Modbus server device of the network.

## 3.4. Gateway Capacity

| Element | 100 version |
|---|---|
| **Modbus interface** | |
| **Maximum number of Modbus server devices supported** | 100 (see note below) |
| **Type of Modbus server devices supported** | Those supporting Modbus protocol communication over RTU (EIA-485) |
| **Modbus addresses range** | 1 to 255 |
| **Maximum number of Modbus signals** | 100 |
| **KNX interface** | |
| **Maximum number of communication objects** | 205 signal objects:<br>100 status objects<br>100 control objects<br>4 error status objects<br>1 optional object for heartbeat |
| **Maximum number of KNX group addresses** | 700 |
| **Maximum number of possible KNX associations** | 700 |

**NOTE**
When installing more than 32 devices, you will need a repeater. This is an inherent limitation of the Modbus protocol.

# 4. Hardware

## 4.1. Mounting

**IMPORTANT**
Before mounting, please ensure that the chosen installation place preserves the gateway from direct solar radiation, water, high relative humidity, or dust.

**IMPORTANT**
Ensure the gateway has sufficient clearances for all connections when mounted.

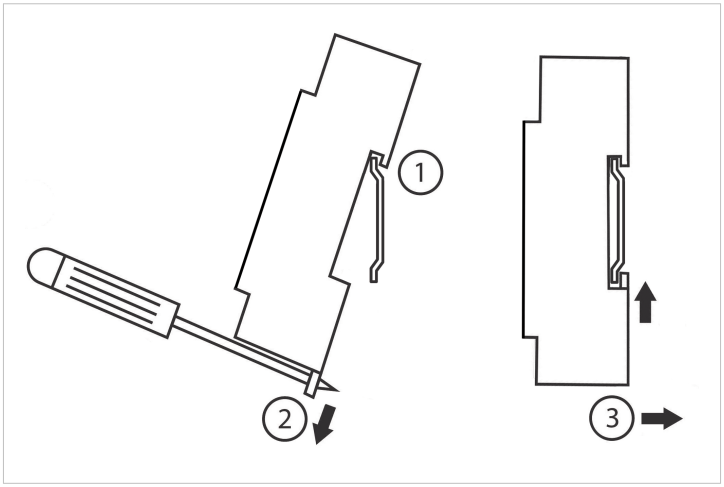
**NOTE**
Mount the gateway over a DIN rail, preferably inside a grounded metallic industrial cabinet.

**NOTE**
This gateway's profile (depth) may be thinner than other devices on the DIN rail. Check the local regulations since installing a blanking plate may be necessary to cover the gateway and avoid any manipulation through the gap.

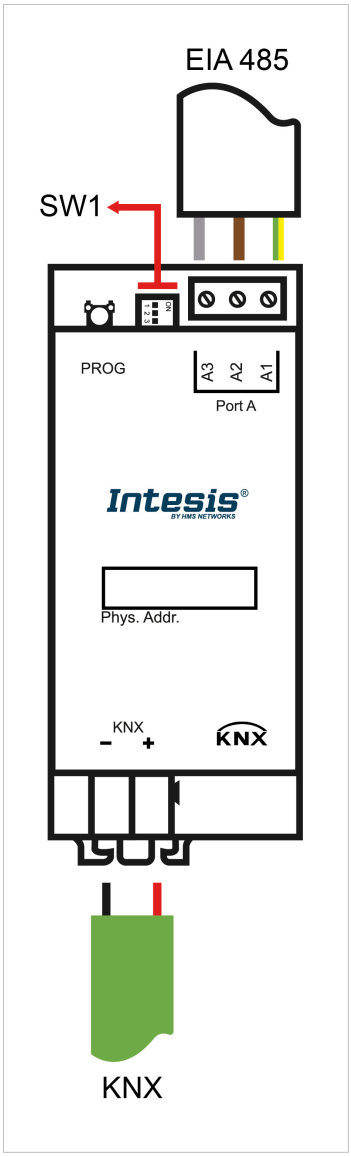
***DIN rail mounting***

1. Insert the gateway in the upper edge of the DIN rail.
2. Press the low side of the gateway gently to lock it in the DIN rail.
3. Make sure the gateway is firmly fixed.

**NOTE**
For some DIN rails, to complete step 2, you may need a small screwdriver or similar to pull the bottom clip down.



## 4.2. Connection Procedure and DIP Switch Setting




**CAUTION**
Before connecting the gateway, disconnect the KNX bus from its power supply.

***SW1***

Use this DIP switch to configure the EIA-485 port:

- Position 1:
  - ON: 120 Ω termination active
  - OFF: 120 Ω termination inactive (Default)

**IMPORTANT**
If you install the gateway at one end of the EIA-485 bus, activate the terminal resistor functionality by setting switch 1 to the ON position.

**CAUTION**
Make sure there are at most two end-of-line resistors on the EIA-485 bus. Having more than that could cause electrical issues.

- Position 2 and 3:
  - ON: Polarization active (Default)
  - OFF: Polarization inactive

**IMPORTANT**
The bus must be polarized only at one point of the line, preferably on the Client side. To disable the gateway's line polarization, set switches 2 and 3 to the OFF position.

1. Connect the EIA-485 bus to connectors A3 (A+), A2 (B-), and A1 (SNGD) of the gateway's Port A.
2. Connect the KNX bus to the connectors + and - of the gateway's KNX port. Respect the polarity and use a KNX standard cable.
3. Reconnect the KNX bus to its power supply.
4. Connect all the devices of your installation (the EIA-485 bus devices) to their power supplies.

**NOTE**
This gateway gets the power from the KNX bus itself, so it does not need an external power supply. If there is a power cut in the bus, the gateway will turn off.

