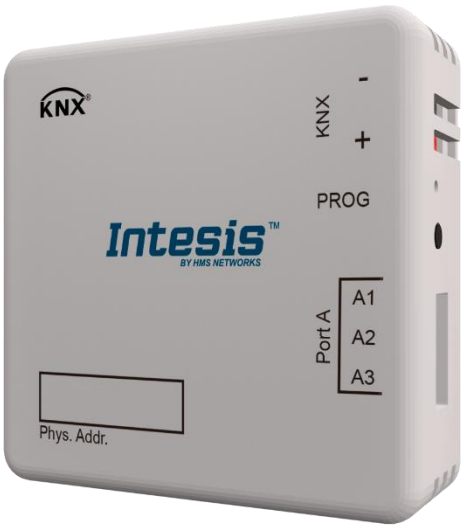
KNX

Modbus RTU Master

USER MANUAL

Issue date: 04/2020 r1.0 ENGLISH



Important User Information

Disclaimer

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

HMS Industrial 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 Industrial Networks and is subject to change without notice. HMS Industrial 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 Industrial 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 Industrial 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.

Gateway for the integration of Modbus RTU installations into KNX TP-1 (EIB) enabled monitoring and control systems.

ORDER CODE
INKNXMBM1000100

INDEX

1 Description

1.1 *Introduction*

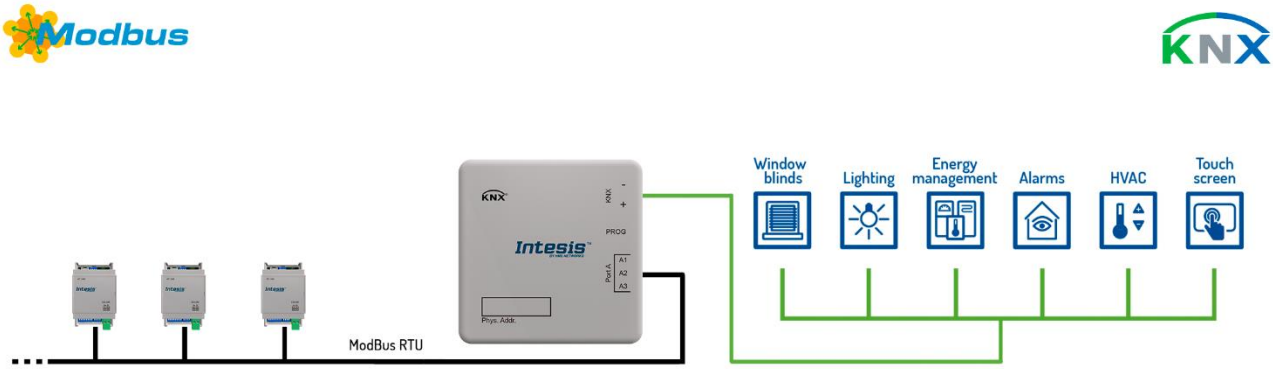
This document describes the integration of Modbus RTU installations into KNX systems using the Intesis *KNX–Modbus Master* gateway.

The aim of this integration is to make accessible Modbus system signals and resources from a KNX control system or device, as if it was a part of the own KNX TP-1 (EIB) system and vice-versa.

For this, Intesis acts as a Modbus RTU master device in its Modbus interface, reading/writing registers of the Modbus slave device(s), and offering these values through its KNX interface, acting in the KNX system as another KNX device of the system. The readings of the Modbus slave device(s) are performed by Intesis gateway automatically through a continuous polling.

Configuration is carried out through ETS configuration tool.

This document assumes that the user is familiar with KNX and Modbus technologies and their technical terms.



Integration of Modbus RTU Slaves
into KNX TP-1 (EIB)s control and monitoring systems

1.2 *Functionality*

From the Modbus system point of view, after the start up process, Intesis reads continuously the points configured to be read in the Modbus RTU Slave devices and updates in its memory all the values received from the Modbus system.

Every one of the mentioned Modbus points is associated to a *KNX group address*, with this, all the Modbus system is seen as *one more KNX device* from the KNX system point of view, with the same configuration and operation characteristics.

