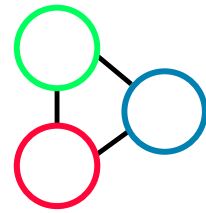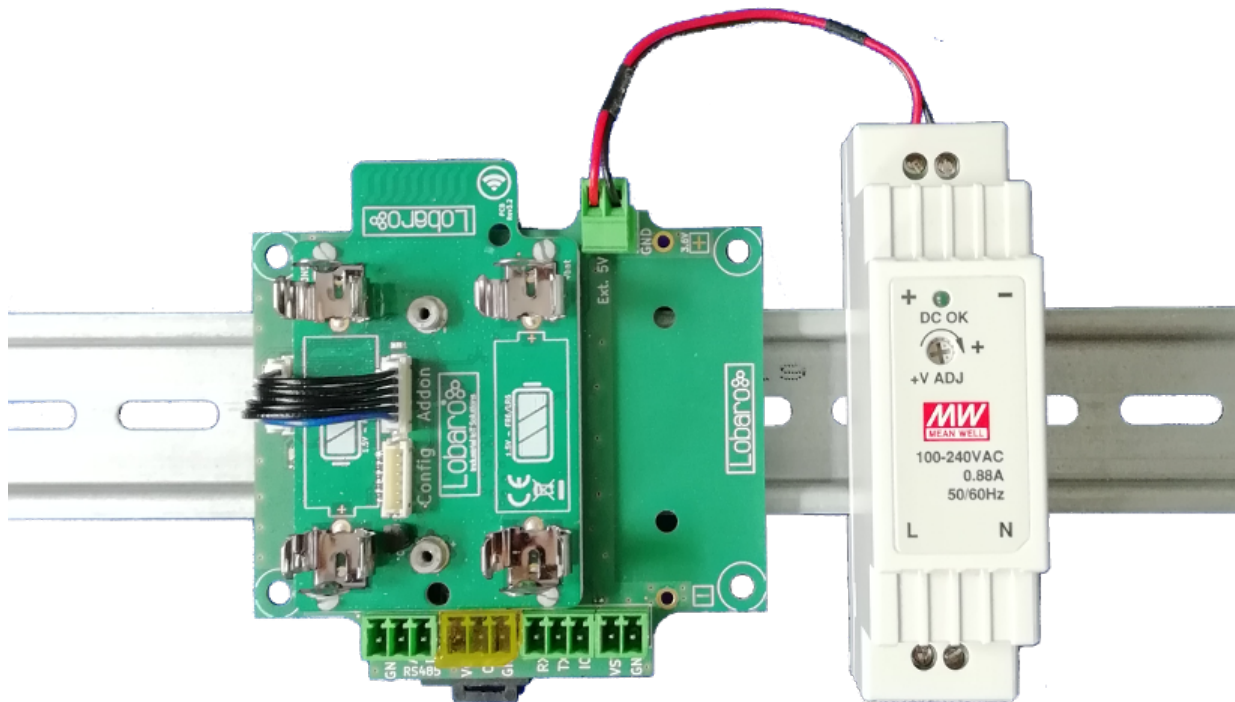# Lobaro®

Industrial IoT Solutions

– Modbus over LoRaWAN Bridge –

User Manual

Product Name: Modbus485-LoRaWAN
Related Firmware: ≥ V0.1.0
Document Date: 2018-08-13

Contents

1. Overview

The Lobaro LoRaWAN Modbus Bridge is a low power device that can be used to read values out of a variety of other devices via Modbus (ASCII/RTU) over a RS-485 interface and forward them over LoRaWAN, so that they can be accessed from an attached system. Typical applications include reading out electric and water meters or retrieving data from environmental sensors like temperature and humidity.

The Bridge can easily be configured to read the Modbus registers required for your application with a small USB-adapter and our Lobaro Maintenance Tool, available on Windows, Mac, and Linux. It is installed on a DIN rail (DIN EN 50022) so it can be added to existing racks or put in an individual box. It can be powered via 5V DC, 100-240V AC (via a power adapter) or by two standard 1.5V AA batteries.

Please read the manual carefully before operating the device. A safe operation of the device is only possible if you follow the guides provided in this manual. Using the device differently than intended by Lobaro my cause damage to people, the environment, or the device.

1.1. The Device

Figure 1 shows the Modbus Bridge with the most important parts highlighted. The bridge is assembled from two boards that are connected by a 6 pin wire. The picture shows the standard version of the Bridge, that is powered from an external 5V DC power supply. On request a second hardware variant is available that comes without the 5V DC connector but instead has a battery clip assembled that can be fitted with one 3.6V C mono-cell (e.g. SL2770, 9Ah) for long time battery based usage.

There are two LEDs present that are used to communicate the activity of the Bridge. LED1 is a green LED next to the 'Addon' connector on the smaller board. LED2 is an RGB-LED on the bigger board next to the connector.

2. Installing the Modbus Bridge

The device can easily snapped on a top hat rail. It can therefore be added to a variety of racks alongside other devices. This also makes it convenient to install an external power supply next to the bridge that provides the 5V to operate it.

