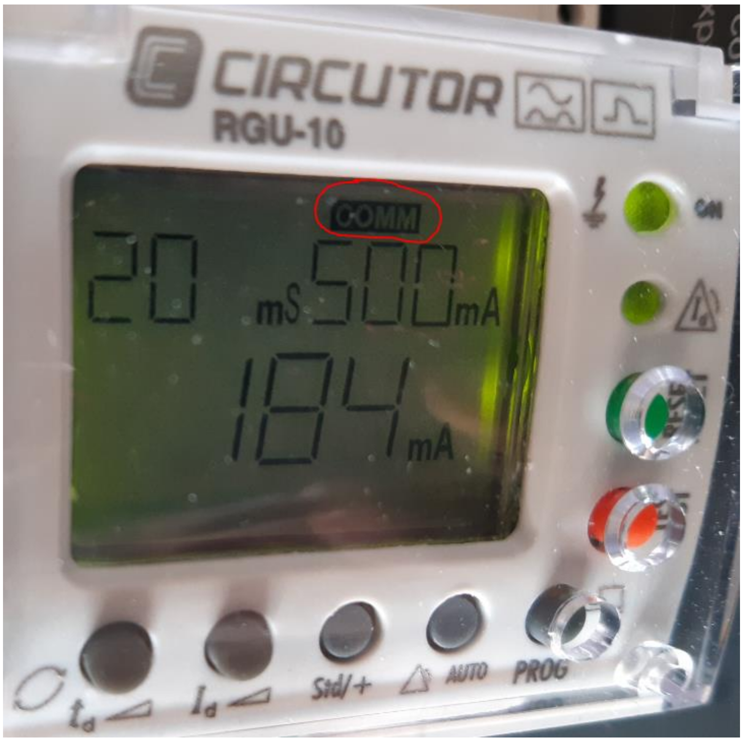
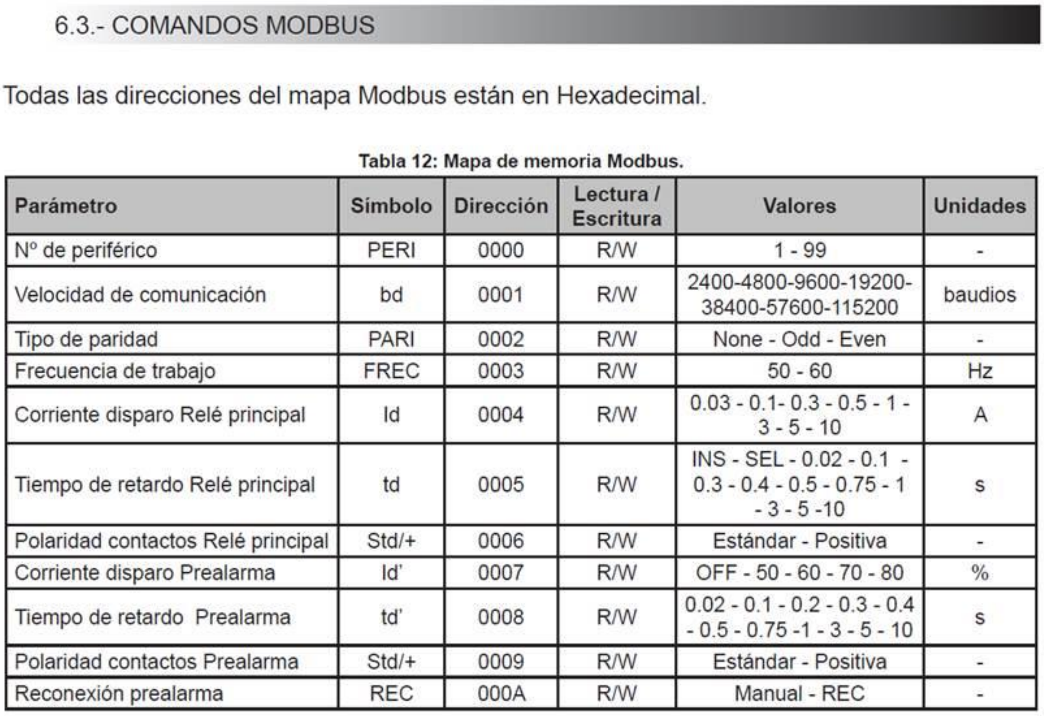
Dragino RS485 to LoRa test

Let’s test with this Modbus device

RGU10C Circutor





6.3.- COMANDOS MODBUS

Todas las direcciones del mapa Modbus están en Hexadecimal.