When a change in any Modbus point occurs, a write telegram is sent to the KNX bus, of the associated *KNX Group*.

When a telegram is received from the KNX bus to the *KNX Group address* associated to a Modbus point, a message is sent immediately to the corresponding Modbus device to perform the corresponding action.

In the continuous polling of the Modbus devices, if a non-response of the Modbus device is detected, the corresponding virtual signal inside Intesis gateway will be activated indicating that there is a communication error with the Modbus device. These virtual signals indicating communication status in real time with the Modbus devices are also accessible from KNX, like the rest of the points of Intesis Gateway.

1.3 *Gateway's capacity*

Intesis capacity is listed below:

Element	100 version	Notes
Number of Communication Objects	100	Maximum number of Communication Objects that can be defined.
Number of Main Group Addresses	100	Maximum number of KNX Main Group Addresses to be used
Number of Associations	200	Maximum number of KNX Association
Type of Modbus slave devices	Modbus RTU (EIA485)	Those supporting Modbus *protocol*.
Number of Modbus Slave devices	Up to 255 Devices	Number of Modbus Slave devices supported by the device

2 KNX System

In this section, a common description for all Intesis KNX series gateways is given, from the point of view of KNX system which is called from now on *internal system* The Modbus system is also called from now on *external system*.

2.1 *Description*

Intesis gateway KNX connects directly to the KNX TP-1 (EIB) bus and behaves as one more device into the KNX system, with the same configuration and operational characteristics as other KNX device.

Internally, the circuit part connected to the KNX bus is opto-isolated from the rest of the electronics.

Intesis KNX receives, manages and sends all the telegrams related to its configuration to the KNX bus.

When receiving telegrams of KNX Groups associated to internal datapoints, the corresponding messages are sent to the external system (Modbus) to maintain both systems synchronised in every moment.

When a change in a signal of the external system is detected, a telegram is sent to the KNX bus (with the associated KNX group) to maintain both systems synchronised in every moment.

The status of the KNX bus is checked continuously and, if a bus drops down is detected, due to a failure in the bus power supply for example, when the KNX bus is restored again, Intesis will retransmit the status of all the KNX groups marked as *"T" Transmit*. Also, the *Updates* of the groups marked as *"U" Update* will be performed. The behaviour of each individual point into Intesis is determined by the flags configured for the point.

2.2 *Comms status Communication Objects*

Intesis gateway has several KNX communication objects to get information about Modbus errors in Modbus communications.

Number	Name	DPT	Description
1000	Status_Error/Alarm	1.005 [0-No Alarm; 1-Alarm]	It reports if there is some error on the Modbus communications. If there are timeouts or exceptions.
1001	Status_Modbus Dev Error	8.* 2-byte signed value [Slave Id]	In case of alarm or error, it reports Modbus device address which has errors or alarms.
1002	Status_Error Text	16.001 Character String	It is a String with all information about the alarm or error “ss:ff:aaaa:ee”, where: • ss: SlaveId (“00”…”3F”) • ff: Function code (“00”…”10”) • aaaa: Address (“0000”…”FFFF”) • ee: Exception code (“00”…”0F”) Exception codes are the same as in Modbus plus “0F” which is the comm error.

2.3 *Points definition*

Every internal datapoint to define has the following KNX properties:

Property	Description
Object name	Descriptive information about the Communication Object or Signal.
DPT	It is the KNX data type used to code the signal's value. It will depend on the type of signal associated in the external system in every case. In some integrations, it is selectable, in others it is fixed due to the intrinsic characteristics of the signal.
Active	If activated, the point will be active in Intesis gateway, if not, the behaviour will be as if the point is not defined. This allows deactivating points without the need of delete them for possible future use.

3 Modbus interface

3.1 *Description*

Intesis gateway acts as master in the Modbus RTU Network and the other Modbus devices connected to the same network communicating with Intesis must be always slave devices.