The Modbus Bridge must not be installed in a height of more than 2m.
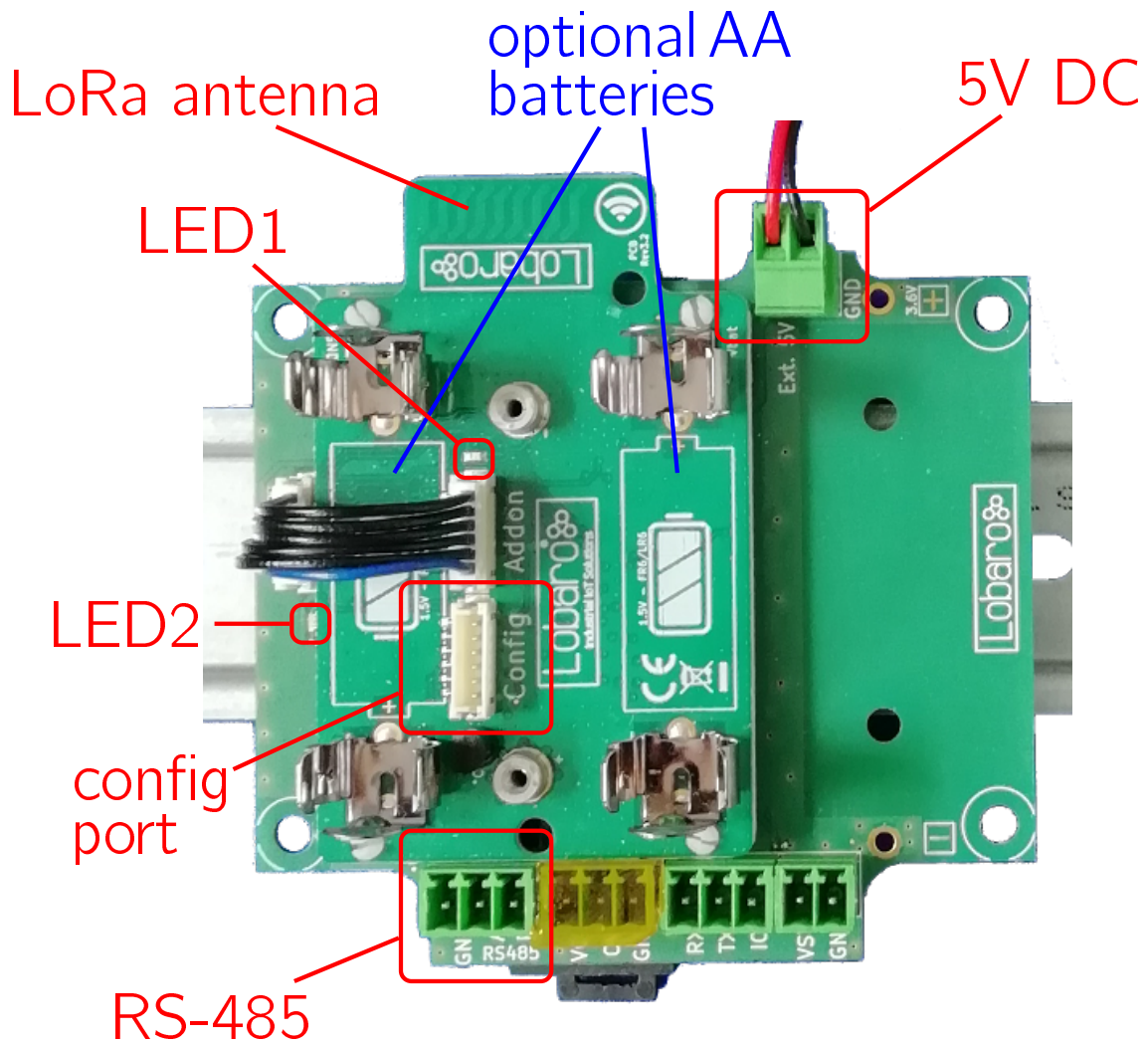
Figure 1: Device overview

2.1. Power Supply

The normal way to power the Modbus Bridge is by an external 5V DC power source at the contact marked 'Ext. 5V' (see figure 1). This can come from any reliable 5V power source in your rack. If there is no such voltage present at your installation, power converters that provide 5V DC from a 100-240V AC supply and fit on a DIN rail are widely available. Please refer to the LoRaWAN Modbus Bridge product page on our website for a recommendation. Alternatively you can insert two standard AA batteries (FR6/LR6) to power the Modbus Bridge. This batteries will only be used to power the Bridge when there is no power on the 5V DC connector. Therefore it is possible to create a backup power, so that the device keeps running through power outages. The bridge will run perfectly fine over the 5V supply without the batteries.

When using the AA batteries, make sure you use the correct batteries and get the polarity right. The device can be powered by two AA battery cells put in series, using with a nominal

voltage not exceeding 1.5V. Batteries must be inserted respecting the polarisation labelling ('+'-Symbol) printed on the device. Only batteries of the types Alkali-Manganese (1.5V, LR6), Lithium-Iron-Sulphide (1.5V, FR6), or NiMH accumulator cells (1.2V, HR6). Batteries or accumulators with a nominal voltage of more than 1.5V must not inserted into the device under any circumstances! In particular, lithium based cells wit 3.6V or 3.7V must not be used on the AA battery slots!

2.2. Modbus Connection

The Bridge can be attached to a Modbus RS-485 line. The two pins labelled 'A' and 'B' must be connected accordingly, ideally using a twisted pair wire. The connection pins are marked as 'RS-485' in figure 1. There is also a ground pin provided, marked 'GND'. This can be used to create a common ground with other devices if needed. This is normally not necessary for a successful Modbus communication.