Once the bus is powered again:

- The gateway turns on, keeping its parameter's configuration.
- It polls the bus and updates all status objects for the Modbus devices connected to it.

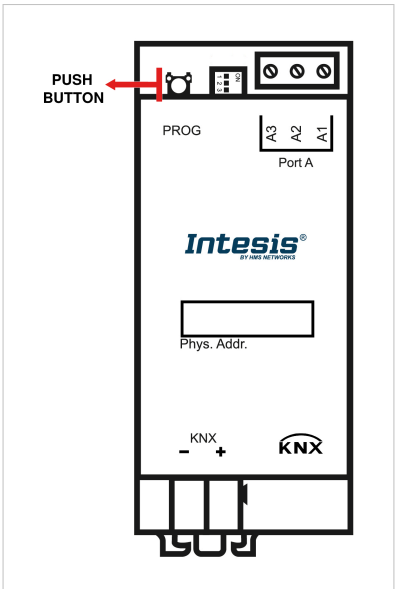
## 4.3. LED Indicators

There are two LEDs below the DIP switch.

*Table 1. LEDs behavior*

| LED ID and color | Pattern | Description |
|---|---|---|
| **KNX programming mode** | | |
| RED | Off | Programming mode disabled |
| | Steady On | Programming mode enabled |
| | Flashing | To locate the device |
| **Power/Modbus activity** | | |
| YELLOW | Off | No power |
| | Steady On | Device powered but without communication |
| | Slow blinking | Invalid answer or no answer received from a server device |
| | Fast blinking | Correct package received from a configured server device |

# 4.4. Push Button




This button activates the gateway's programming mode, which is needed for some ETS download variants:

- **Download All**: It downloads the Individual Address, the Application Program, the Parameters, and the Group addresses.
- **Download Individual Address**.

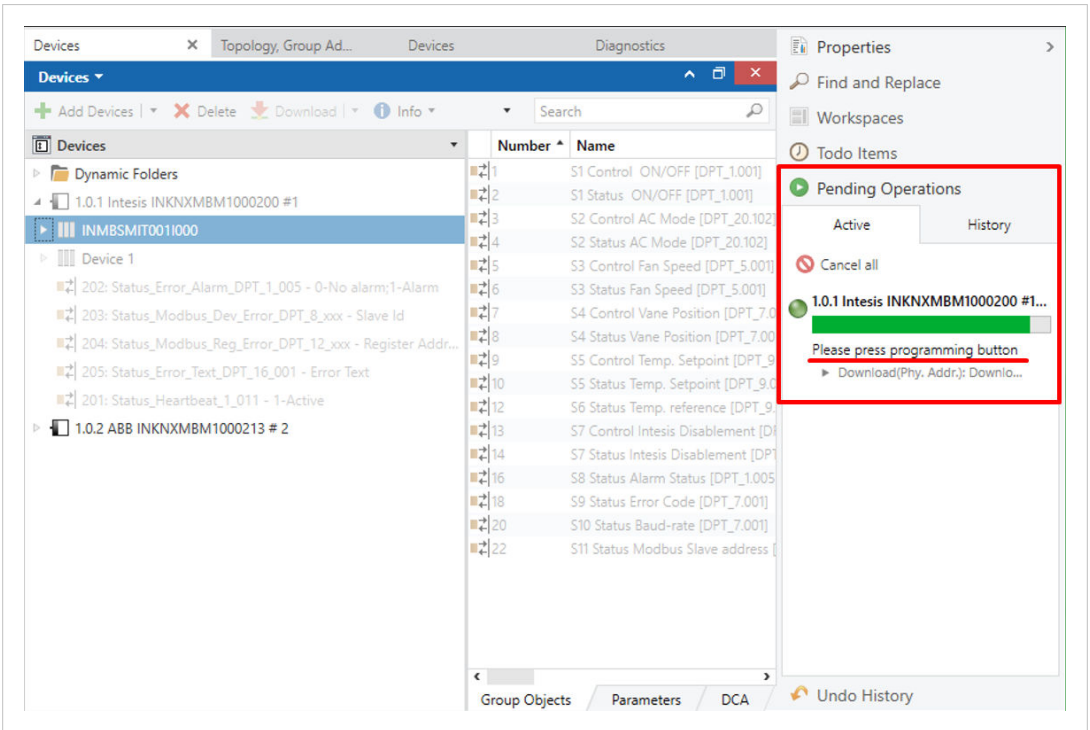
The button is also needed to use the **Unload Application and Address function**.

You can read a broader explanation in these KNX articles:

Download functions: https://support.knx.org/hc/en-us/articles/360007474340-Download-functions

Unload device: https://support.knx.org/hc/en-us/articles/4402998506386-Unload-Device

ETS software will tell you when to push the button:



