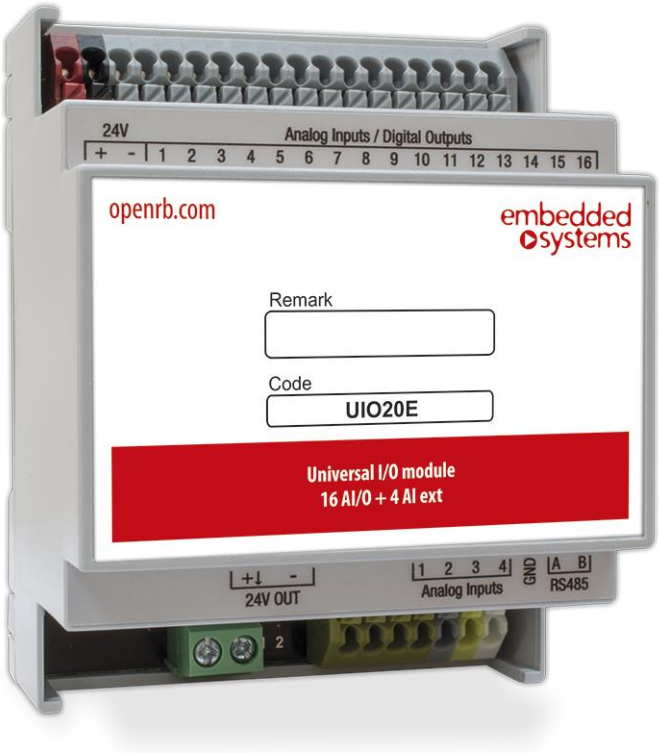
Embedded Systems SIA, VAT No LV40003411103
47. Katolu str., Riga, LV 1003, LATVIA
Phone: +371 67648888, fax: +371 67205036, e-mail: sales@openrb.com

Universal 20 channel I/O device



ENG - Data sheet

Issue date 19.09.2013

Application

Universal 20 channel IO device is designed to be used in meter reading applications or other building automation applications as an extension module to LogicMachine series devices. The configuration of the device is done through separate Modbus tab in LogicMachine. 4 channels can be used as analog 0-30V inputs, binary inputs. 16 channels can be used as analog 0-30V inputs, binary inputs, digital outputs. The device is designed for DIN-rail mounting and requires 4 DIN-units.

Types of product

UIO20E	Universal I/O module 16 AI/O + 4 AI

Standards and norms compliance

CE conformity:	EMBS-CE-110926/01	Electromagnetic compatibility
EMC:	EN61000-6-1 EN61000-6-3	
PCT	Certificate	

Technical data:

Power supply:	24V DC	Power supply
	Power consumption	12 mA
Interface:	Universal Inputs/Outputs	16
	Inputs	4
	Input resolution	12bits
	RS-485	1
Clamps:	KNX/EIB	1.5mm2
	Inputs/Outputs	1.5mm2
Enclosure:	Material:	Polyamide
	Color:	Gray
	Dimensions:	70(W)x91(H)x56(L) mm
Protection:	IP20 according to EN 60529	
Usage temperature:	-5C ... +55C	
Storage temperature:	-20C ... +70C	
Net weight:	104g	
Gross weight:	122g	

Caution

Security advice

The installation and assembly of electrical equipment may only be performed by skilled electrician. The devices must not be used in any relation with equipment that supports, directly or indirectly, human health or life or with application that can result danger of people, animals or real value