2.3. LoRaWAN connection

Normally, the Bridge uses its internal on-board LoRa PCB antenna to transmit the data to the LoRaWAN network. No installation is required in this case. If the LoRaWAN gateway you want to reach is far away or blocked by obstacles, this antenna might not be sufficient to communicate reliably. If this is the case, the bridge can be supplied with an external antenna attached that increases communication range. Please contact Lobaro for such a modification.

An external antenna is advisable when the bridge is installed inside a metal casing with strong walls. For such setups an antenna should be installed outside the box to prevent signals from being blocked.
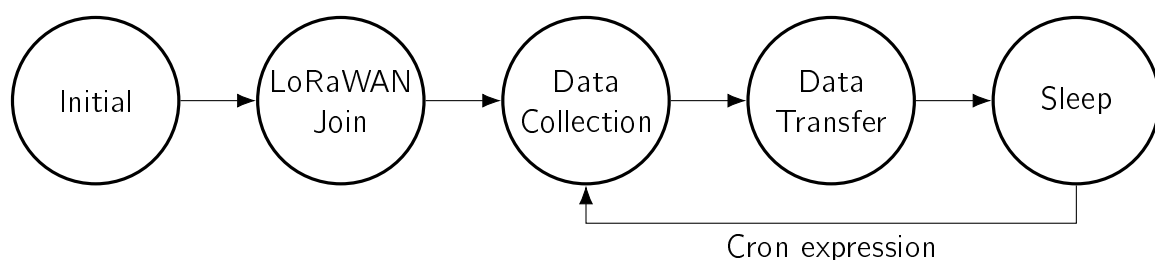
3. Work Cycle

Figure 2: The five phases of the Modbus Bridge Workflow

The Bridge has a simple work cycle that consists of five phases. It is illustrated in figure 2.

3.1. Initial Phase

This is the phase that is executed after the device is started of restarted. The Bridge performs a quick self test which you can easily spot by the RGB-LED on the lower board cycling through the colours red, green, and blue. After that, the configuration is evaluated. If successful, the LoRaWAN Join phase is executed next.

3.2. LoRaWAN Join Phase

If the Bridge is configured to use over the air activation (OTAA), the OTAA Join is performed at this point. The device will repeatedly try to join its LoRaWAN network until the process is successful. It then enters the Data Collection Phase.

If the Bridge is configured to use ABP instead of OTAA, this phase is left immediately and the Data Collection Phase is entered.

3.3. Data Collection Phase

During the Data Collection Phase is the device accesses the RS-485 Modbus and tries to read the values of all the registered listed in the configuration. The registers are accessed in the order given by the configuration and each address is requested exactly once. You can monitor the reading on the RGB-LED. It indicates the beginning of reading with a blue flash. For each register the outcome of the reading attempt is reported by a green flash for success and a red flash for failure.

After all registers have been tried to access, the Bridge enters the Data Transfer Phase.

3.4. Data Transfer Phase

During the Data Transfer Phase the Bridge uploads all register data it collected in the prior Data Collection Phase using LoRaWAN. Depending on the number of registers read this can require multiple LoRaWAN messages to be sent. Since LoRa requires any device to respect a strict duty cycle, it is possible, that the Bridge will need to wait before sending its messages. If this happens, the device will enter a power saving modus while waiting for the next message. It is possible that transferring all data will take several minutes.

In addition to the register data, the Bridge sends a status packet once a day during this phase. The status packet will always be transmitted prior to any data packets.

For a detailed description of the data sent refer to chapter 5.

3.5. Sleep Phase

After transfer of all data packets the Bridge enters the Sleep Phase. During this it is completely inactive to avoid wasting power (sleep current $< 30\mu$A). It remains sleeping until the cron expression given in the configuration triggers. When that happens, it enters the Data Collection Phase again.

4. Configuration

4.1. The Lobaro Maintenance Tool
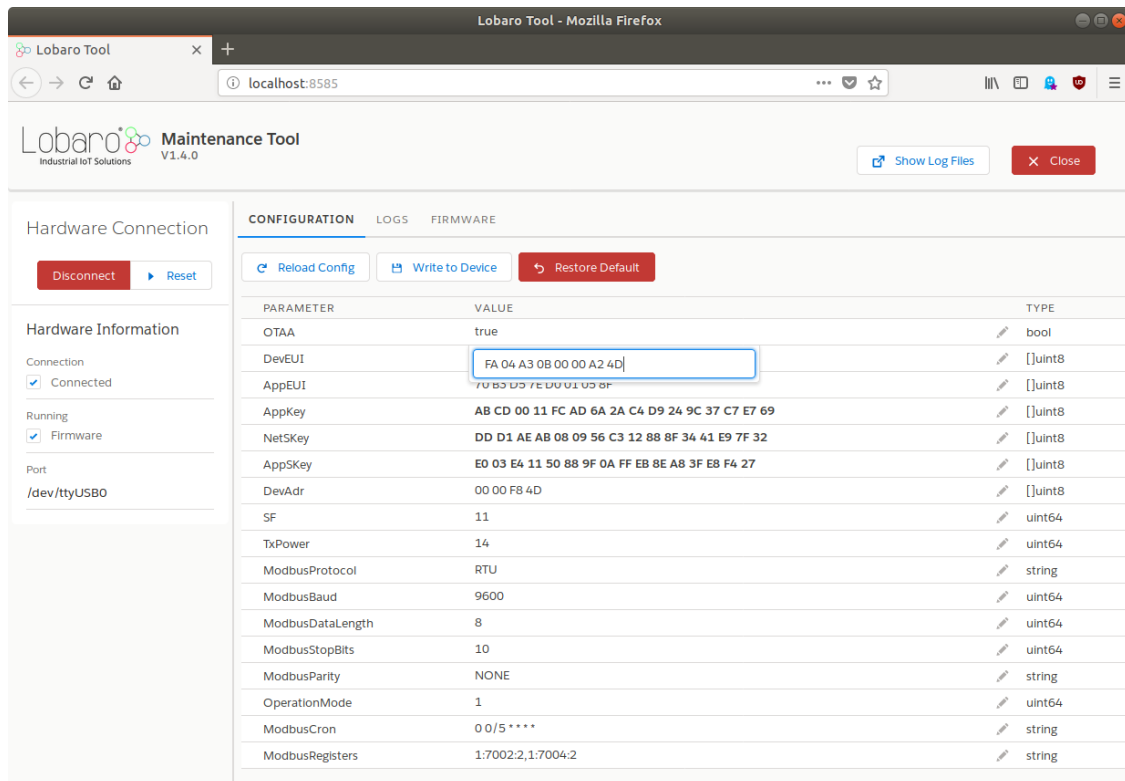
Figure 3: Lobaro maintenance tool (Windows, Linux, Mac)