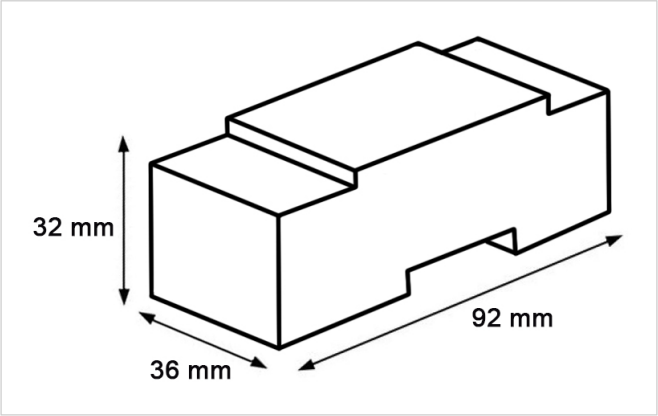
# 4.5. Technical Specifications

| | |
|---|---|
| **Housing** | • Front side: PC (UL 94 V-0)<br>• Back side: PPO (UL 94 V-0)<br>• Net dimensions (HxWxD):<br>Millimeters: 92 x 36 x 32 mm<br>Inches: 3.62 x 1.41 x 1.25" |
| | Colors:<br>• Front side: Light grey. RAL 7035<br>• Rear side: Black |
| **Mounting** | DIN rail (35 mm/1.4"; two modules) |
| **Power** | Supplied through KNX bus |
| **KNX Port** | 1 x KNX TP-1 Plug-in terminal block (2 poles)<br>• 1500 VDC isolation from other ports<br>• KNX power consumption: 11 .. 22 mA<br>• Voltage rating: 21 .. 32 VDC |
| **Port A** | 1 x Green serial pluggable terminal block (3 poles)<br>• A1: SNGD (Reference ground or shield)<br>• A2 (B-)<br>• A3 (A+)<br>2500 VDC isolation from other ports |
| **Push button** | Activates the gateway's programming mode |
| **Operational temperature** | 0°C to +60°C / 32 to 140°F |
| **Operational humidity** | 5 to 95%, no condensation |
| **SW1** | 1 x DIP switch (three positions) for serial EIA-485 configuration:<br>• Position 1:<br>– ON: 120 Ω termination active<br>– OFF: 120 Ω termination inactive (Default)<br>• Position 2 and 3:<br>– ON: Polarization active (Default)<br>– OFF: Polarization inactive |
| **LED indicators** | 2 x Onboard LED indicators<br>• 1 x Power/Port A activity<br>• 1 x KNX programming mode |

## 4.6. Dimensions

- **Net dimensions (HxWxD)**
  Millimeters: 92 x 36 x 32 mm
  Inches: 3.62 x 1.41 x 1.25"

**IMPORTANT**
Leave enough clear space to wire the gateway easily and for the subsequent manipulation of elements.



# 5. Communication Interfaces

## 5.1. KNX Interface

This Intesis gateway continuously checks the status of the KNX bus. The behavior of each signal point is defined by the flags you set in the ETS configuration tool. This means, for instance, that when the communication is restored after a blackout, the gateway will retransmit the status of all the KNX groups marked as Transmit (T) and the update groups marked as Update (U).

**NOTE**
By default, the gateway's physical address is **15.15.255**. If you change this value, the gateway will reboot, keeping all its configuration parameters and proceeding to poll the bus to update all status objects for the Modbus devices connected to it.

**IMPORTANT**
If having communication issues in large installations, check if a line coupler is filtering the telegrams from/to the Intesis gateway.

### 5.1.1. KNX Communication Objects

| Object number | Name | Length | Datapoint type | | Flags | | | | Function |
|---|---|---|---|---|---|---|---|---|---|
| | | | DPT_Name | DPT_ID | R | W | T | U | |
| 201 | Status_Heartbeat | 1 bit | State | 1.011 | R | | T | | 1: Active |
| 202 | Status_Error_Alarm | 1 bit | Alarm | 1.005 | R | | T | | 0: No alarm; 1: Alarm |
| 203 | Status_Modbus_Dev_Error | 2 bytes | 2 byte signed value | 8.xxx | R | | T | | Address of the last Modbus server device that currently has an error |
| 204 | Status_Modbus_Reg_Error | 4 bytes | 4 byte unsigned value | 12.xxx | R | | T | | Register address of the last Modbus server device that currently has an error |
| 205 | Status_Error_Text | 14 bytes | Character string (ISO 9959-1) | 16.001 | R | | T | | MBM Overload: an electric overload is detected from the power supply.<br>Error Text: string message about the error or the alarm in the format **ss:ff:aaaa:ee**<br>• ss: server id (00 .. 3F)<br>• ff: function code (00 .. 10)<br>• aaaa: address (0000 .. FFFF)<br>• ee: exception code (see tables below) |

*Table 2. Modbus exception codes*

| Code | Name | Meaning |
|---|---|---|
| 01 | Illegal function | The function code received in the query is not an allowable action for the server. Example: The function code only applies to newer devices and was not implemented in the selected device. |
| 02 | Illegal data address | The data address received in the query is not an allowable address for the server (the combination of the reference number and transfer length is invalid). Example: The submitted request attempts to operate on registers 96, 97, 98, 99, and 100, and there is no register with address 100. |
| 03 | Illegal data value | A value contained in the query data field is not an allowable value for the server. This indicates a fault in the structure of the remainder of a complex request, such as that the implied length is incorrect. |
| 04 | Server device failure | An unrecoverable error occurred while the server attempted to perform the requested action. |
| 05 | Acknowledge | The server is processing the request, but it will require a long time to do so. This response is returned to prevent a timeout error on the client side. |
| 06 | Server device busy | The server is processing a long–duration program command. The client should retransmit the message later when the server is free. |
| 07 | Memory parity error | The server detected a parity error in the memory when reading the record file. The client may require service on the server device to retry the request. |
| 0A | Gateway path unavailable | The gateway could not allocate an internal communication path from the input to the output port for processing the request. Example: The gateway is misconfigured or overloaded. |
| 0B | Gateway target device failed to respond | The target device does not respond. Example: The target device is not present on the network. |

*Table 3. Gateway exception codes*

| Code | Name | Meaning |
|---|---|---|
| ER0 | Unknown error | The error type cannot be determined. |
| ER1 | CRC error | The cyclic redundancy check has detected an error in the response structure. |
| ER2 | Reception length error | The response length is not correct. |
| ER3 | Reception time out | No response after the timeout. |

### 5.1.2. KNX Points Definition

The internal datapoints have the following KNX properties:

| Property | Description |
|---|---|
| **Active** | If enabled, this signal will be active; if not, the behavior will be as if the point was not defined. This allows you to deactivate points instead of deleting them, so that you can save those points for future use. |
| **#** | This is the number of the signal. |
| **Object Name** | Descriptive information about the signal. You can edit these names. |
| **DPT** | This is the KNX data type used to code the signal's value, which depends on the type of signal associated with the Modbus side. |

## 5.2. Modbus Interface

This Intesis gateway acts as a client device in the Modbus RTU network. Other Modbus devices connected to the same line and communicating with the Intesis gateway must always be server devices.