Tabla 12: Mapa de memoria Modbus.

Parámetro	Símbolo	Dirección	Lectura / Escritura	Valores	Unidades
Nº de periférico	PERI	0000	R/W	1 - 99	-
Velocidad de comunicación	bd	0001	R/W	2400-4800-9600-19200-38400-57600-115200	baudios
Tipo de paridad	PARI	0002	R/W	None - Odd - Even	-
Frecuencia de trabajo	FREC	0003	R/W	50 - 60	Hz
Corriente disparo Relé principal	Id	0004	R/W	0.03 - 0.1- 0.3 - 0.5 - 1 - 3 - 5 - 10	A
Tiempo de retardo Relé principal	td	0005	R/W	INS - SEL - 0.02 - 0.1 - 0.3 - 0.4 - 0.5 - 0.75 - 1 - 3 - 5 -10	s
Polaridad contactos Relé principal	Std/+	0006	R/W	Estándar - Positiva	-
Corriente disparo Prealarma	Id'	0007	R/W	OFF - 50 - 60 - 70 - 80	%
Tiempo de retardo Prealarma	td'	0008	R/W	0.02 - 0.1 - 0.2 - 0.3 - 0.4 - 0.5 - 0.75 -1 - 3 - 5 - 10	s
Polaridad contactos Prealarma	Std/+	0009	R/W	Estándar - Positiva	-
Reconexión prealarma	REC	000A	R/W	Manual - REC	-

Let’s read frequency on register 3

04 means read input registers

0003 is the address we read: Frequency

0002 is the byte number (2)

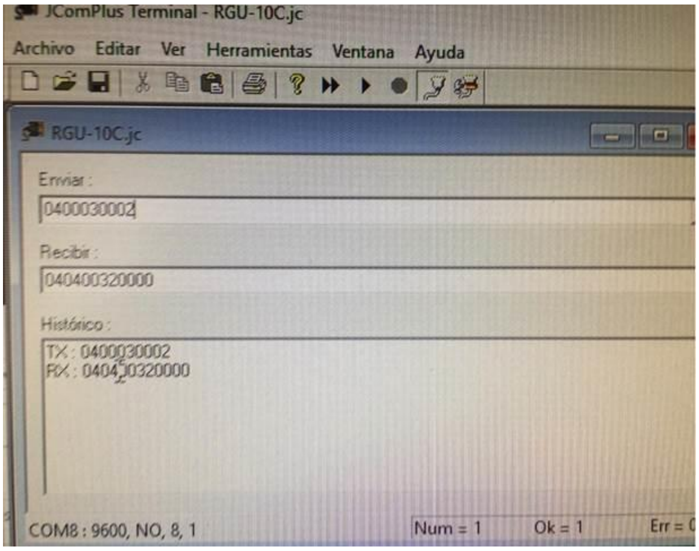
And this is the answer

04 means read input registers

04 is the returned bytes amount, yes 00 32 00 00 (4 bytes)

0032 is 50 in decimal, so the working frequency

With Jcom


JComPlus Terminal - RGU-10C.jc
Archivo Editar Ver Herramientas Ventana Ayuda
RGU-10C.jc