The Modbus protocol defines different types of function codes to use to read/write different type of registers that can be found in Modbus devices, and also different data formats to encode values. See below (section 3.2) for all the function codes and data formats supported by Intesis's Modbus interface.

Furthermore, the data encoding used for 16 bits registers (big-endian or little-endian) can be configured in Intesis' Modbus interface. This is the byte order for data encoding (MSB..LSB or LSB..MSB). This data encoding, although is specified as big-endian in Modbus protocol specification, it varies depending on manufacturer/type of slave.

This functionality provides a great flexibility to integrate a wide range of Modbus slave devices that can be found in the market.

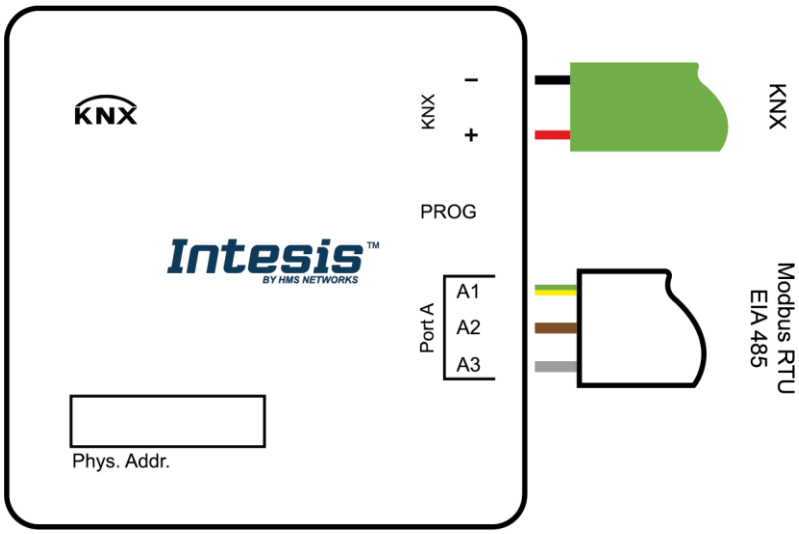
3.2 *Points definition*

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

Feature	Description
# Slave	Modbus slave number to which the point belongs to.
Function code **Read Func** **Write Func**	**One of the following Modbus function codes can be used:** • 1- Read Coils. • 2- Read Discrete Inputs. • 3- Read Holding Registers. • 4- Read Input Registers. • 5- Write Single Coil. • 6- Write Single Register. • 15- Write Multiple Coils. • 16- Write Multiple Registers.
Data Coding Format	**One of the following Modbus data coding formats can be used:** • 16/32/48/64 bits unsigned. • 16/32/48/64 bits signed (one's complement – C1). • 16/32/48/64 bits signed (two's complement – C2). • 16/32/48/64 bits Float. • 16/32/48/64 bits Bitfields. • Error comm
Byte Order	• Big Endian • Little Endian • Word Inverted Big Endian • Word Inverted Little Endian
Register Address	The Modbus register address inside the slave device for the point.
Bit inside the register	Bit inside the Modbus register (optional). The gateway allows bit decoding from generic 16 bits input/holding Modbus registers. *Bit coding into 16 bit input/holding Modbus registers is used for some devices to encode digital values into this type of registers, being these registers normally accessible using Modbus function codes 3 and 4 (read holding/input registers).*
#Bits	Number of bits to be used by this signal.
Operation	Opeations that can be applied to Modbus values before to be sent to KNX network. On the reverse way, the inverse operation is applied to KNX values before to be sent to Modbus network. The next operations can be used: • **-** (No operation) • **x10** – Modbus value x 10; KNX value / 10 • **/10** – Modbus value / 10; KNX value x 10 • **x100** – Modbus value x 100; KNX value / 100 • **/100** – Modbus value / 100; KNX value x 100 • **Is equal** – Checks if Modbus value is equal to Operation value configured, if so, send 1 to KNX, otherwise, send 0.
Operation value	The operation value is the operand of the operations, which need it. Only **Is equal** needs an operand.

4 Connections

Find below information regarding the Intesis connections available.