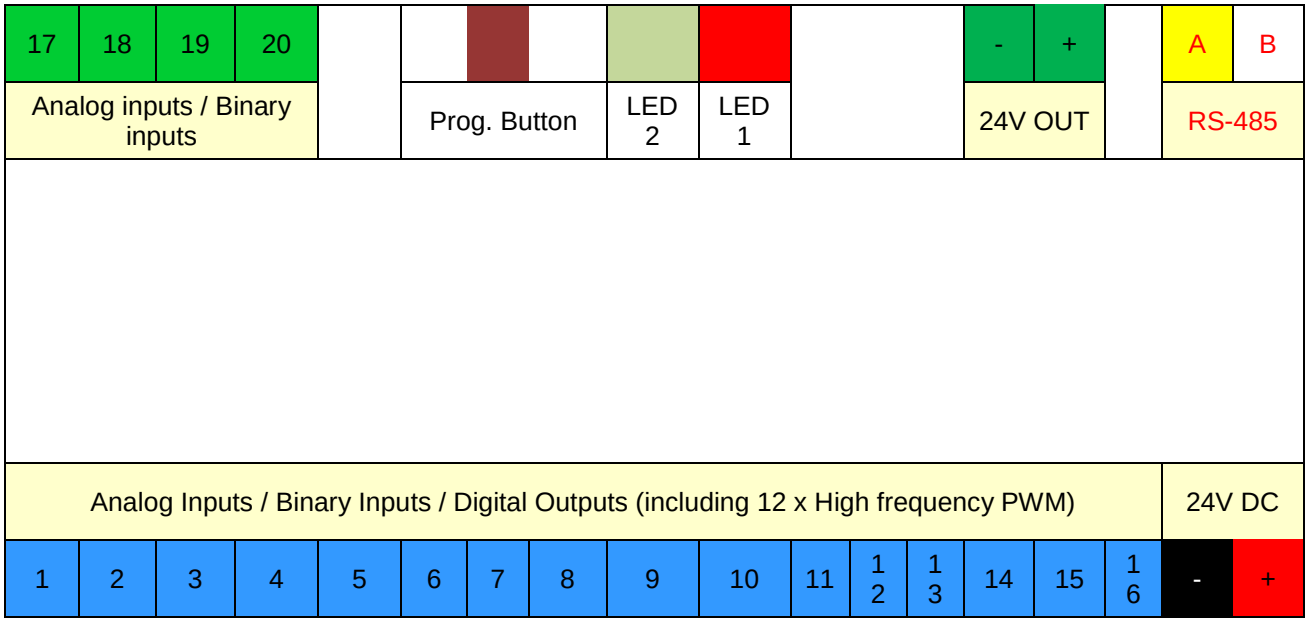
Mounting advice

The devices are supplied in operational status. The cables connections included can be clamped to the housing if required.