Enviar : 0400030002

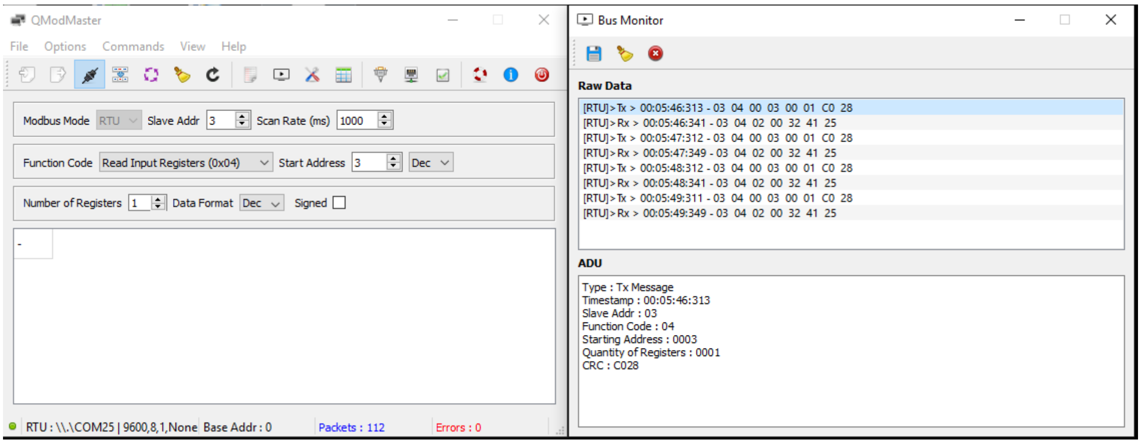
Recibir : 040400320000

Histórico :

```
TX : 0400030002
RX : 040400320000
```

COM8 : 9600, NO, 8, 1   Num = 1   Ok = 1   Err = (

And in more Depth including CRC



Modbus Mode RTU   Slave Addr 3   Scan Rate (ms) 1000

Function Code Read Input Registers (0x04)   Start Address 3   Dec

Number of Registers 1   Data Format Dec   Signed □

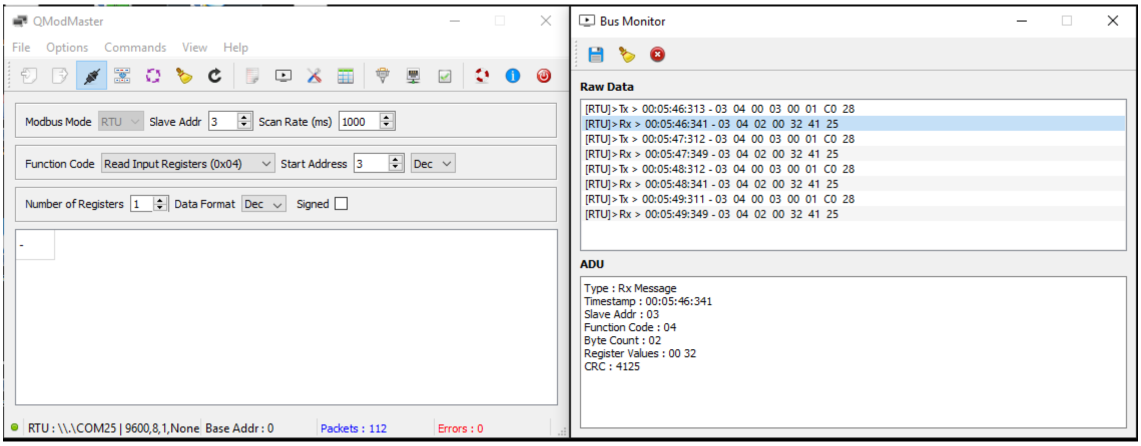
RTU : \\.\COM25 | 9600,8,1,None Base Addr : 0   Packets : 112   Errors : 0

Raw Data

```
[RTU]>Tx > 00:05:46:313 - 03  04  00  03  00  01  C0  28
[RTU]>Rx > 00:05:46:341 - 03  04  02  00  32  41  25
[RTU]>Tx > 00:05:47:312 - 03  04  00  03  00  01  C0  28
[RTU]>Rx > 00:05:47:349 - 03  04  02  00  32  41  25
[RTU]>Tx > 00:05:48:312 - 03  04  00  03  00  01  C0  28
[RTU]>Rx > 00:05:48:341 - 03  04  02  00  32  41  25
[RTU]>Tx > 00:05:49:311 - 03  04  00  03  00  01  C0  28
[RTU]>Rx > 00:05:49:349 - 03  04  02  00  32  41  25
```

