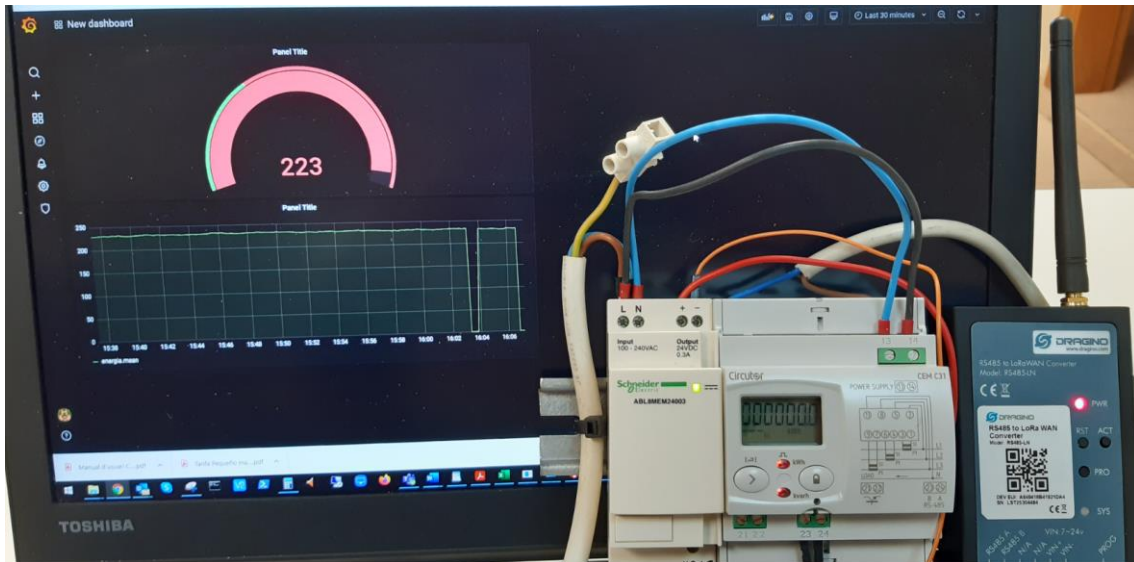


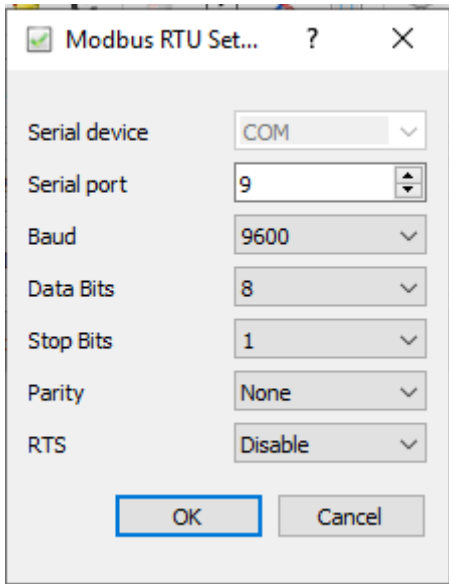
ELECTRICAL CABINET ENERGY READINGS WITH CIRCUTOR CEM C31 485-T1-MID





VOLTAGE PHASE 1

Configure QModMaster as follows



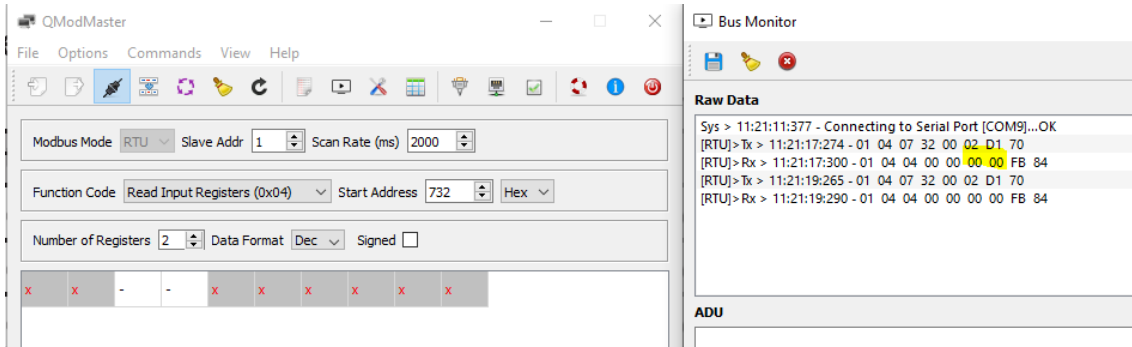
Slave unit is 1 as default

7.2.3.3.- Instantaneous values

The **Read** function is implemented for these variables.