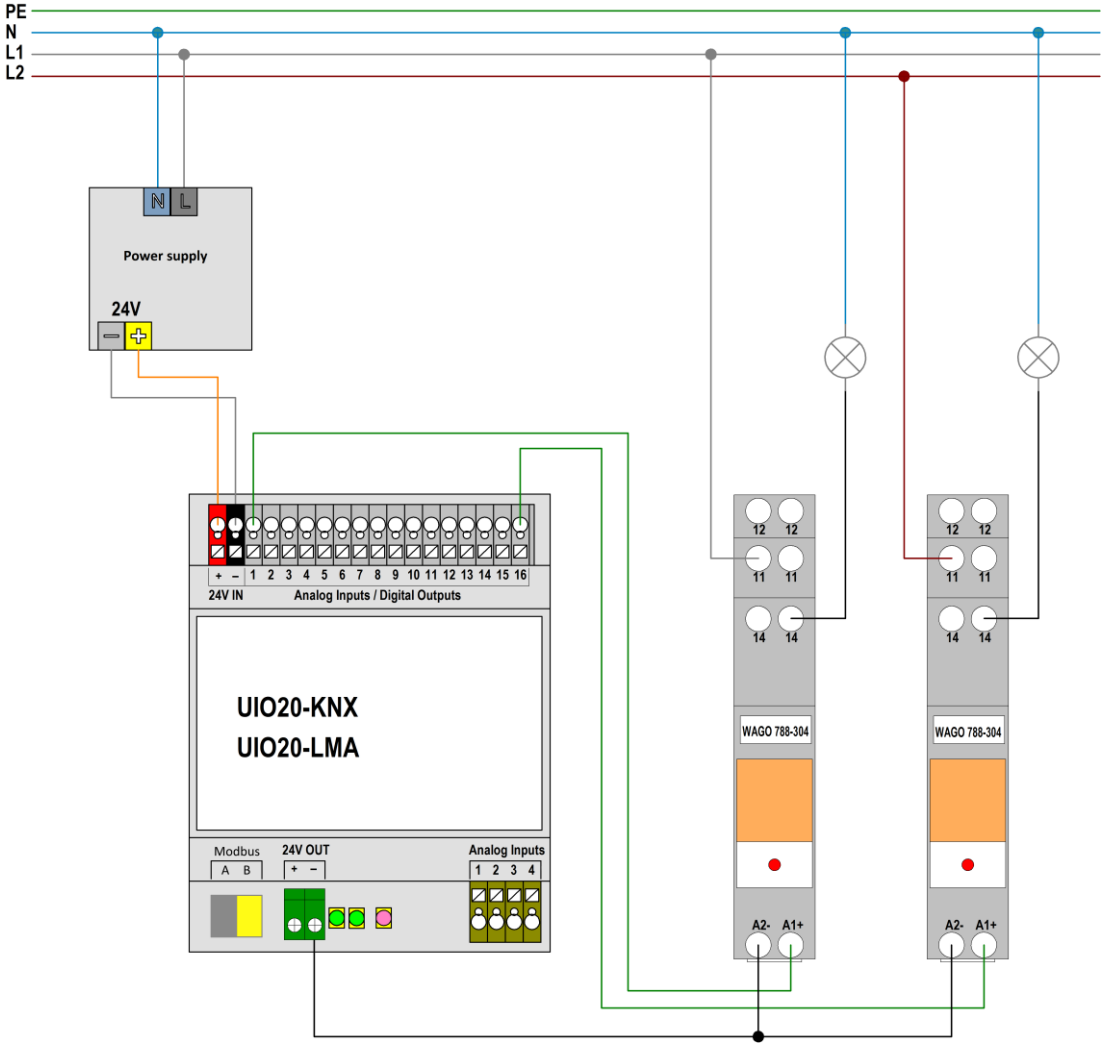
Electrical connection

The devices are constructed for the operation of protective low voltage (SELV). Grounding of device not needed. When switching the power supply on or off, power surges must be avoided.