KNX Port
Connect the KNX TP1 bus to connectors + and - of gateway's KNX Port. Respect the polarity

Port A / Modbus RTU
Connect the EIA485 bus to connectors A2 (A-), A3 (B+) and A1 (SNGD) of gateway's Port A. Respect the polarity.

Note for Port A; *Remember the characteristics of the standard EIA485 bus: maximum distance of 1200 meters , maximum 32 devices connected to the bus, and in each end of the bus it must be a termination resistor of 120 Ω. The gateway has an internal bus biasing circuit that incorporates the termination resistor. If you install the gateway in one of the ends of the bus, then do not install an additional termination resistor in that end.*

4.1 *Powering the device*

Power is supplied by power supply present in the KNX network (see section 4.2).

4.2 *Connection to KNX*

Connect the communication cable coming from the KNX network to the KNX Port (Figure above) of Intesis. Respect the polarity.

In case there is no response from the KNX installation or KNX devices to the frames sent by Intesis, check that they are operative and reachable from the network connection where the Intesis is located.

Check as well if there is a line coupler that it is not filtering the telegrams from/to the Intesis.

4.3 *Connection to Modbus*

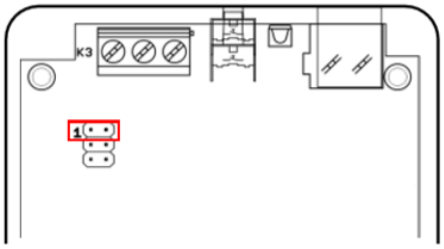
4.3.1 Modbus RTU

Connect the communication cable coming from the Modbus network to the port marked as Port A of Intesis. Connect the EIA485 bus to connectors A2 (A-), A3 (B+) and A1 (SNGD) of gateway's PortA. Respect the polarity.

Remember the characteristics of the standard EIA485 bus: maximum distance of 1200 meters, maximum 32 devices connected to the bus, bus polarization and in each end of the bus it must be a termination resistor of 120 Ω. The gateway has an internal bus biasing circuit that incorporates the termination resistor. If you install the gateway in one of the ends of the bus, then do not install an additional termination resistor in that end.

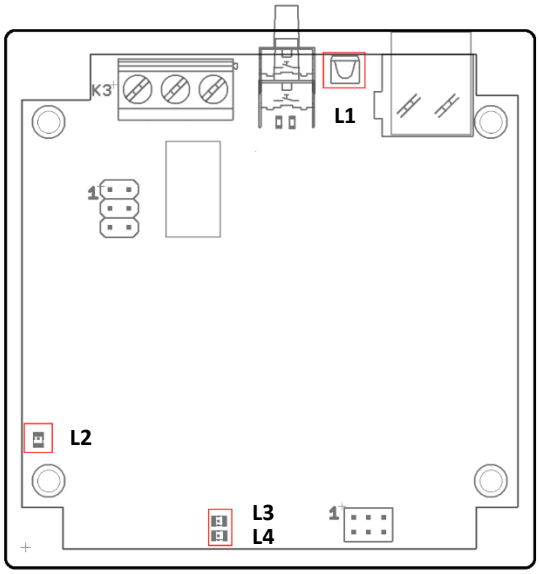
IMPORTANT: If the INKNXMBM1000100 gateway is not placed at one end of the Modbus channel, the terminal resistor should be deactivated. Remove Jumper 1 to deactivate the 120 Ω terminal resistor.

The bus should be polarized only in one location on the line. The INKNXMBM1000100 incorporates 2 jumpers to introduce polarization to the line. It is recommended to keep polarization in the master only. If other device is polarized, remove jumpers 2 and 3 to deactivate polarization in the gateway.



5 Status LEDs

Find below information regarding LEDs present in the device.



LED	Colour	Indication
KNX Programing Mode (L1)	Off	Device isn't in programing mode on KNX
	Blinking red	Device is in programing m ode on KNX
Power (L2)	Off	No power
	Green	Device powered and working.
Port A activity (L3)	Off	There is no answer from Slave devices.
	Blinking green	Every valid answer received from a Slave device it blinks.
Port A communication (L4)	Red	There is no communication with any of the configured slave devices.
	Green	There is communication with at least one of the configured slave devices.