The Modbus protocol defines different types of function codes to read/write the registers of Modbus devices. It also defines different data formats to encode values.

In the Modbus interface, you can configure the data encoding used for 16-bit registers (big-endian or little-endian). This is the byte order for data encoding (MSB .. LSB or LSB .. MSB)[1]. Although the Modbus protocol specifies this data encoding as big-endian, it varies depending on the manufacturer and/or the type of server.

[1]MBS: Most Significant Bit
LSB: Less Significant Bit

### 5.2.1. Modbus Points Definition

Each signal point defined in the Intesis gateway has the following associated Modbus features:

| Feature | Description |
|---|---|
| **# Server** | Modbus server address where the point belongs to. |
| **Function code**<br>**Read function**<br>**Write function** | One of the following Modbus data coding formats can be used:<br>• 1 - Read Coils<br>• 2 - Read Discrete Inputs<br>• 3 - Read Holding Registers<br>• 4 - Read Input Registers<br>• 5 - Write Single Coil<br>• 6 - Write Single Register<br>• 15 - Write Multiple Coils<br>• 16 - Write Multiple Registers |
| **Data coding format** | One of the following Modbus data coding formats can be used:<br>• 16/32/48/64 bits unsigned<br>• 16/32/48/64 bits signed (one's complement - C1)<br>• 16/32/48/64 bits signed (two's complement - C2)<br>• 16/32/48/64 bits Float<br>• 16 bits Bitfields |
| **Byte order** | • Big-endian<br>• Little-endian<br>• World inverted big-endian<br>• World inverted little-endian |
| **Register address** | The Modbus register address on the server device for the signal point. |
| **Bit** | You can decide where the signal starts in the register (optional function). The gateway allows bit decoding from generic 16 bits input/holding Modbus registers.<br>Bit coding into 16-bit input/holding Modbus registers is used to encode digital values into the registers, being usually accessible using Modbus function codes 3 and 4 (read holding/input registers). |
| **# Bits** | Number of bits used by the signal. |
| **Deadband** | Determine the change of value (COV) in a signal value. |
| **Operation** | The following options are available:<br>• No operation<br>• Arithmetic operations: Operations that can be applied to Modbus values before sending them to the KNX control system. The inverse operation is also applied to KNX values before sending them to the Modbus installation.<br>– Multiply by ( x )<br>– Divide by ( / )<br>• Logical operations: Operations that can be applied to read-only or write-only signals. These operations return “1” when the result matches the condition and “0” when it doesn’t.<br>– Is equal ( == )<br>– Is different ( != )<br>– Is less than ( < )<br>– Is greater than ( > ) |
| **Operation Value** | Value for the operation formula. |
| **Operation Definition** | It shows the formula applied to the signal and the direction (from Modbus to KNX or vice versa). |

*Figure 2. Points definition window. See the different operation options in the last three columns*

| | # | Object Name | DPT | Server Address | Read Function | Write Function | Data Length | Format | Byte Order | Register Address | Bit | # Bits | Deadband | Operation | Operation Value | Operation Definition |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ✓ | 1 | Only Write x 10 | 7.001: pulses | 1 | - | 6: Write Single Register | 16 | 0: Unsigned | 0: Big Endian | 1 | - | - | 0 | Multiply by ( x ) | 10 | mb = (knx * 10) |
| ✓ | 2 | Only Read x 10 | 7.001: pulses | 1 | 3: Read Holding Registers | - | 16 | 0: Unsigned | 0: Big Endian | 2 | - | - | 0 | Multiply by ( x ) | 10 | knx = (mb * 10) |
| ✓ | 3 | Read,Write x 10 | 7.001: pulses | 1 | 3: Read Holding Registers | 6: Write Single Register | 16 | 0: Unsigned | 0: Big Endian | 3 | - | - | 0 | Multiply by ( x ) | 10 | knx = (mb * 10)<br>mb = (knx / 10) |
| ✓ | 4 | Only Write / 10 | 7.001: pulses | 1 | - | 6: Write Single Register | 16 | 0: Unsigned | 0: Big Endian | 4 | - | - | 0 | Divide by ( / ) | 10 | mb = (knx / 10) |
| ✓ | 5 | Only Read / 10 | 7.001: pulses | 1 | 3: Read Holding Registers | - | 16 | 0: Unsigned | 0: Big Endian | 5 | - | - | 0 | Divide by ( / ) | 10 | knx = (mb / 10) |
| ✓ | 6 | Read,Write / 10 | 7.001: pulses | 1 | 3: Read Holding Registers | 6: Write Single Register | 16 | 0: Unsigned | 0: Big Endian | 6 | - | - | 0 | Divide by ( / ) | 10 | knx = (mb / 10)<br>mb = (knx * 10) |
| ✓ | 7 | Write = 10 | 7.001: pulses | 1 | - | 6: Write Single Register | 16 | 0: Unsigned | 0: Big Endian | 7 | - | - | 0 | Is equal ( = ) | 10 | mb = (knx == 10) |
| ✓ | 8 | Read = 10 | 7.001: pulses | 1 | 3: Read Holding Registers | - | 16 | 0: Unsigned | 0: Big Endian | 8 | - | - | 0 | Is equal ( = ) | 10 | knx = (mb == 10) |
| ✓ | 9 | Write != 10 | 7.001: pulses | 1 | - | 6: Write Single Register | 16 | 0: Unsigned | 0: Big Endian | 9 | - | - | 0 | Is different ( ≠ ) | 10 | mb = (knx != 10) |
| ✓ | 10 | Read != 10 | 7.001: pulses | 1 | 3: Read Holding Registers | - | 16 | 0: Unsigned | 0: Big Endian | 10 | - | - | 0 | Is different ( ≠ ) | 10 | knx = (mb != 10) |
| ✓ | 11 | Write < 10 | 7.001: pulses | 1 | - | 6: Write Single Register | 16 | 0: Unsigned | 0: Big Endian | 11 | - | - | 0 | Is less than ( < ) | 10 | mb = (knx < 10) |
| ✓ | 12 | Read < 10 | 7.001: pulses | 1 | 3: Read Holding Registers | - | 16 | 0: Unsigned | 0: Big Endian | 12 | - | - | 0 | Is less than ( < ) | 10 | knx = (mb < 10) |
| ✓ | 13 | Write > 10 | 7.001: pulses | 1 | - | 6: Write Single Register | 16 | 0: Unsigned | 0: Big Endian | 13 | - | - | 0 | Is greater than ( > ) | 10 | mb = (knx > 10) |
| ✓ | 14 | Read > 10 | 7.001: pulses | 1 | 3: Read Holding Registers | - | 16 | 0: Unsigned | 0: Big Endian | 14 | - | - | 0 | Is greater than ( > ) | 10 | knx = (mb > 10) |