ADU

```
Type : Tx Message
Timestamp : 00:05:46:313
Slave Addr : 03
Function Code : 04
Starting Address : 0003
Quantity of Registers : 0001
CRC : C028
```



Modbus Mode RTU Slave Addr 3 Scan Rate (ms) 1000

Function Code Read Input Registers (0x04) Start Address 3 Dec

Number of Registers 1 Data Format Dec Signed □

RTU : \\.\COM25 | 9600,8,1,None Base Addr : 0 Packets : 112 Errors : 0

```
[RTU]>Tx > 00:05:46:313 - 03  04  00  03  00  01  C0  28
[RTU]>Rx > 00:05:46:341 - 03  04  02  00  32  41  25
[RTU]>Tx > 00:05:47:312 - 03  04  00  03  00  01  C0  28
[RTU]>Rx > 00:05:47:349 - 03  04  02  00  32  41  25
[RTU]>Tx > 00:05:48:312 - 03  04  00  03  00  01  C0  28
[RTU]>Rx > 00:05:48:341 - 03  04  02  00  32  41  25
[RTU]>Tx > 00:05:49:311 - 03  04  00  03  00  01  C0  28
[RTU]>Rx > 00:05:49:349 - 03  04  02  00  32  41  25
```

```
Type : Rx Message
Timestamp : 00:05:46:341
Slave Addr : 03
Function Code : 04
Byte Count : 02
Register Values : 00 32
CRC : 4125
```

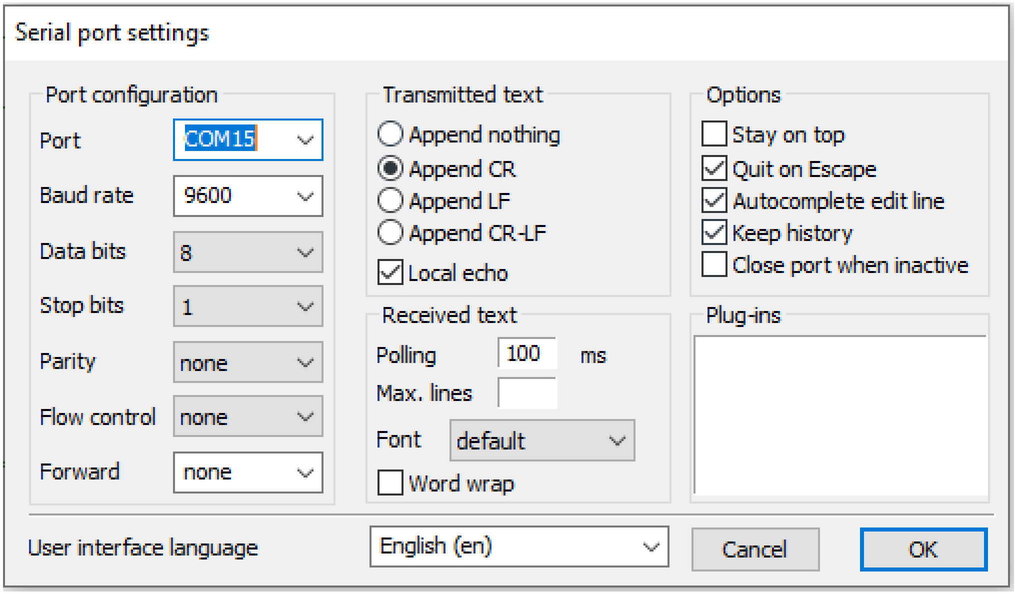
```
Sys > 22:39:57:938 - Connecting to Serial Port [\\.\COM25]...OK

[RTU]>Tx > 22:40:05:301 - 03  04  00  03  00  01  C0  28

[RTU]>Rx > 22:40:05:326 - 03  04  02  00  32  41  25

[RTU]>Tx > 22:40:06:303 - 03  04  00  03  00  01  C0  28

[RTU]>Rx > 22:40:06:334 - 03  04  02  00  32  41  25

[RTU]>Tx > 22:40:07:304 - 03  04  00  03  00  01  C0  28

[RTU]>Rx > 22:40:07:342 - 03  04  02  00  32  41  25

[RTU]>Tx > 22:40:08:302 - 03  04  00  03  00  01  C0  28

[RTU]>Rx > 22:40:08:333 - 03  04  02  00  32  41  25
```