The initial device configuration can be done very comfortably from your PC via the serial configuration interface. Beside the needed Lobaro USB to UART adapter the *Lobaro Maintenance Tool*[1] needs to be installed. This tool is freely available for various operating systems including Windows, Linux, Mac and Linux-ARM (e.g. Raspberry-PI) on and works with all Lobaro sensors.

Technically this software opens a webserver on port 8585 that runs in a background console window. The actual user interface can be accessed normally using a standard web browser at address `http://localhost:8585` (see fig. 3). Normally your default browser should be opened with this URL automatically after tool startup . Even remote configuration and log-observation over the Internet is possible, e.g. having a Raspberry PI via USB connected to the Lobaro device and accessing the maintenance tool from a remote machines browser over the Internet.

Additionally to the device setup the tool can also be used for firmware updates ('Firmware Tab') , watching real-time device diagnostic output ('Logs Tab') and initiating device restarts.

---

[1]Lobaro Maintenance Tool free download: `https://www.lobaro.com/lobaro-maintenance-tool/`

Please note that the device is automatically restarted each time the configuration has been changed!

4.2. Connecting the USB config adapter

For configuration and firmware updates we provide a special serial-USB adapter that can be connected as shown in figure 4. The corresponding connector on the PCB is marked with the word 'Config'.

The USB-adapter will add a virtual serial 'COM' Port to your system. Your operating system needs the *CP210x USB to UART Bridge*[2] driver installed. A download link is provided next to the 'Connect' button when you start the Maintenance Tool.

While the config adapter is connected, the device will be powered from the USB port with a regulated voltage of 3.3V. It is not necessary – although it would be no problem – having batteries inserted or a different supply connected while using the config adapter. All configuration parameters will be kept non-volatile regardless of the power supply.

Figure 4: Connected Lobaro USB configuration adapter

4.3. System parameters

After being successfully connected to the hardware using the Lobaro Maintenance Tool you can press 'Reload Config' in the 'Configuration' tab to read the current configuration from the

---

[2]https://www.silabs.com/products/development-tools/software/usb-to-uart-bridge-vcp-drivers

device. For every parameter a default value is stored non volatile inside the hardware to which you can revert using the 'Restore default' button in case anything got miss configured.

All LoRaWAN & other firmware parameters are explained in the following.

4.3.1. LoRaWAN network parameters

A large part of the configuration parameters are used to control the device's usage of LoRaWAN. Table 1 lists all of them. There are two different ways to use LoRaWAN: *over-the-air activation* (OTAA) and *activation by personalization* (ABP). Some configuration parameters are only used with one of those methods, others are used for both.

Name	Type	Used	Description
OTAA	bool	both	*true*: use over-the-air activation (OTAA) *false*: use activation by personalization (ABP)
DevEUI	bytes[8]	OTAA	the 8 byte long DevEUI is used with OTAA to identify the device on join. The default is predefined in the hardware and guarantees an ID that is unique world wide. Should not be changed unless required by the network provider. Hex format without 0x prefix.
AppEUI	bytes[8]	OTAA	ID defining the application used in the LoRaWAN network. Hex format without 0x prefix.
AppKey	bytes[16]	OTAA	Application Key as defined by the LoRaWAN network operator. This is used to encrypt communication, so keep it secret. Hex format without 0x prefix.
OTAADelay	int	OTAA	Seconds to wait for a new attempt after an unsuccessful OTAA join. The actual waiting time will be randomly increased by up to a third of that amount, in order to avoid devices repeatedly interfering with each other through bad timing. The default value is 300, which means the timeout between attempts is 300-400 seconds.
AppSKey	bytes[16]	ABP	App Session Key as defined by the LoRaWAN network operator. Hex format without 0x prefix.
NetSKey	bytes[16]	ABP	Network Session Key ad defined by the LoRaWAN network operator. Hex format without 0x prefix.
DevAdr	bytes[4]	ABP	Device Address as defined by the LoRaWAN network operator. Hex format without 0x prefix.
SF	int	both	Initial LoRa spreading factor used for transmissions. Valid range is 7-12. The actual spreading factor used might change during operation of the device if Adaptive Data Rate (ADR) is used.
TxPower	int	both	Initial transmission output power in dBm. The LoRaWAN protocol allows only specific values: 2, 5, 8, 11, 14. The actual power used might change during operation if Adaptive Data Rate (ADR) is used.
ADR	bool	both	*true*: use adaptive data rate (ADR) *false*: don't use adaptive data rate (ADR)

Table 1: LoRaWAN network parameters

4.3.2. Modbus configuration

The communication between devices over Modbus is done over a serial port (UART), that is communicating over a RS-485 line. There are a number of details that must be configured correctly for the communication to succeed. Table 2 lists the configuration parameters used for that.