# 6. Configuration

## 6.1. Prerequisites

For this integration, you will need:

- The items supplied by HMS Networks:
  - Intesis gateway
  - Gateway documentation:
    - Installation guide
    - User manual
  - ETS database for this gateway:
    - Find it under the Configuration files section on the HMS website.
- An ETS5 license (version 5.7.7 onwards)
- An operative KNX control system, properly connected to the KNX port of the Intesis gateway.
- The Modbus server device(s) or installation you are about to integrate, connected to the Port A (EIA-485) of the Intesis gateway.
- A Windows® computer to run the ETS configuration tool.

## 6.2. Configuration Process with the ETS Tool

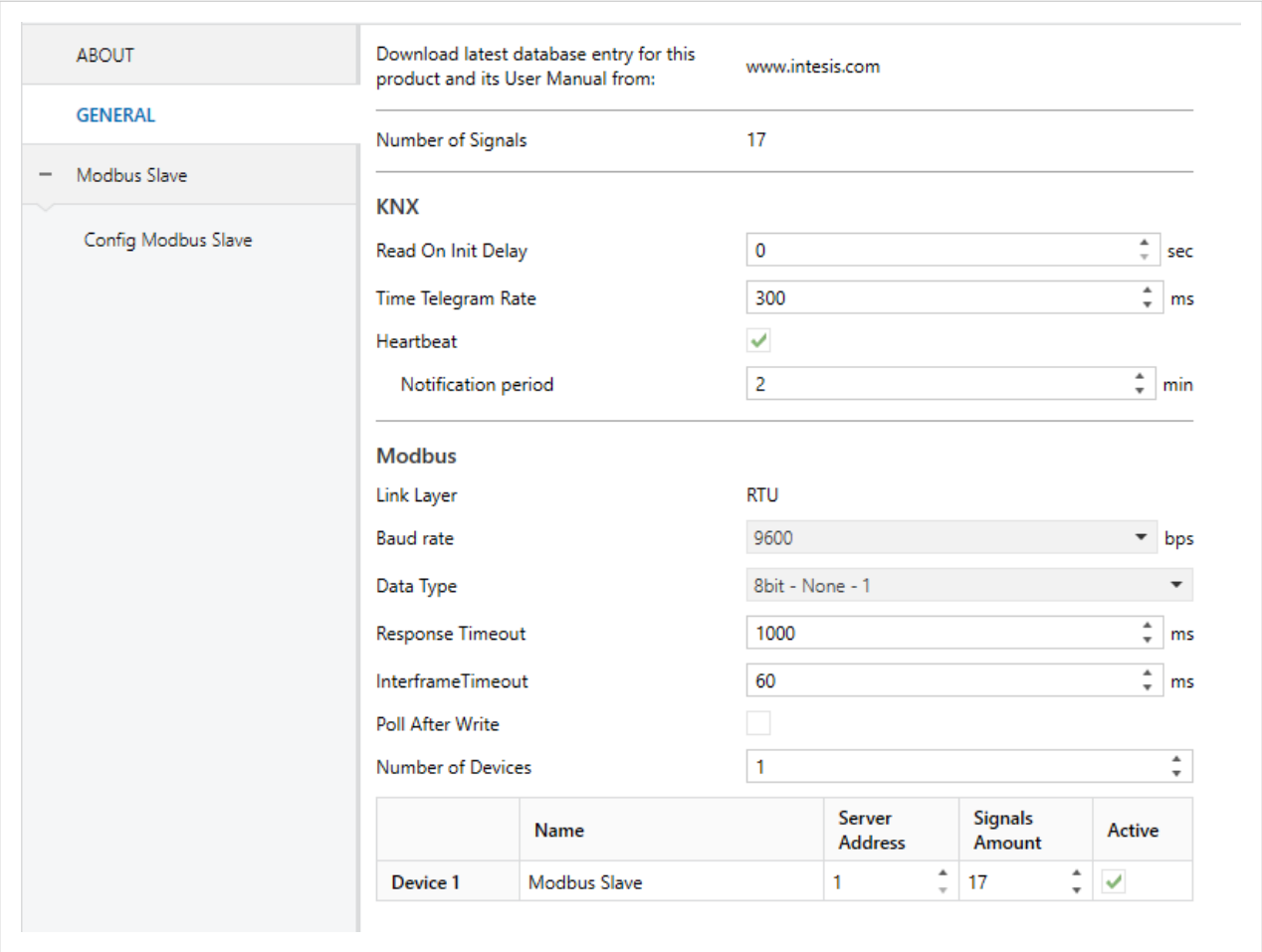
This Intesis gateway is officially certified by KNX. To set it up, use the standard KNX configuration tool ETS. The ETS database for this gateway can be found in the ETS catalog or downloaded from here.