6 Set-up process and troubleshooting

6.1 *Pre-requisites*

It is necessary to have a KNX device or interface operative and well connected to the corresponding KNX port of Intesis and a Modbus RTU slave connected to their corresponding port as well.

Connectors, connection cables, PC to use the configuration tool and other auxiliary material, if needed, are not supplied by HMS Industrial Networks S.L.U for this standard integration.

Items supplied by HMS Networks for this integration are:

- Intesis gateway.
- Product documentation.
- Product ETS database.

6.2 *Configuration and Setup*

6.2.1 Introduction

This is a fully compatible KNX device which must be configured and setup using the standard KNX tool ETS.

ETS database for this device can be downloaded from:

http://intesis.com/docs/configuration-files/inknxmbm1000100_ets_database

6.2.2 ETS Parameters

When imported to the ETS software for the first time, the gateway has 10 Modbus signals active, each one configured as a Read and Write signal, so configuration looks like:

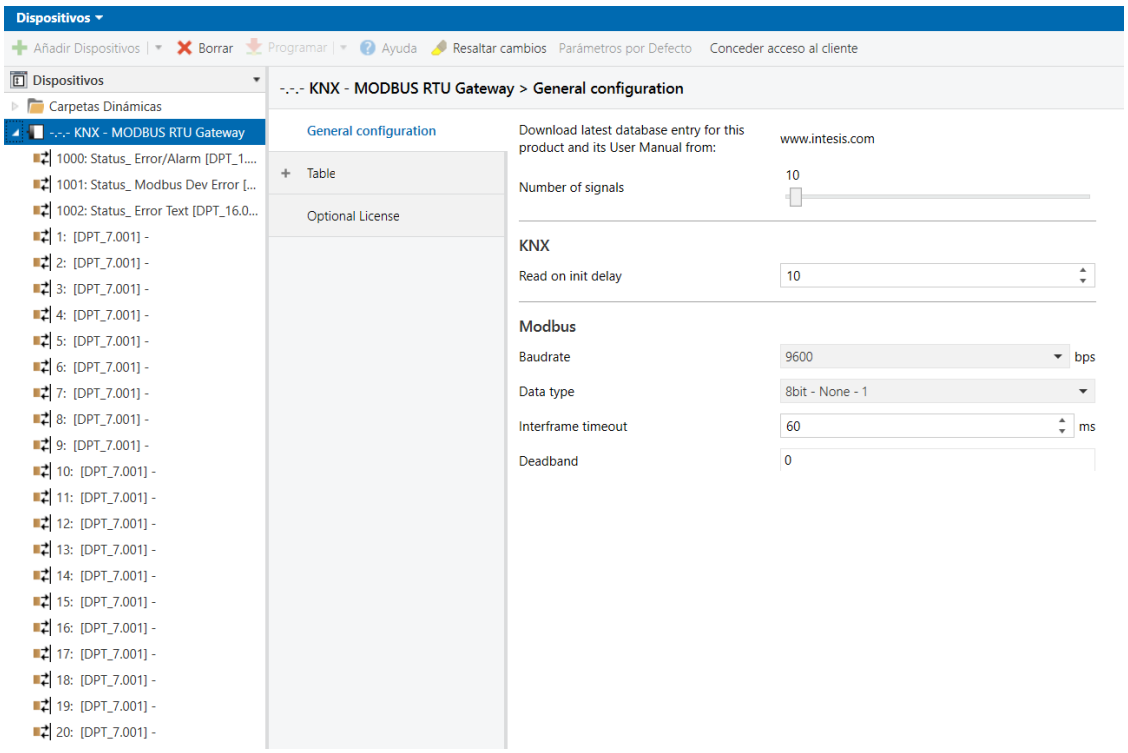


Figure 6.1 Default configuration

6.2.3 General configuration

Select ***General configuration*** tab to configure the Modbus connection parameters and number of signals to integrate.