Name	Type	Description	Default
ModbusProtocol	string	The kind of Modbus protocol used to communicate over RS-485. Supported values are RTU and ASCII.	RTU
ModbusBaud	int	The Baud rate used on the UART.	9600
ModbusDataLength	int	Number of bits used for data.	8
ModbusStopBits	string	Number of stop bits used in UART. Values: 0.5, 1, 1.5, 2	1
ModbusParity	string	Parity for UART. Values: EVEN, ODD, NONE	EVEN

Table 2: Modbus UART configuration parameters

4.3.3. External sensor/device configuration

The Bridge can be configured to read any Modbus register from any device accessible over the bus. The frequency of the data collection can also be defined individually using a cron expression. Table 3 lists all configuration parameters used to control data collection.

Name	Type	Description
ModbusCron	string	Cron expression controlling how often and when the device wakes up and reads registers. Chapter 4.3.4 gives a short introduction into cron expressions.
ModbusRegisters	string	Expression defining which Modbus registers to read. Chapter 4.3.5 explains the format used.
OperationMode	int	Operation mode used by the device. Use fixed value of 1.

Table 3: Metering configuration parameters

4.3.4. Cron expressions

Cron expressions are used to define specific points in time and regular repetitions of them. The schedule for data collecting phases is defined using the Cron[3] format which is very powerful format to define repeatedly occurring events[4].

Standard Lobaro devices typically do not need to know the real time for proper operation. All times are relative to the random time when batteries are inserted. If needed by the target application Lobaro can deliver on request special hardware support for keeping data acquisition intervals based on a real time clock which stays in sync with the *real* time. Please contact Lobaro directly if you need such a custom product variant.

A cron expression consists of 6 values separated by spaces:

- Seconds (0-59)

- Minutes (0-59)

- Hours (0-23)

- Days (1-31)

- Month (1-12)

- Day of Week (SUN-SAT $\widehat{=}$ [0,6])

Examples of CRON definitions:

```
0 5 * * * *        Hourly at minute 5, second 0 (at 00:05:00, 01:05:00, ...)
0 1/10 * * * *     every 10 minutes from minute 1, second 0 (minutes 1, 11, 21, ...)
0 0 6 * * *        Daily at 6:00:00
0 0 13 1,15 * *    1st and 15th day of every month at 13:00:00
0 0 9 1-5 * *      Every month daily from day 1 till 5 at 9:00:00
```

4.3.5. Register definitions

The Modbus registers that the device should read are defined over a single string configuration value. Each individual register is addressed using three numbers:

- The slave device's Modbus address

- The number of the first register to read

- The number of consecutive registers to read

---

[3]For more information about Cron see https://en.wikipedia.org/wiki/Cron
[4]Online introduction: https://github.com/lobaro/docs/wiki/CRON-Expressions

This three numbers are separated by a colon ":". For example the expression `1:10:3` would assign the Bridge to access the slave device responding to address 1 and try to read the consecutive registers numbered 10, 11, and 12. As each register holds 16 bits of data, this would result in a total 6 bytes of data.

Multiple of this register addresses can be added to the configuration string by separating them with a comma ",". No spaces or other additional characters are allowed in the register definition.

Examples of register definitions:

`1:0:3`	Read the first three registers of device 1.
`2:40001:1,2:2000:10`	Read registers 40001 and 2000 to 2009 of device 2.
`4:0:2,5:0:2,5:20:1`	Read registers 0 and 1 of device 4 as well as registers 0, 1, and register 20 of device 5.

Please be advised that the maximal number of consecutive registers the device is able to transmit over LoRaWAN is limited by the size of a single LoRaWAN message. The actual maximal size of a message is dependent on the spreading factor used and on the regulations of the country your device is operating. In the EU, the maximal message size for a spreading factor of 12 is 50 bytes (spreading factor 12 has the hardest size limitation). So if you want to make sure that your data is not dropped because of message size restrictions even on the highest spreading factor, you should limit the number of consecutive registers to 20, so that the data can fit into a single message. If you need more registers, you can split the register definition into separated parts. E.g. instead of `0:0:30` you could use `0:0:20,0:20:10` to make sure you get the data of all 30 registers.

5. LoRaWAN Data Upload Formats

After reading registers via Modbus, the Bridge starts uploading data via LoRaWAN. There exist two data formats that are transmitted over different LoRaWAN ports.

As LoRaWAN can only transmit very short messages, the message formats contain only data bytes. The meaning of a byte is determined by its position within a message. The following describes the package formats used by the Modbus Bridge.

Multi byte integers are transmitted as *big endian* (this does not apply to the values read from Modbus registers – those are transmitted as they are reported by their devices). Values that would require decimal places are transmitted in smaller units (e.g. mV instead of V). Since data packets sent over LoRa can be lost, a timestamp is added to every data packet. Timestamps are encoded as signed 40 bit big endian integers and express the number of seconds passed since 00:00:00 January 1st, 1970 (UNIX timestamp). Timestamps are according to the devices internal clock, which might be set to an incorrect value. The timestamp always indicates the begin of the corresponding measurement phase.