Terminal connection scheme

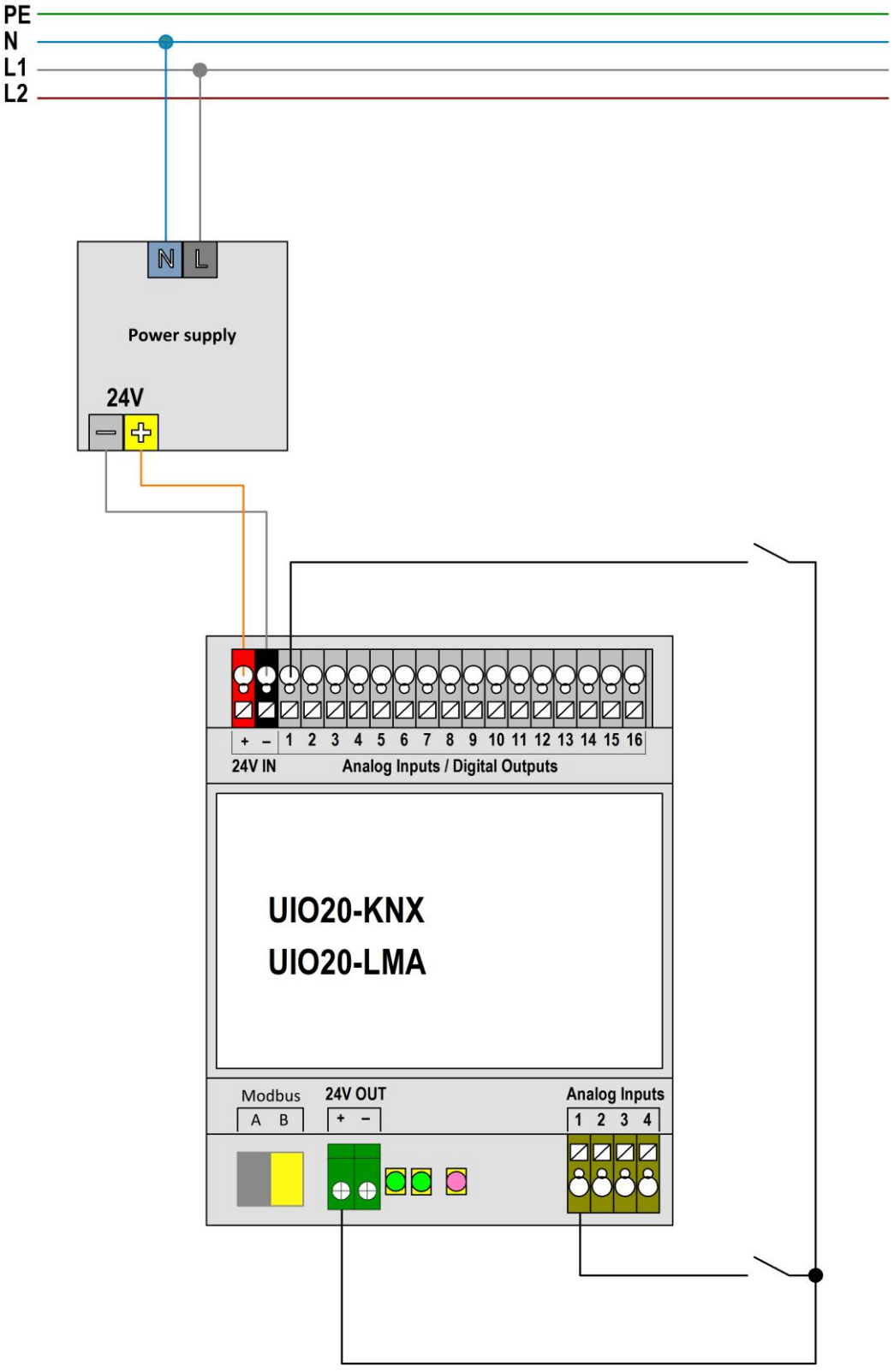


External relay connection

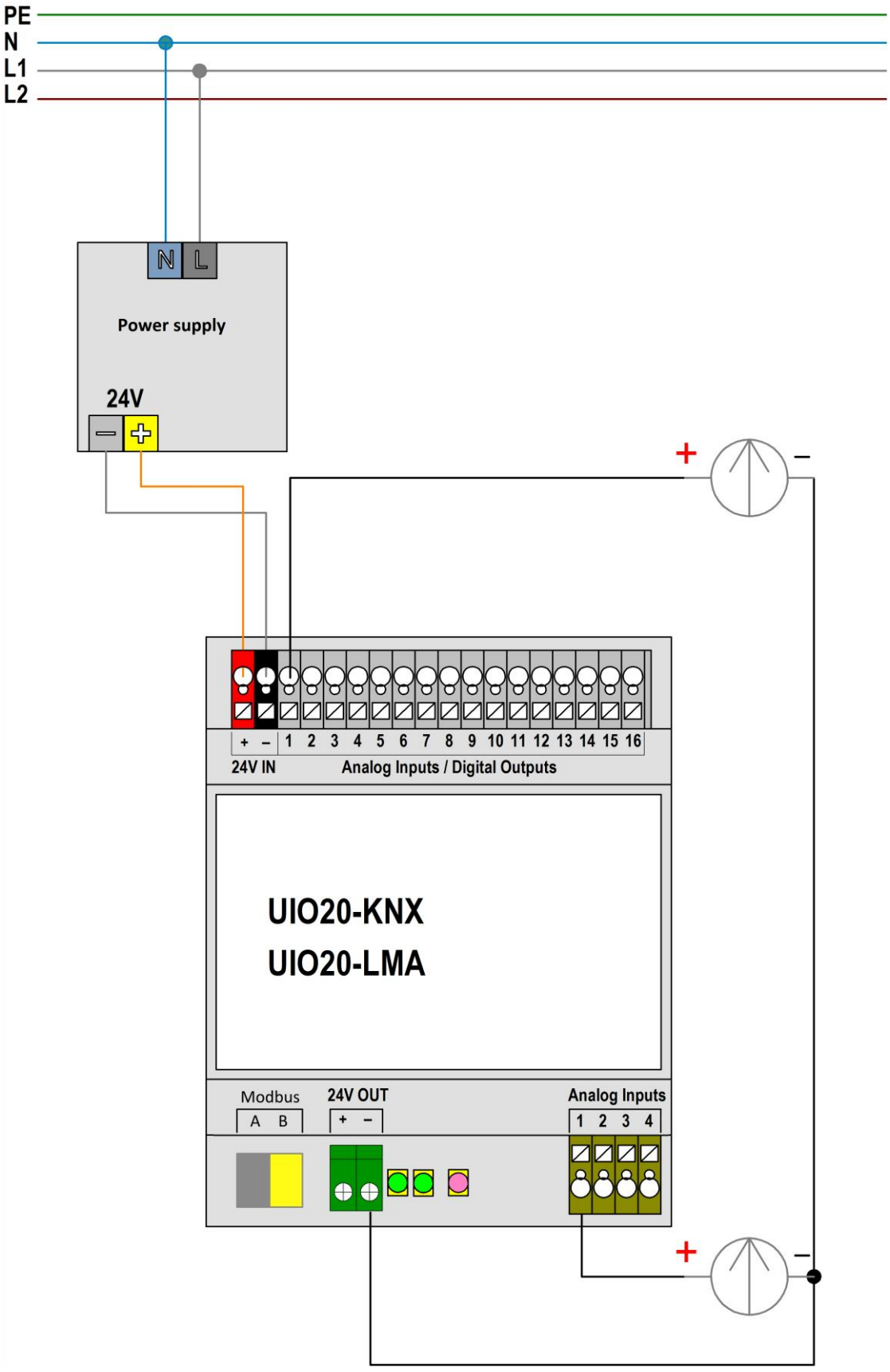


Push-button connection

We do not recommend to interconnect pushbuttons to Modbus device because of the polling – the commands will not be executed in real-time.



Voltage sensor connection

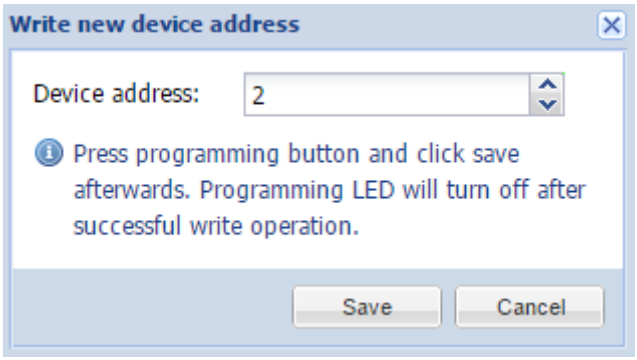