**NOTE**
When importing the gateway to the ETS software for the first time, it will show 10 Modbus active signals, each configured as a Read and Write signal. You can configure these signals later and adapt them to your project needs.

## 6.3. General Configuration Tab

Access this tab to configure the Modbus connection parameters and the total number of signals to map.

*Figure 3. General configuration tab*




- **Read on Init Delay**: Determines the time in seconds to wait before all values are uploaded to the KNX network.

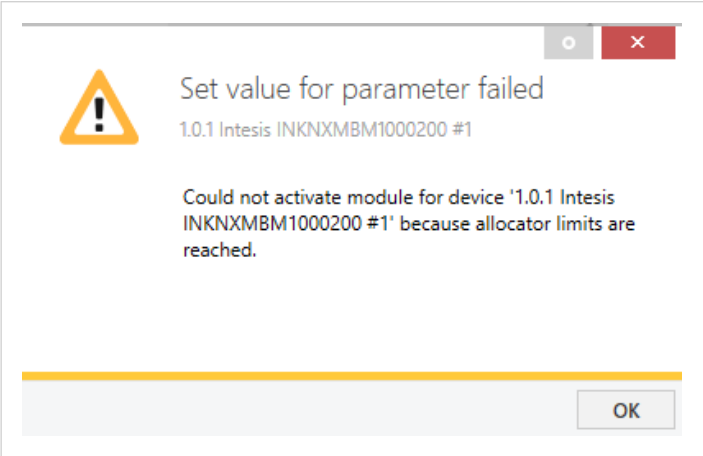
> **IMPORTANT**
> This setting applies only to objects with the **Read on Init** flag enabled.

- **Time Telegram Rate**: Determines the time in milliseconds between telegrams when some data is updated simultaneously.
- **Heartbeat**: Activating this function, the device will report that it is online.
- **Modbus connection parameters**:
  - **Baudrate**: Defines the communication speed for the RTU communication (from 1200 to 115200 bps.)
  - **Data Type**: Defines, in this order: the number of bits used for data (8bit), the parity of the communication (None/Odd/Even), and the number of stop bits (1 or 2).
  - **Response timeout**: Determines the time in seconds to wait for a reply to a sent request
  - **Interframe timeout**: Determines the minimum time between received frame and sent frame.
  - **Poll After Write**: Enables or disables a poll for a particular signal after sending a write command to it.
  - **Number of devices**: Determines the number of Modbus devices you are integrating.

**NOTE**
ETS creates a tab for each device.

– **Number of signals**: Defines the number of Modbus signals to integrate. Once a signal is defined, ETS automatically creates one or two KNX objects, depending on the selected function. When selecting both **write** and **read** functions, ETS creates two objects (one **status** object and one **command** object); but if you only select the function **write** or the function **read**, ETS creates one single object (a command object for the write function or a status object for the read function).

**IMPORTANT**
If you try to integrate more than 100 devices or signals, this message will popup:




The bottom side of the window shows the Modbus device table, where you can:

- See each device and its name.
- Change the server address.
- Change the number of signals.

**NOTE**
You can change this from the **Device Tab**, also.

- Activate or deactivate each device.

**IMPORTANT**
The signals of deactivated devices won't be transferred to the gateway.

## 6.4. Device Tab

On the top side of this window, you can see the device information:

- **Device Name**
- **Device Server Address**
- **Device Signals Amount**: Indicates the number of signals to integrate from the Modbus server device to the KNX system. You can add up to 100 signals.