Now we try to enter this telegram on the Draguino RS485 to send the answer to TTN

Let'see the response from the Dragino device

Let's connect the programming USB to TTL. This i sport 15

Let's open Termite terminal



Serial port settings

Port configuration

- Port: COM15
- 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

Options

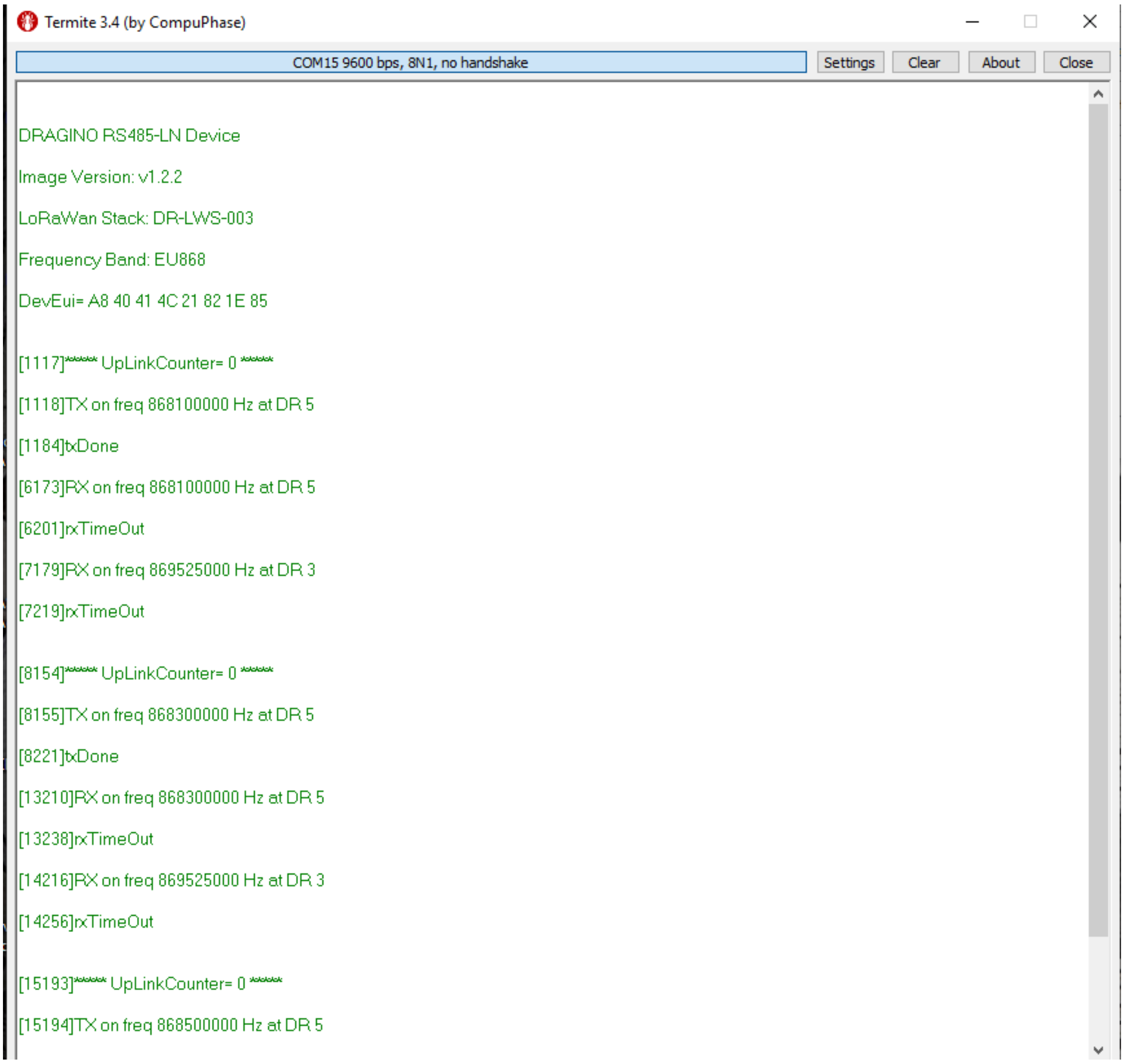
- □ Stay on top
- ☑ Quit on Escape
- ☑ Autocomplete edit line
- ☑ Keep history
- □ Close port when inactive