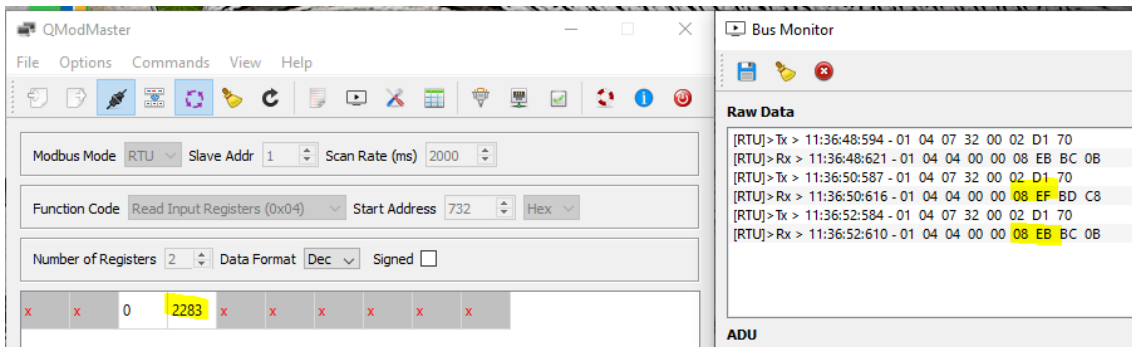
Table 19: Modbus variables: Instantaneous values.

Description	Address	Size	Units
Phase 1 voltage	0x0732	32 bits	V (1 primary decimal place)

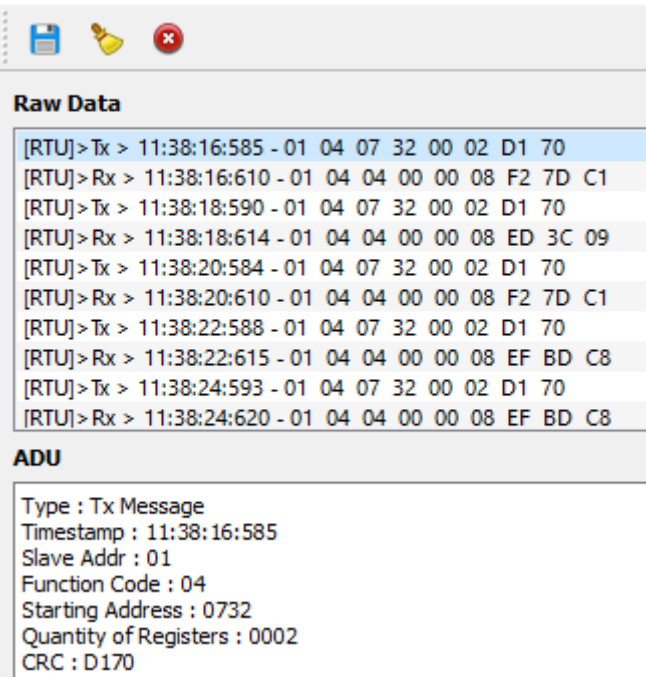


If we apply 230V on phase 1

Then we get 08EE or 2283 which is 228,3V so the value we get from Modbus has to be divided by 10

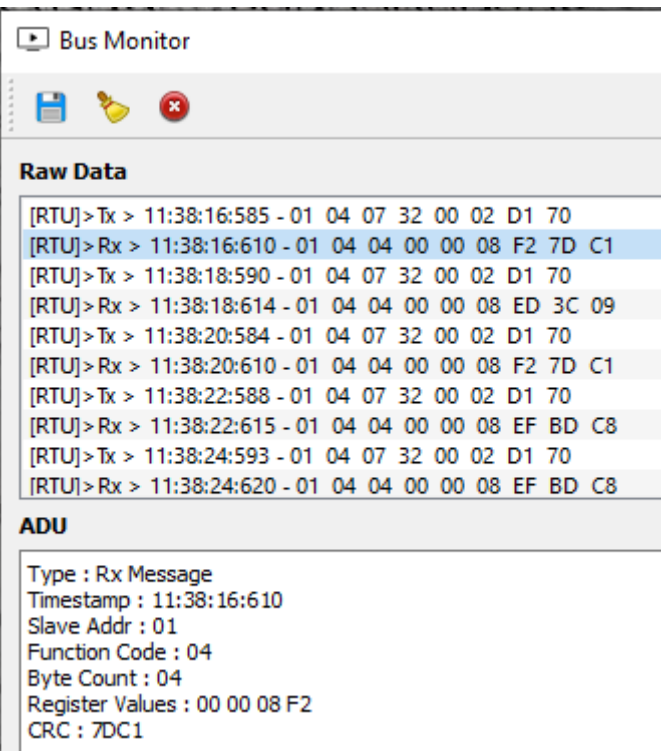


Bus Monitor



The screenshot shows a 'Bus Monitor' window with a toolbar containing icons for save, bell, and close. Below the toolbar is a 'Raw Data' section with a list of 10 RTU messages. The first message is highlighted in blue. Below the raw data is an 'ADU' section with the following details:

```
Type : Tx Message
Timestamp : 11:38:16:585
Slave Addr : 01
Function Code : 04
Starting Address : 0732
Quantity of Registers : 0002
CRC : D170
```



The screenshot shows a 'Bus Monitor' window with a toolbar containing icons for save, bell, and close. Below the toolbar is a 'Raw Data' section with a list of 10 RTU messages. The second message is highlighted in blue. Below the raw data is an 'ADU' section with the following details:

```
Type : Rx Message
Timestamp : 11:38:16:610
Slave Addr : 01
Function Code : 04
Byte Count : 04
Register Values : 00 00 08 F2
CRC : 7DC1
```