- **Number of signals.** Define number of Modbus signals to integrate. Once defined, it will create all communication objects required on the ETS Project.

- **Read on init delay.** Time after initialization process to start uploading of all values in KNX network.

- Modbus connection parameters.

    - **Baudrate.** Defines the communication speed for the RTU communication. Values from 2400 to 115200 bps.
    - **Data Type.** On the first field, it defines the number of bits used for data, on the second field it defines the parity of the communication (None/Odd/Even), on the last field it defines the number of stop bits.
    - **Interframe timeout.** Minimum time between received frame and sent frame.
    - **Deadband.** Minimum COV (change of value) in a signal value to update it in the KNX system.

Once general configuration is finished, go to Table tab, to configure all Modbus registers.

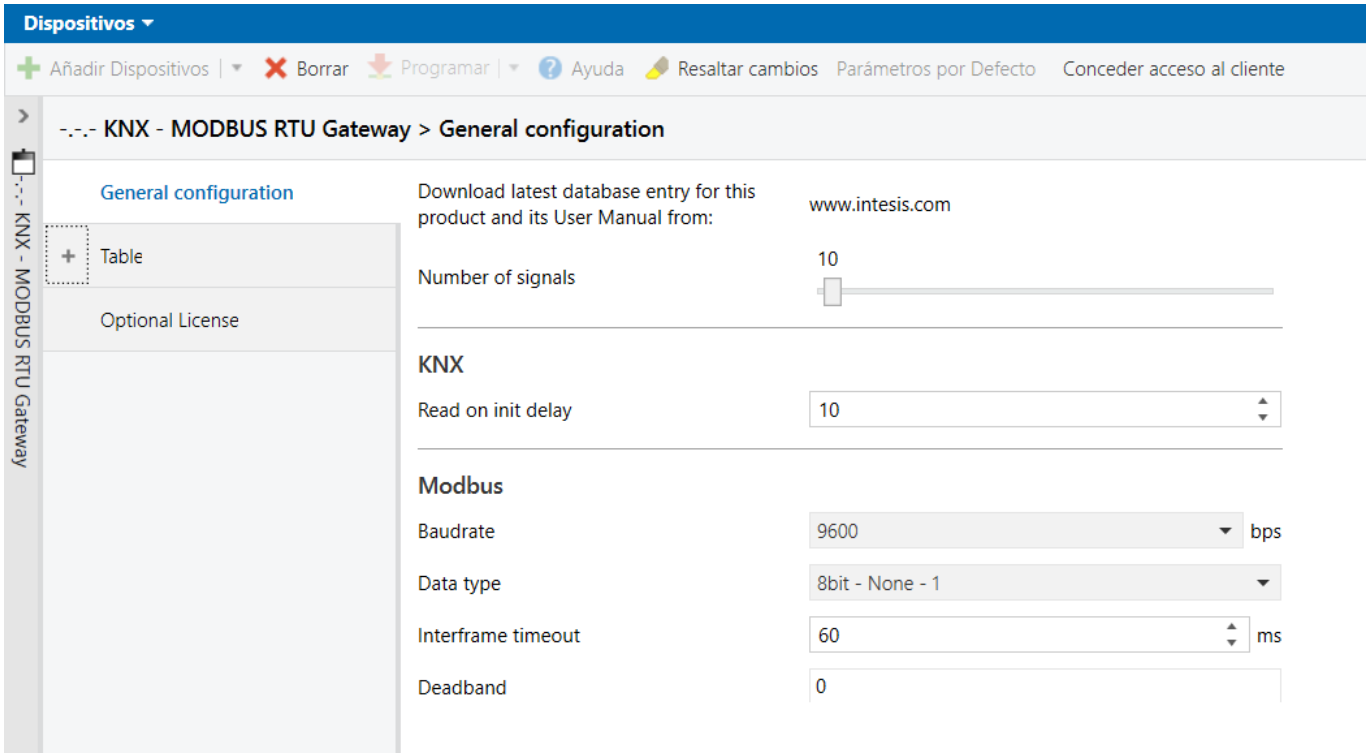
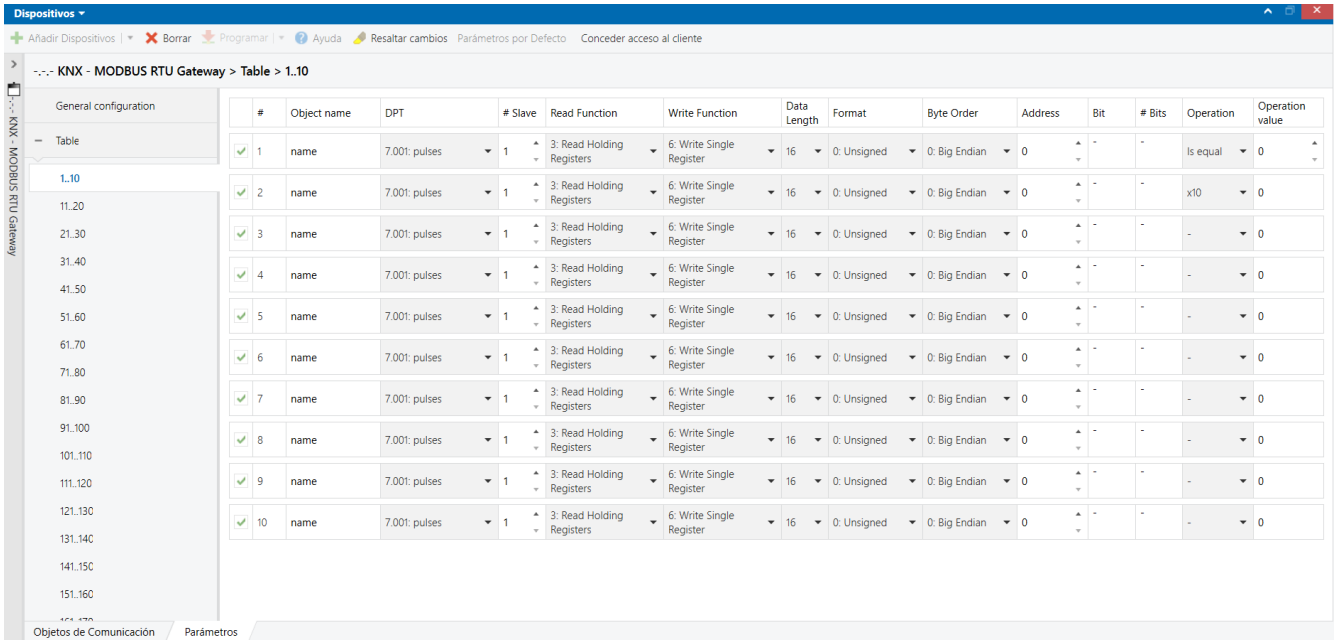