Received text

- Polling: 100 ms
- Max. lines:
- Font: default
- □ Word wrap

Plug-ins

User interface language: English (en)

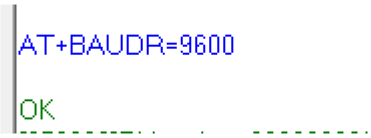


Termite 3.4 (by CompuPhase)

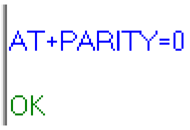
COM15 9600 bps, 8N1, no handshake

```
DRAGINO RS485-LN Device
Image Version: v1.2.2
LoRaWan Stack: DR-LWS-003
Frequency Band: EU868
DevEui= A8 40 41 4C 21 82 1E 85

[1117]***** UpLinkCounter= 0 *****
[1118]TX on freq 868100000 Hz at DR 5
[1184]txDone
[6173]RX on freq 868100000 Hz at DR 5
[6201]rxTimeOut
[7179]RX on freq 869525000 Hz at DR 3
[7219]rxTimeOut

[8154]***** UpLinkCounter= 0 *****
[8155]TX on freq 868300000 Hz at DR 5
[8221]txDone
[13210]RX on freq 868300000 Hz at DR 5
[13238]rxTimeOut
[14216]RX on freq 869525000 Hz at DR 3
[14256]rxTimeOut

[15193]***** UpLinkCounter= 0 *****
[15194]TX on freq 868500000 Hz at DR 5
```

Now let’s try to send some AT commands



```
AT+BAUDR=9600
OK
```



```
AT+PARITY=0
OK
```

Yeah, it is alive!

Let’s connect our RGU10C to the RS485-LN

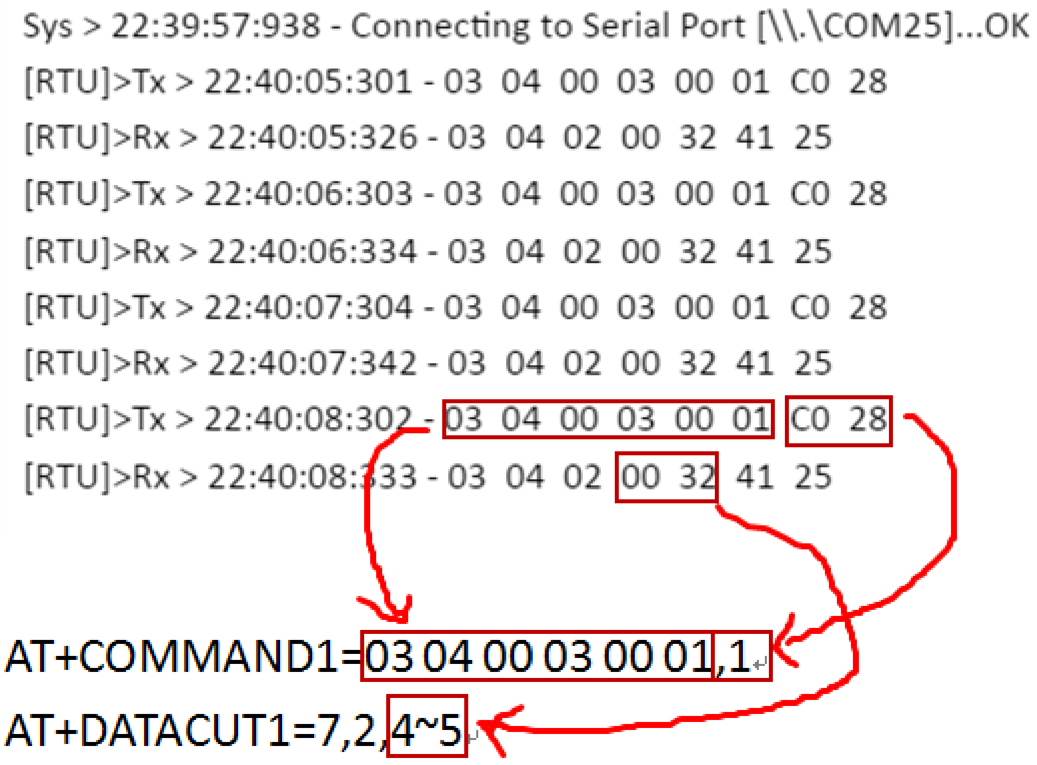
The AT+BAUDR command can set the baud rate;