We provide a JavaScript reference implementation of a decoder for the data packages on GitHub[5], which can be used directly for decoding in The Things Network[6].

5.1. Status Packet

Port 1 – In order to provide some information about the health of the device itself, the Modbus Bridge sends a status update at a daily basis. The status packet is sent on the first upload phase after activation of the device (after reboot) and then repeatedly in every upload phase that takes place a day or longer after the previous status packet. It has a fixed length of 14 bytes.
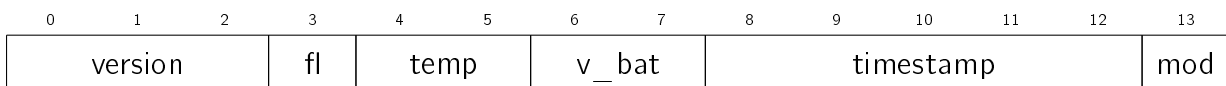
0	1	2	3	4	5	6	7	8	9	10	11	12	13
version			fl	temp		v_bat		timestamp					mod

Figure 5: Bytes, port 1 – Status Packet

name	type	description	example
version	uint8[3]	Version of the firmware running on the device	1, 0, 4 $\equiv$ v1.0.4
fl	uint8	Status flags, for internal use	0
temp	int16	Temperature measured inside the device in $\frac{1}{10}^{\circ}$C	246 $\equiv$ 24.6°C
v_bat	uint16	Voltage currently supplied by power source in mV	3547 $\equiv$ 3.547V
timestamp	int40	Internal date/time at creation of the status packet	1533055905
mod	uint8	Operation mode the device runs	1

Table 4: Fields port 1 – Status Packet

5.2. Data Packet

Port 2 – The data packet is used to transmit values successfully read from Modbus registers. It begins with a 5 byte timestamp followed by 1 or more data packs of 6 or more bytes each. The number and length of the data packs depends on the configuration of the Bridge. Figure 6 and table 5 illustrate the complete data packet as well as one individual data pack.

Each pack corresponds to a single register address configuration in your device configuration. It is prefixed by the registers address using the same numbering as in the configuration, that contains of the Modbus address (8 bit) of the slave device the value is read from, the number of the first register to be read (16 bit), and the number of consecutive registers to read (7 bit). For the last part of that address, only a 7 bit value is used. The highest bit of the 4th address byte is used to indicate an error condition. If that bit is set to 1, the attempt to read that register failed, and the data bytes are filled with the last successful read of that register

---

[5]https://github.com/lobaro/ttn-data-formats/blob/master/modbus-bridge/decoder.js

[6]The Things Network (TTN): An open source LoRaWAN network provider

see https://www.thethingsnetwork.org/

(or zero bytes, if that never happened). The address is followed by the raw bytes returned by the slave device on a successful read. Since each register contains a 16 bits value, the data consists of twice as many bytes as the number *cnt* indicates. Since there is no common definition of what the bytes in a Modbus register represent, the data is handed on exactly as it is returned from the slave device. The data type (signed vs. unsigned, byte order, etc.) is only dependent on what the accessed device defines. It lays application that receives this data over LoRaWAN is responsible to decode the bytes.
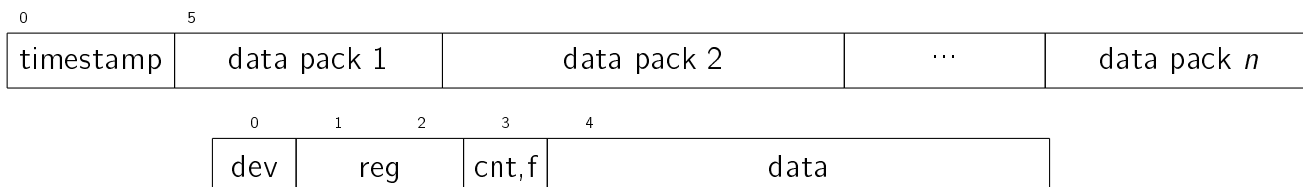
0	5			
timestamp	data pack 1	data pack 2	⋯	data pack *n*

0	1	2	3	4	
dev	reg		cnt,f	data	

Figure 6: Bytes, port 2 – Data Packet and an individual data pack

name	type	description	example
dev	uint8	Modbus address of slave device.	1
reg	uint16	Number of first register to be read.	40001
cnt	uint7	Number of consecutive registers to read.	3
f	bit	Single bit indicating failure.	0
data	uint8[2cnt]	Bytes read from registers. Two bytes per register.	00 00 01 20 f1 3a

Table 5: Fields port 2 – Bytes of a single data pack

6. Complex Setups

The Modbus Bridge as described in this manual can be individually configured to read out any registers. For some setups, this is not enough. There can be situations in which you want to read registers at different intervals, e.g. you might need some values with hourly updates and others only every other day. For measuring values with high variance it can be necessary to take multiple reads over a period of time and create an average value. Maybe you need to read a status register first and depending on its value you want to decide which registers to read and transmit the values of. You could also want to write values into registers with the Bridge.

All this scenarios are possible to solve using the Modbus Bridge, but their complexity leave the scope of our standard firmware. If you need any special processing for your Modbus setup, please contact us with your requirements, and we will make you an offer for an individual firmware that processes data the way you need.

If you find that the data rate LoRaWAN offers is a limitation for your setup, we could also provide you with a Modbus solution that uses alternate data transmission technologies, for example GSM or NarrowBand-IoT.

Find our contact information under https://www.lobaro.com/contact/, or simple send us an email to info@lobaro.com – either English or German is fine.