Figure 6.2 General Configuration

6.2.4 Table

All available KNX objects and its corresponding Modbus register and other main parmaters are listed in this table.



#	Object name	DPT	# Slave	Read Function	Write Function	Data Length	Format	Byte Order	Address	Bit	# Bits	Operation	Operation value
1	name	7.001: pulses	1	3: Read Holding Registers	6: Write Single Register	16	0: Unsigned	0: Big Endian	0	-	-	Is equal	0
2	name	7.001: pulses	1	3: Read Holding Registers	6: Write Single Register	16	0: Unsigned	0: Big Endian	0	-	-	x10	0
3	name	7.001: pulses	1	3: Read Holding Registers	6: Write Single Register	16	0: Unsigned	0: Big Endian	0	-	-	-	0
4	name	7.001: pulses	1	3: Read Holding Registers	6: Write Single Register	16	0: Unsigned	0: Big Endian	0	-	-	-	0
5	name	7.001: pulses	1	3: Read Holding Registers	6: Write Single Register	16	0: Unsigned	0: Big Endian	0	-	-	-	0
6	name	7.001: pulses	1	3: Read Holding Registers	6: Write Single Register	16	0: Unsigned	0: Big Endian	0	-	-	-	0
7	name	7.001: pulses	1	3: Read Holding Registers	6: Write Single Register	16	0: Unsigned	0: Big Endian	0	-	-	-	0
8	name	7.001: pulses	1	3: Read Holding Registers	6: Write Single Register	16	0: Unsigned	0: Big Endian	0	-	-	-	0
9	name	7.001: pulses	1	3: Read Holding Registers	6: Write Single Register	16	0: Unsigned	0: Big Endian	0	-	-	-	0
10	name	7.001: pulses	1	3: Read Holding Registers	6: Write Single Register	16	0: Unsigned	0: Big Endian	0	-	-	-	0