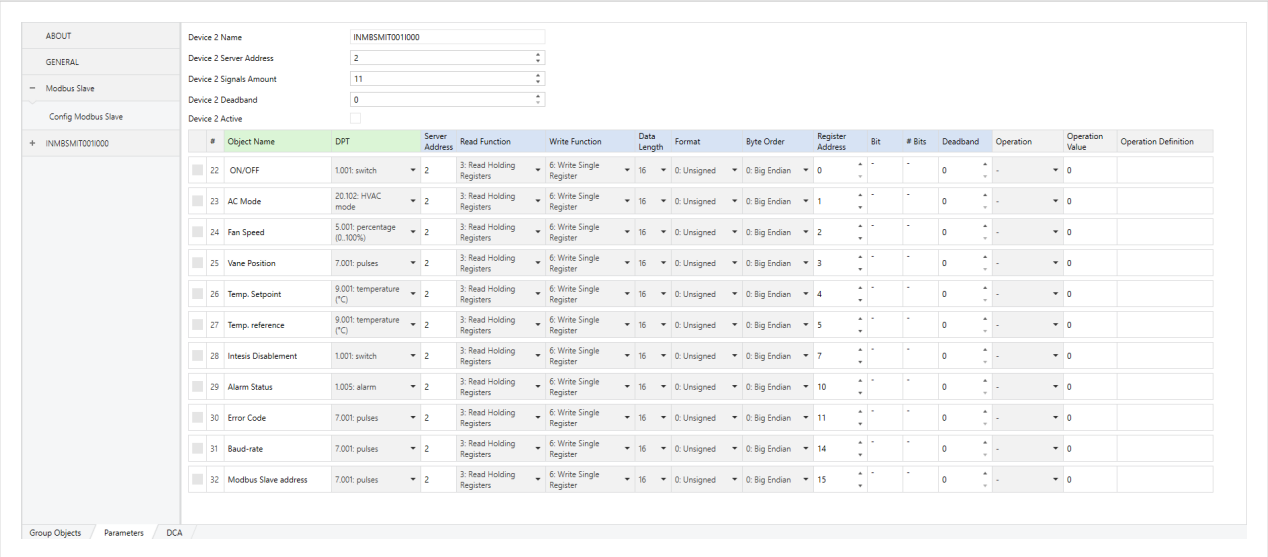
> **NOTE**
> By default, the list for the first device shows ten signals. For the rest of the devices, only one signal is shown. You can add more according to your needs.

- **Device Deadband**: Determines the minimum change of value (COV) for all signals to update it in the KNX system.

> **IMPORTANT**
> Modifying this general value will overwrite the specific Deadband values of each signal on the device configuration page.

Below this information, you can see the list of all available KNX objects, their corresponding Modbus registers, and other relevant parameters for every Modbus server device of the installation.

*Figure 4. Device tab*




| # | Object Name | DPT | Server Address | Read Function | Write Function | Data Length | Format | Byte Order | Register Address | Bit | # Bits | Deadband | Operation | Operation Value | Operation Definition |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 22 | ON/OFF | 1.001: switch | 2 | 3: Read Holding Registers | 6: Write Single Register | 16 | 0: Unsigned | 0: Big Endian | 0 | - | - | 0 | - | 0 | |
| 23 | AC Mode | 20.102: HVAC mode | 2 | 3: Read Holding Registers | 6: Write Single Register | 16 | 0: Unsigned | 0: Big Endian | 1 | - | - | 0 | - | 0 | |
| 24 | Fan Speed | 5.001: percentage (0..100%) | 2 | 3: Read Holding Registers | 6: Write Single Register | 16 | 0: Unsigned | 0: Big Endian | 2 | - | - | 0 | - | 0 | |
| 25 | Vane Position | 7.001: pulses | 2 | 3: Read Holding Registers | 6: Write Single Register | 16 | 0: Unsigned | 0: Big Endian | 3 | - | - | 0 | - | 0 | |
| 26 | Temp. Setpoint | 9.001: temperature (°C) | 2 | 3: Read Holding Registers | 6: Write Single Register | 16 | 0: Unsigned | 0: Big Endian | 4 | - | - | 0 | - | 0 | |
| 27 | Temp. reference | 9.001: temperature (°C) | 2 | 3: Read Holding Registers | 6: Write Single Register | 16 | 0: Unsigned | 0: Big Endian | 5 | - | - | 0 | - | 0 | |
| 28 | Intesis Disablement | 1.001: switch | 2 | 3: Read Holding Registers | 6: Write Single Register | 16 | 0: Unsigned | 0: Big Endian | 7 | - | - | 0 | - | 0 | |
| 29 | Alarm Status | 1.005: alarm | 2 | 3: Read Holding Registers | 6: Write Single Register | 16 | 0: Unsigned | 0: Big Endian | 10 | - | - | 0 | - | 0 | |
| 30 | Error Code | 7.001: pulses | 2 | 3: Read Holding Registers | 6: Write Single Register | 16 | 0: Unsigned | 0: Big Endian | 11 | - | - | 0 | - | 0 | |
| 31 | Baud-rate | 7.001: pulses | 2 | 3: Read Holding Registers | 6: Write Single Register | 16 | 0: Unsigned | 0: Big Endian | 14 | - | - | 0 | - | 0 | |
| 32 | Modbus Slave address | 7.001: pulses | 2 | 3: Read Holding Registers | 6: Write Single Register | 16 | 0: Unsigned | 0: Big Endian | 15 | - | - | 0 | - | 0 | |

The first four columns correspond to the KNX control system side. To know more about each parameter, see KNX Points Definition (page 13).

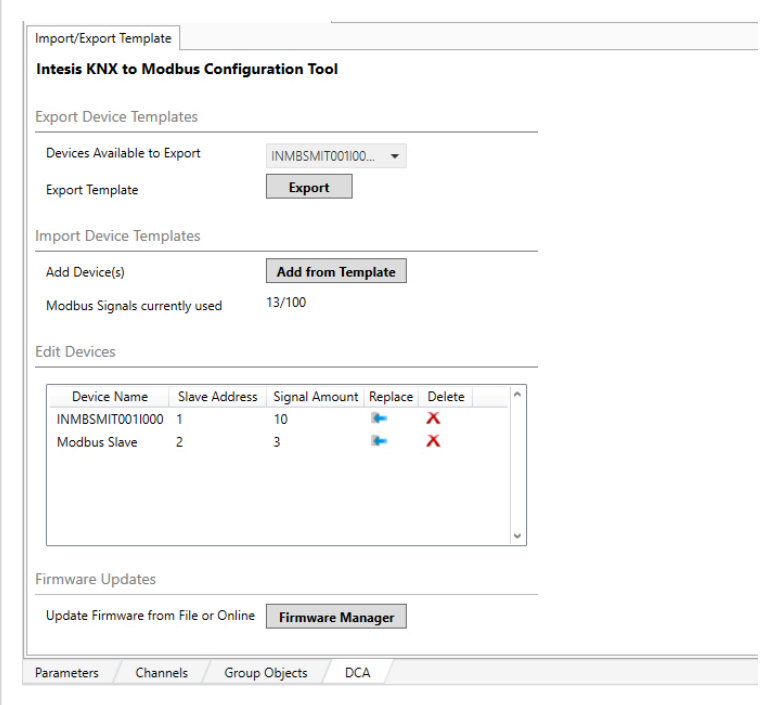
The rest of the columns correspond to the Modbus installation side. To know more about each parameter, see Modbus Points Definition (page 14).