Default address

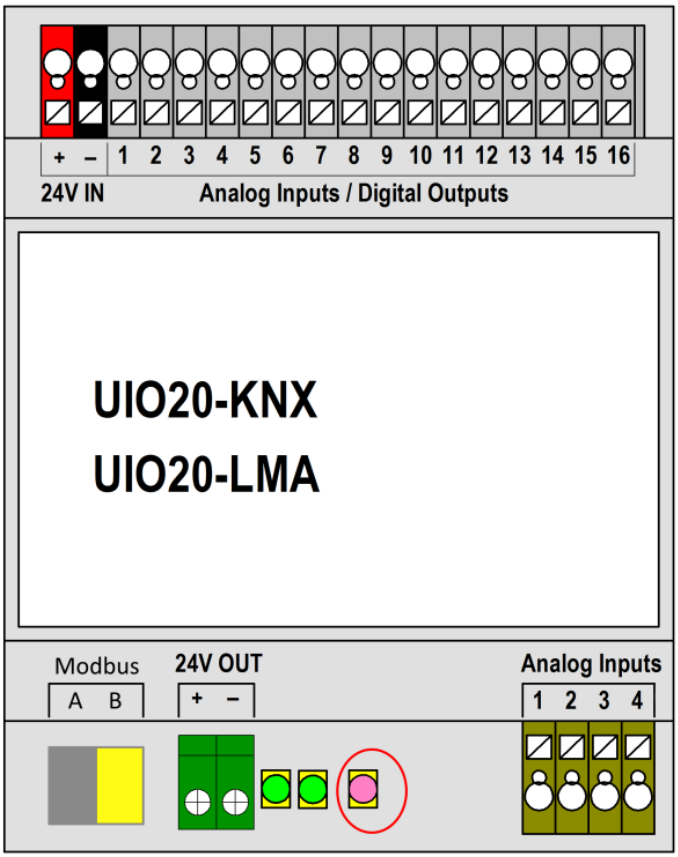
247

Configuring UIO20 on LogicMachine

Program address for UIO20 Modbus device



There is a separate Write address button to program address for UIO20 device. Press programming button and click save afterwards. Programming LED will turn off after successful write operation.