Figure 6.3 Table tab

For the KNX configuration as an internal protocol, the following parameters need to be considered:

1. **Object name**
   Descriptive name of the signal. It will be used to set the name of their communication objects.

2. **DPT**
   Select the KNX Data Point Type (DPT) to be used for each signal or KNX communication object.

For the Modbus configuration as an external protocol, the following parameters need to be considered:

1. **# Slave**
   It indicates the slave number where signal is.

2. **Read Function**
   It indicates the Modbus function used to read, if allowed or required. Modbus functions 1, 2, 3 and 4 supported.

3. **Write Function**
   It indicates the Modbus function used to write, if allowed or required Modbus functions 5, 6, 15 and 16 supported.

4. **Data Length**
   It indicates the signal size expressed in bits.

5. **Format**
   It indicates the register information format. Unsigned, Signed C1 (one's complement), Signed C2 (two's complement), Float and Bit Fields.

6. **Byte Order**
   It indicates the byte order.

7. **Address**
   It indicates the register signal starting address.

8. **Bit**
   If using multiple bit (bit fields), it indicates the first bit you want to read

9. **# Bits**
   If using multiple bit (bit fields), it indeicates the number of bits you want to read

10. **Operation**
    It indicates if any operation is being applied to the signal. Posible operetations are multiplication and division per 10 and 100; and relational operator equal.

11. **Operation value**
    It defines an operand for the operation configured. Only **Is equal** operation allow to configure the operand.

6.2.5 Sending the configuration to Intesis

When the configuration is finished, use the standard way of ETS software to download the configuration to Intesis gateway, using the download button.

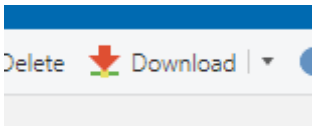


Figure 6.4 Downoad button from ETS

6.3 *Set-up procedure*

1. Install Intesis in the desired installation site. Installation can be on a stable not vibrating surface.
2. Connect the KNX communication cable coming from the KNX network to the port marked as KNX Port on Intesis (More details in section 4).
3. Connect the communication cable coming from the EIA485 port of the Modbus RTU installation to the port marked as Port A of Intesis (More details in section 4).
4. Open ETS, open a project and add product from ETS catalog.
5. Modify the configuration as desired, and then download to the device.

7 Electrical & Mechanical Features

Enclosure	Plastic, type ABS (UL 94 V-0) Net dimensions (dxwxh): 71x71x27 mm Color: White. RAL 9010
Mounting	Wall.
Power	Supplied through KNX bus. See on KNX Port.
KNX Port	1 x KNX TP-1 Plug-in screw terminal block (2 poles) 2500VDC isolation from other ports KNX power consumption: 20mA Voltage rating: 29VDC
PORT A	1 x Serial EIA485 Plug-in screw terminal block (3 poles) A, B, SGND (Reference ground or shield) 1500VDC isolation from other ports
Push Button	Sets device in programing mode in KNX network
Operation Temperature	0°C to +60°C
Operational Humidity	5 to 95%, no condensation
Configuration Jumpers	3 x Jumpers for serial EIA485 configuration: Jumper 1: **Connected:** 120 Ω termination active. **Disconnected:** 120 Ω termination inactive. Jumper 2 & 3: **Connected:** Polarization active. **Disconnected:** Polarization inactive.
LED Indicators	3 x Onboard LED indicators 2 x Port A TX/RX 1 x KNX Prog Mode