And the AT values are

AT+COMMAND1=01 04 07 32 00 02,1

AT+DATA CUT1=9,1,6~7

Let's connect Termitte to configure Dragino

Use this settings

Serial port settings

Port configuration

Port: COM27

Baud rate: 9600

Data bits: 8

Stop bits: 1

Parity: none

Flow control: none

Forward: none

Transmitted text

Append nothing

Append CR

Append LF

Append CR-LF

Local echo

Received text

Polling: 100 ms

Max. lines:

Font: default

Word wrap

Options

Stay on top

Quit on Escape

Autocomplete edit line

Keep history

Close port when inactive

Plug-ins

User interface language: English (en)

Cancel OK

Ensure we have the right comm parameters

Termit 3.4 (by CompuPhase)

COM27 9600 bps, 8N1, no handshake

AT+BAUDR=?

9600

OK

Termit 3.4 (by CompuPhase)

COM27 9600 bps, 8N1, no handshake

AT+PARITY=?

0

OK

So let's program the Dragino for reading voltage Phase 1

```
AT+COMMAND1=01 04 07 32 00 02,1
```

```
AT+DATA CUT1=9,1,6+7
```

But you have to do this when there is a Gateway on the surrounding, and the node is connected to the loraWAN network

```
AT+COMMAND1=01 04 07 32 00 02,1
```

```
OK
```

```
AT+DATACUT1=9,1,6~7
```

```
OK
```

We reset the device by pressing RST button

```
Join Accept:
```

```
DevAddr:26 01 55 84
```

```
Px1DrOffset:0
```

```
Px2Datarate:3
```

```
ReceiveDelay1:1000 ms
```

```
ReceiveDelay2:2000 ms
```

```
CMD1 = 01 04 07 32 00 02 d1 70
```

```
RETURN1 = 00 00 00 00 00 00 00 00 00
```

```
Payload = 01 00
```

let's check the transmission time

```
AT+TDC=?
```

```
600000
```

```
OK
```

This is a transmission every 60 seconds

Let's change to every 10 seconds

```
AT+TDC=10000
```

```
OK
```

And let's make a reset with button RST

We have it every 10 seconds

APPLICATION DATA

Filters

uplink	downlink	activation	ack	error
--------	----------	------------	-----	-------

time	counter	port	
▲ 12:19:09	3	2	payload: 01 00
▲ 12:18:59	2	2	payload: 01 00
▲ 12:18:49	1	2	payload: 01 00

The response is 0 since we do not have any slave listening.

Let's connect slave 1

```
CMD1 = 01 04 07 32 00 02 d1 70
RETURN1 = 01 04 04 00 00 08 eb bc 0b
Payload = 01 08
```

Voilà, but we see that the payload is not correct

So the right command is

```
AT+DATA CUT1=9,1,6+7
OK
```

AT+DATA CUT1=9,1,6+7

voilà

```
CMD1 = 01 04 07 32 00 02 d1 70
RETURN1 = 01 04 04 00 00 08 e7 bc 0e
Payload = 01 08 e7
```

Now we want to decode this payload

Payload Format

The payload format sent by your devices

Custom

decoder

converter

validator

encoder

```
1 function Decoder(bytes, port) {  
2   // Decode an uplink message from a buffer  
3   // (array) of bytes to an object of fields.  
4   var decoded = {};  
5   if (port === 2) {  
6     decoded.voltage_Phase1 = (bytes[1]*256+bytes[2])/10;  
7   }  
8 }  
9  
10 return decoded;  
11 }
```

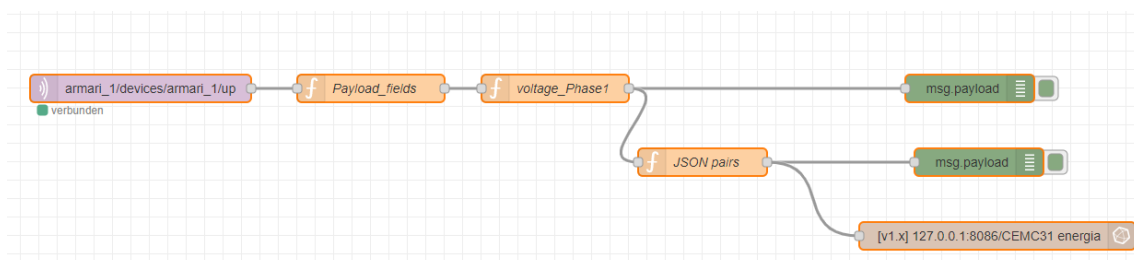
Voilà

APPLICATION DATA

Filters uplink downlink activation ack error

time	counter	port	dev id: armari_1	payload: 01 08 F7	voltage_Phase1: 229.5
▲ 12:37:19	113	2			

And the Node RED Flow to inject the data on a database



You can get the code here

<https://github.com/xavierflorensa/CIRCUTOR-CEM-C31-RS485-to-LoRaWAN>