7. Technical characteristics

Product	
Type name	Modbus485-LoRaWAN
Description	Modbus over LoRaWAN Bridge
RF transceiver	
Type	Semtech SX1272
Frequency	863 MHz to 870 MHz
Max. TX Power	max. +14 dBm
Typical RF Range	≤2 km
Ideal RF Range	≤10 km (free line of sight)
LoRa communication	
Protocol	Class A LoRaWAN 1.0.1 EU868
Activation method	Over-the-air activation (OTAA)
	Activation by personalization (ABP)
Encryption	AES128
Modbus communication	
Bus	RS-485 twisted pair wires (with optional GND)
Protocol	RTU/ASCII
Power	
Power supply (alternatives)	5 V DC external supply
	2×AA sized battery, 1.5 V, LR6/FR6
Mechanical features	
Dimensions (board)	85 mm × 85 mm × 26 mm
Installation	Hat rail (DIN EN 50022), 35 mm
Environmental Requirements	
Operating temperature	−20 °C to 55 °C
Max installation height	2 m
Standards	

A. CE Declaration of Conformity

$C\epsilon$

Declaration of Conformity
RED directive (2014/53/EU)
RoHS directive (2011/65/EU)

We, the manufacturer

Lobaro GmbH
Tempowerkring 21d
21079 Hamburg
Germany

Lobaro®

declare on our sole responsibility that the product with following identification:

Modbus485-LoRaWAN

to which this declaration relates is in conformity with the essential requirements and other relevant requirements of the RED directive (2014/53/EU) if used for its intended purpose and in accordance with the manufacturers operating instructions. The mentioned product is also compliant to the RoHS directive (2011/65/EU).

The following norms, standards, or documents have been applied:

Health and safety (RED Art. 3(1)(a)):
      IEC 62368-1:2014 (Ed. 2) + Cor.:2015
      IEC 62479:2010
EMC (RED Art.3(1)(b)):
      ETSI EN 301 489-1 V1.9.2 (V2.2.0)[†]
      ETSI EN 301 489-3 V1.6.1 (V3.2.0)[†]
      IEC 61000-6-2:2005
Radio spectrum (RED Art. 3(2)):
      ETSI EN 300 220-1 V3.1.1 (2017-02)
      ETSI EN 300 220-2 V3.1.1 (2017-02)

[†]Referenced newer standard versions in parenthesis will supersede its predecessors. At time of testing the newer standards were not harmonized yet. All tests were carried out successfully in accordance to the already harmonized standards as well as to the superseding ones.

Lobaro®
Industrial IoT Solutions
Tempowerkring 21d - 21079 Hamburg
info@lobaro.com +49 40 22816531-0

Hamburg, 27.07.2018
Place and date of issue

Dipl.–Ing. Theodor Rohde
Managing Director Lobaro GmbH

B. Disposal

This chapter informs you on our policy in respect to EU 'Waste electrical and electronic equipment Directive' 2002/96/EC, implemented in German law by the 'Gesetz über das Inverkehrbringen, die Rücknahme und die umweltverträgliche Entsorgung von Elektro- und Elektronikgeräten (Elektro- und Elektronikgerätegesetz – ElektroG)' The following statements about the disposal of retired Lobaro devices are binding for customers in Germany. Customers in other countries should contact there local authorities in order to acquire equivalent information for their respective country.

It is a major goal of EU directive 2002/96/EC reduce the amount of retired electronic devices disposed in household waste, but have them gathered in special facilities instead for recycling and orderly disposal. Electronic devices often contain hazardous substances which are harmful to the environment as well as to human health. Under no circumstances should they be disposed in household trash.

B.1. WEEE

The WEEE-directive is the EU-directive 'Waste of Electrical and Electronic Equipment' (2002/96/EC). It aims to reduce the growing amount of electrical and electronic waste created by disposed electronic devices. The goal is to avoid and reduce disposal, and to make the disposal more friendly for the environment, by including the manufacturer in a broader responsibility.

B.1.1. Proper disposal of Lobaro B2B devices

Lobaro develops and manufactures devices for professional use. Because of this, our products have been recognised by the responsible agency 'Stiftung EAR' (Elektro-Altgeräte Register) as B2B (business to business) products. Lobaro GmbH is registered under DE18824018.

Classification as B2B implies that Lobaro is not required to pick up disposed electronic devices from local recycling points, as the law 'ElektroG' dictates manufacturers of consumer electronics to do.

Consequently, our products may not be disposed in this way.

Instead, we offer our customers to take back retired Lobaro devices and dispose them properly free of charge. Devices to be disposed must be sent to Lobaro GmbH postpaid and marked with the words 'ZUR ENTSORGUNG'.

B.2. RoHS

As a manufacturer of industrial devices of information and telecommunication technology we are producing conforming to RoHS as specified in the EU-directive 2011/65/EU.

B.3. Batteries

Conforming to the German law about the disposal of used batteries (§ 12 Satz 1 Nr. 1-3 BattV (Verordnung über die Rücknahme und Entsorgung gebrauchter Batterien und Akkumulatoren (BattVO) vom 27. März 1998 (BGBl. I S. 658))), we as the manufacturer advise you, that the customer and end-consumer is required by law to return used batteries for a proper disposal. After usage, batteries can be returned to the vendor or at a place close proximity free of charge. If used batteries are sent to the vendor, parcels must be sent fully postpaid.