## 6.5. Intesis Modbus Device Manager DCA

**NOTE**
The DCA Tab is a product-specific app and it is not available for ETS demo versions. You can download it from the KNX shop: look for **Intesis Modbus Device Manager**.

If you have any questions about how to activate the DCA, read this KNX article.

*Figure 5. DCA main menu*




In the main menu, the following features are available:

***Export Device Templates***

- Click the **Export** button to export the configuration of a Modbus device in a .knxmbr file format and save it on your PC.

***Import Device Templates***

- Click the **Add from Templates** to open the import window.

  **NOTE**
  Select which signals you want to import using the **Active** checkbox.

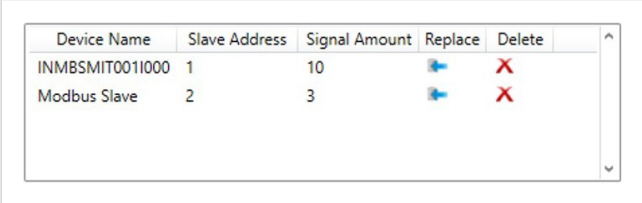
  – Click the **Import File** button to import any template saved on your PC.
  – Click the **Download** button to download a template from the cloud.

*Figure 6. Import window*

### *Edit devices*

- Click the **Replace** icon to replace that device with a template.
- Click the **Delete** icon to delete that device.

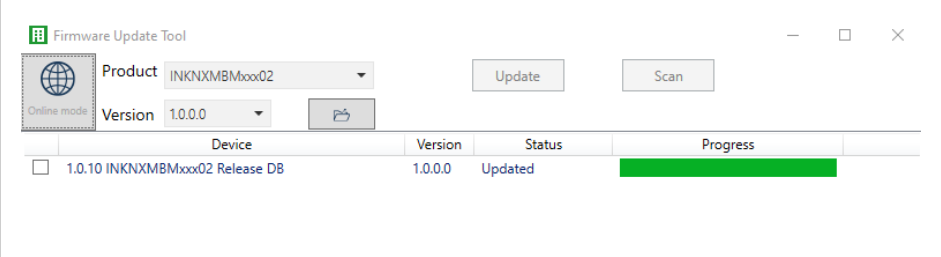
*Figure 7. Edit devices from the DCA main menu*




### *Update the firmware*

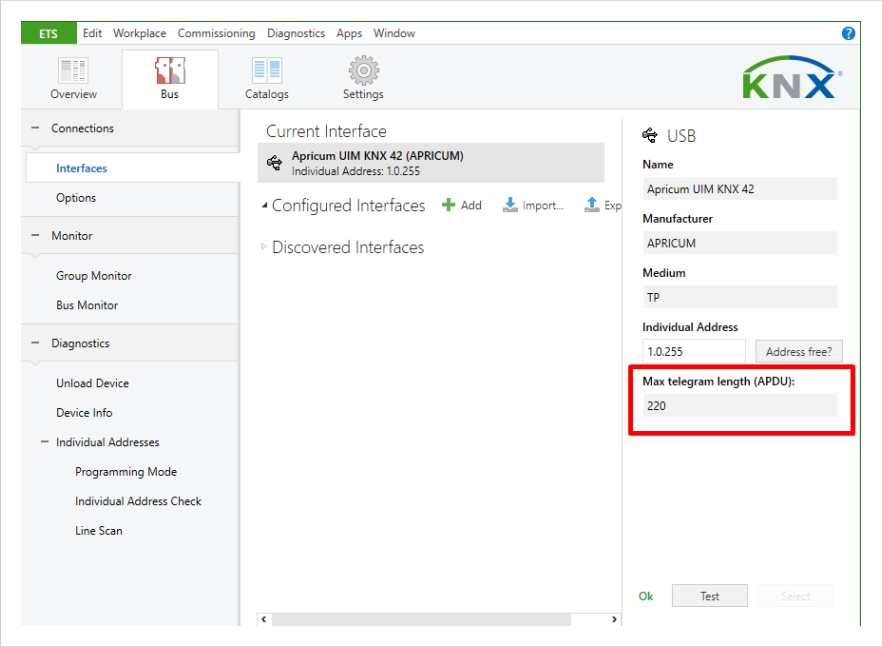
- Click the **Firmware Manager** button to update the firmware version of the gateway.

*Figure 8. Firmware Update Tool*




**IMPORTANT**
This gateway requires a KNX programming interface supporting extended addresses range.

**NOTE**
The firmware download speed depends on the APDU capacity of the interface you are using and the current bus traffic. The higher the APDU number, the higher the download speed.




## 6.6. Sending the Configuration to the Gateway

**IMPORTANT**
Depending on the data you have to send, remember to push the gateway's button to activate the programming mode. See Push Button (page 9).

When the configuration is finished, use the standard procedure to download the configuration from ETS to the Intesis gateway through the **Download** variants.