UIO20 Modbus device profile

In LogicMachine Modbus devices are added as *.json profile files. You can download UIO20.json here:

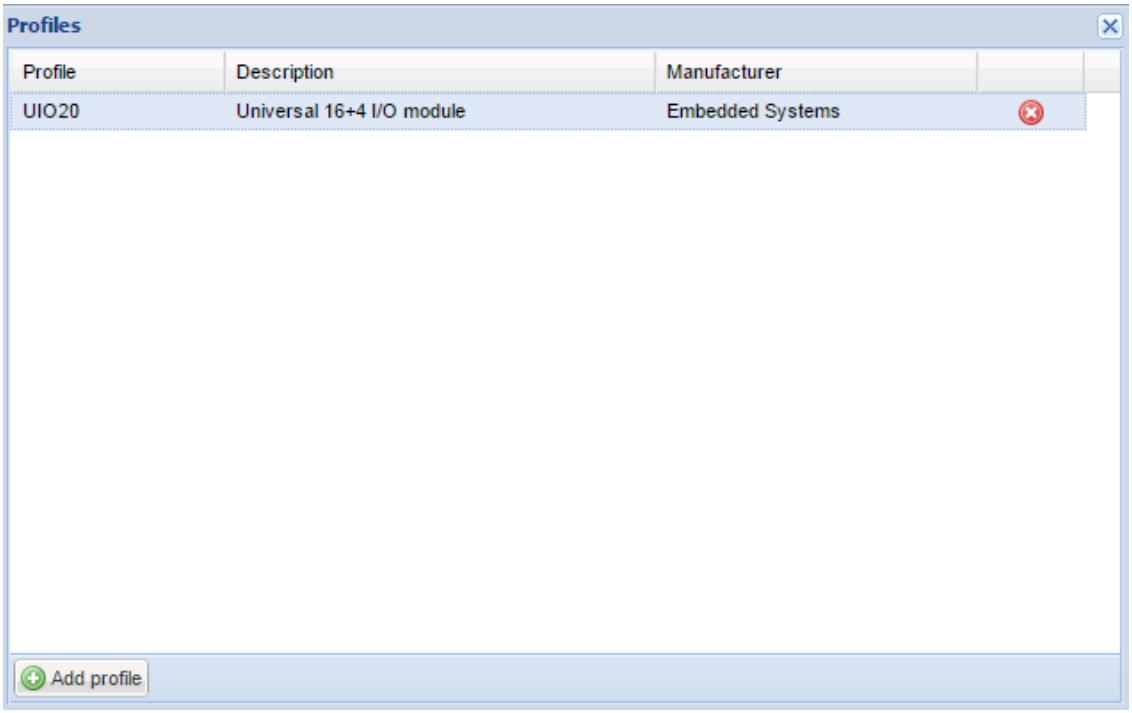
http://openrb.com/wp-content/uploads/2015/06/UIO20.json

A fragment from UIO20 profile file:

{
"manufacturer": "Embedded Systems",
"description": "Universal 16+4 I/O module",
"mapping": [
{ "name": "Output 1", "bus_datatype": "bool", "type": "coil", "address": 0, "writable": 1 },
{ "name": "Input 1", "bus_datatype": "float16", "type": "inputregister", "address": 0,
"value_multiplier": 0.001, "units": "V" }
]
}