The following symbol indicates, that a battery contains heavy metal or other hazardous substances, and that it must not be disposed in common household or industrial waste. The abbreviations under the symbol stand for: 'Cd' (Cadmium), 'Li' (Lithium) / 'Li-Ion' (Lithium-Ion), 'Ni' (Nickel), 'Mh' (Metal Hydride), 'Pb' (Lead), 'Zi' (Zinc).
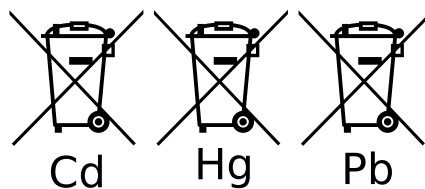
C. Entsorgung (German)

Dieses Kapitel informiert Sie über das Vorgehen der Lobaro GmbH in Bezug auf die EU Richtlinie 2002/96/EC über Elektro- und Elektronikaltgeräte (engl. WEEE), umgesetzt in deutsches Recht durch das „Gesetz über das Inverkehrbringen, die Rücknahme und die umweltverträgliche Entsorgung von Elektro- und Elektronikgeräten (Elektro- und Elektronikgerätegesetz – ElektroG)". Die nachfolgend gemachten Aussagen über die Entsorgung von außer Gebrauch genommenen Lobaro Geräten gelten verbindlich nur für Kunden innerhalb Deutschlands. Kunden in anderen Ländern sollten ihre örtlichen Behörden kontaktieren, um entsprechende Informationen zu erhalten.

Ein vordringliches Ziel der EU Richtlinie 2002/96/EC ist es, die Entsorgung von Elektro-/Elektronikaltgeräten zusammen mit dem unsortierten Hausmüll zu minimieren und einen hohen Grad an getrennter Sammlung und Verwertung von Elektro-/Elektronikaltgeräten zu erreichen. Elektronische Geräte können Stoffe enthalten, die schädlich für die Umwelt und die menschliche Gesundheit sind und auf keinen Fall zusammen mit dem Hausmüll entsorgt werden sollten.

C.1. WEEE

Die WEEE-Richtlinie (von engl.: Waste of Electrical and Electronic Equipment; deutsch: Elektro- und Elektronikgeräte-Abfall) ist die EU-Richtlinie 2002/96/EC zur Reduktion der zunehmenden Menge an Elektronikschrott aus nicht mehr benutzten Elektro- und Elektronikgeräten. Ziel ist das Vermeiden, Verringern sowie umweltverträgliche Entsorgen der zunehmenden Mengen an Elektronikschrott durch eine erweiterte Herstellerverantwortung.

C.1.1. Ordnungsgemäße Entsorgung von B2B Lobaro Geräten

Lobaro Geräte werden für den professionellen Gebrauch entwickelt und vermarktet. Dementsprechend sind sie von der zuständigen deutschen Agentur Stiftung EAR (Elektro- Altgeräte Register) als B2B (business to business) Produkte anerkannt worden. Die Lobaro GmbH ist dort unter der Nr. DE18824018 registriert.

Die B2B Klassifizierung bedeutet, dass Lobaro nicht zur Abholung von Elektro-/Elektronikaltgeräten bei den kommunalen Sammelstellen verpflichtet ist, wie es das ElektroG den Produzenten von für den Konsumenten-Markt bestimmten Elektro-/Elektronikgeräten vorschreibt.

Folglich dürfen unsere Produkte auch nicht auf diesem Wege entsorgt werden.

Stattdessen bieten wir unseren Kunden an, nicht mehr gebrauchte Lobaro Geräte zurückzunehmen und auf unsere Kosten ordnungsgemäß zu entsorgen.

Geräte zur Entsorgung müssen mit bezahltem Porto an die Lobaro GmbH eingesendet werden und mit der Kennzeichnung „ZUR ENTSORGUNG" versehen sein.

C.2. RoHS

Als Hersteller von industriellen Geräten der Informations- und Telekommunikationstechnik produzieren wir RoHS konform, gemäß den Inhalten der EU-Richtlinie 2011/65/EU.

C.3. Batteriegesetz

Gemäß § 12 Satz 1 Nr. 1-3 BattV (Verordnung über die Rücknahme und Entsorgung gebrauchter Batterien und Akkumulatoren (BattVO) vom 27. März 1998 (BGBl. I S. 658)) weisen wir als Verkäufer darauf hin, dass der Kunde als Endverbraucher zur Rückgabe gebrauchter Batterien gesetzlich verpflichtet ist. Batterien können nach Gebrauch beim Verkäufer oder in dessen unmittelbarer Nähe unentgeltlich zurückgegeben werden. Werden Batterien an den Verkäufer übersandt, ist das Paket ausreichend zu frankieren.

Das nachfolgende Symbol bedeutet, dass es sich um schwermetallhaltige, schadstoffhaltige Batterien handelt, die nicht mit dem einfachen Haus- oder Gewerbeabfall entsorgt werden dürfen. Die unter dem Symbol befindlichen Abkürzungen bedeuten: „Cd" (Cadmium), „Li" (Lithium) / „Li-Ion" (Lithium-Ionen), „Ni" (Nickel), „Mh" (Metallhydrid), „Pb" (Blei), „Zi" (Zink).
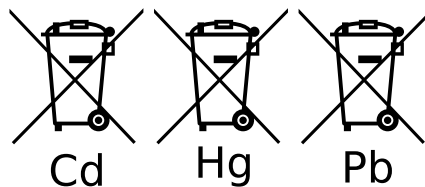
D. Document Versions

Changes to this document:

- *2018-08-16* – Initial public version.