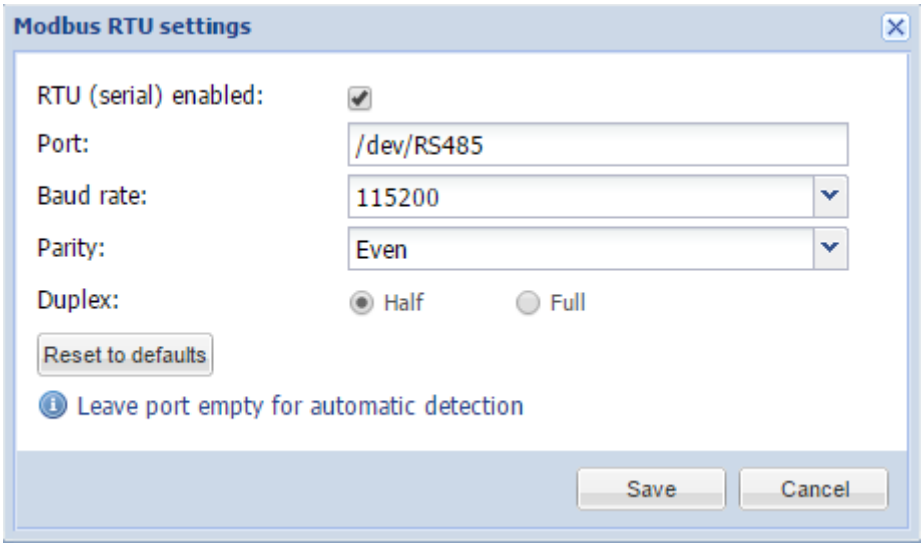
Name – Object name, e.g. Output 2 (String, Required)
Bus_datatype - KNX object data type, key from dt table, e.g. float32 *(String/Number, Required)*
Type – Modbus register type, possible values: coil discreteinput register inputregister *(String, Required)*
Address – Register address (0-based*) (Number, Required)*
Writable - Set to true to enable writing to register if type is either coil or register *(Boolean)*
Datatype – Modbus value data type. If set, conversion will be done automatically. Possible values: uint16 int16 float16 uint32 int32 float32 uint64 int64 quad10k s10k *(String)*
Value_delta – New value is sent when the difference between previously sent value and current value is larger than delta. Defaults to 0 (send after each read) *(Number)*
Value_multiplier – Multiply resulting value by the specified number, value = value_base + value * value_multiplier *(Number)*
Value_bitmask – Bit mask to apply, shifting is done automatically based on least significant 1 found in the mask *(Number)*
Value_nan – Array of 16-bit integers. If specified and read operation returns the same array no further processing of value is done *(Array)*
Value_conv – Apply one of built-in conversion functions *(String, Internal)*
Value_custom – Name of a built-in enumeration or a list of key -> value mapping, resulting value will be 0 if key is not found *(String/Object)*
Internal – Not visible to user when set to true, should be used for scale registers *(Boolean)*
Units – KNX object units/suffix *(String)*
Address_scale – Address of register containing value scale, value = value * 10 ^ scale *(Number)*
Read_count – Number of register to read at once (for devices that only support reading of a specific block of registers) (*Number)*
Read_swap – Swap register order during conversion (endianness) *(Boolean)*
Read_offset – Position of first register of data from the block of registers (0-based) *(Number)*

When the Modbus device profile file is created, upload it by clicking on *Profiles* button.



Modbus RTU settings

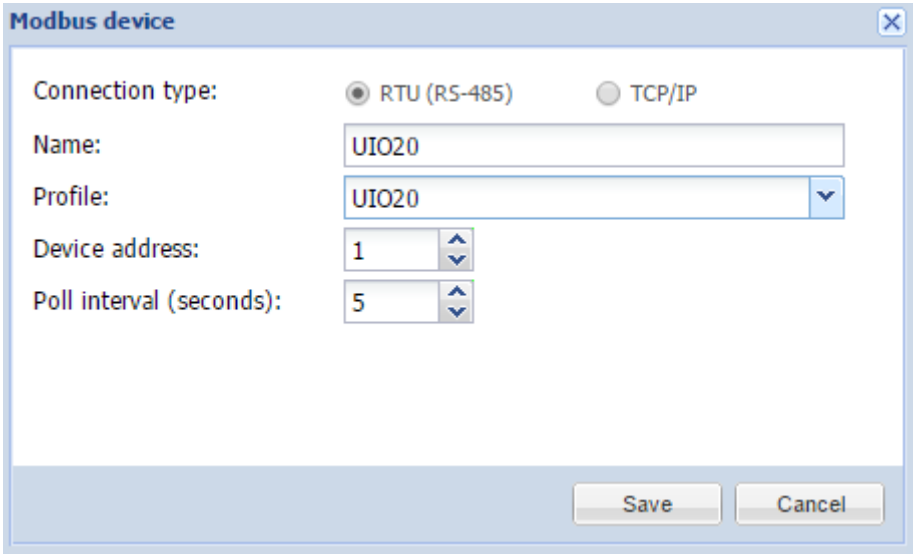
If the communication is over Modbus RTU protocol (over RS-485 serial port), you should do base serial port settings by clicking on *RTU settings* button.



RTU (serial) enabled – define either RTU is enabled or not
Port – port name. In case of several RS-485 ports on the device, the name of the port is incremented by one, e.g. RS485-1, RS485-2, RS485-3 etc.
Baud rate – baud rate for the connection
Parity – parity for the connection
Duplex – specify either it is *half* or *full* duplex
Reset to defaults – reset RTU settings to defaults

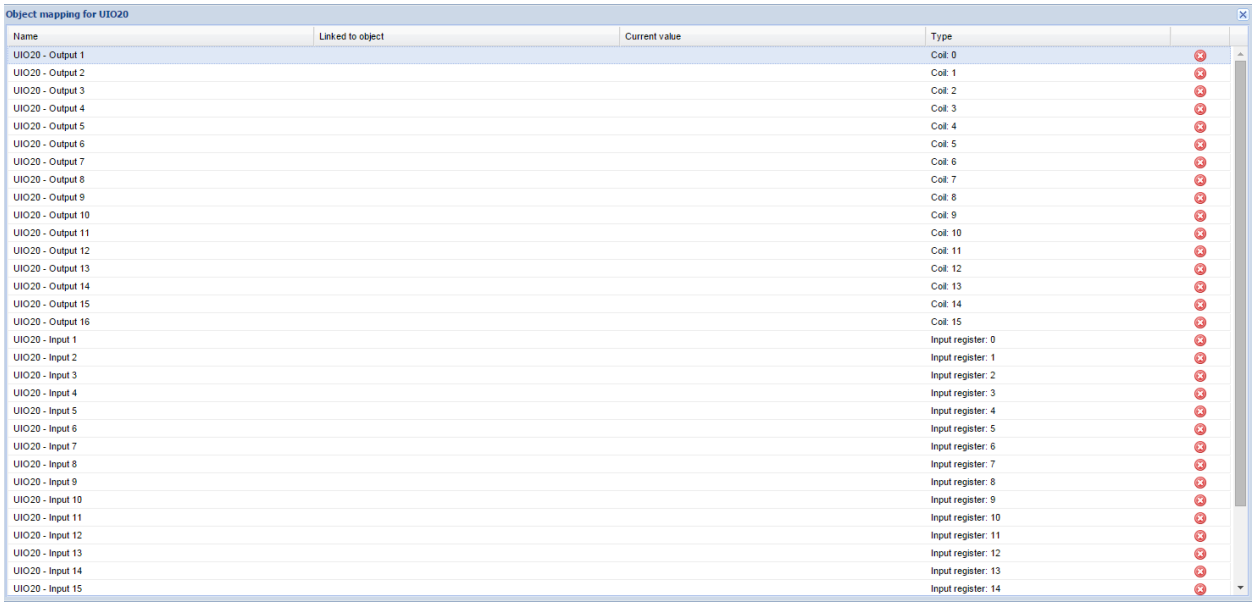
Adding Modbus device

Once profiles are defined and RTU settings set, add Modbus device by clicking *Add device* button.



Connection type – define either it is Modbus RTU or Modbus TCP connection
Name – name of the device
Profile – profile of the device
Device address – device address
Poll interval (seconds) – interval to poll the device
IP – IP address of the device in case Modbus TCP is used
Port – Communication port of the device in case Modbus TCP is used

Once the device is added, you can do mapping to KNX addresses by clicking on  icon. First, you see a list of all objects on the Modbus device.



Click on specific object to do mapping.