The AT+PARITY command can set the data verification method;

The example given by your email, the corresponding configuration of the device:

AT+COMMAND1=03 04 00 03 00 01,1

AT+DATACUT1=7,2,4~5



```
Sys > 22:39:57:938 - Connecting to Serial Port [\\.\COM25]...OK
[RTU]>Tx > 22:40:05:301 - 03  04  00  03  00  01  C0  28
[RTU]>Rx > 22:40:05:326 - 03  04  02  00  32  41  25
[RTU]>Tx > 22:40:06:303 - 03  04  00  03  00  01  C0  28
[RTU]>Rx > 22:40:06:334 - 03  04  02  00  32  41  25
[RTU]>Tx > 22:40:07:304 - 03  04  00  03  00  01  C0  28
[RTU]>Rx > 22:40:07:342 - 03  04  02  00  32  41  25
[RTU]>Tx > 22:40:08:302 - 03  04  00  03  00  01  C0  28
[RTU]>Rx > 22:40:08:333 - 03  04  02  00  32  41  25
```

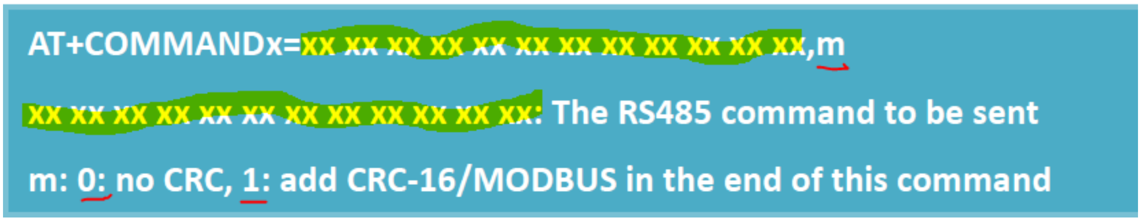
```
AT+COMMAND1=03 04 00 03 00 01,1
AT+DATACUT1=7,2,4~5
```

Give the commands to program the transactions

The example given by your email, the corresponding configuration of the device:

AT+COMMAND1=03 04 00 03 00 01,1

AT+COMMAND1

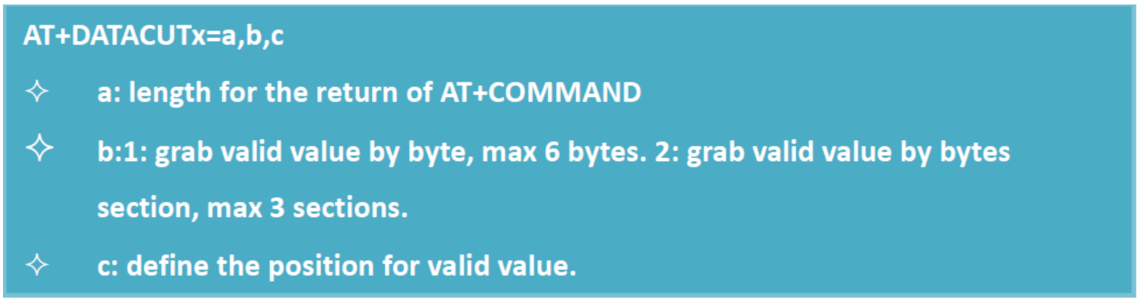


AT+COMMANDx=xx xx xx xx xx xx xx xx xx xx xx xx,m

xx xx xx xx xx xx xx xx xx xx xx xx: The RS485 command to be sent

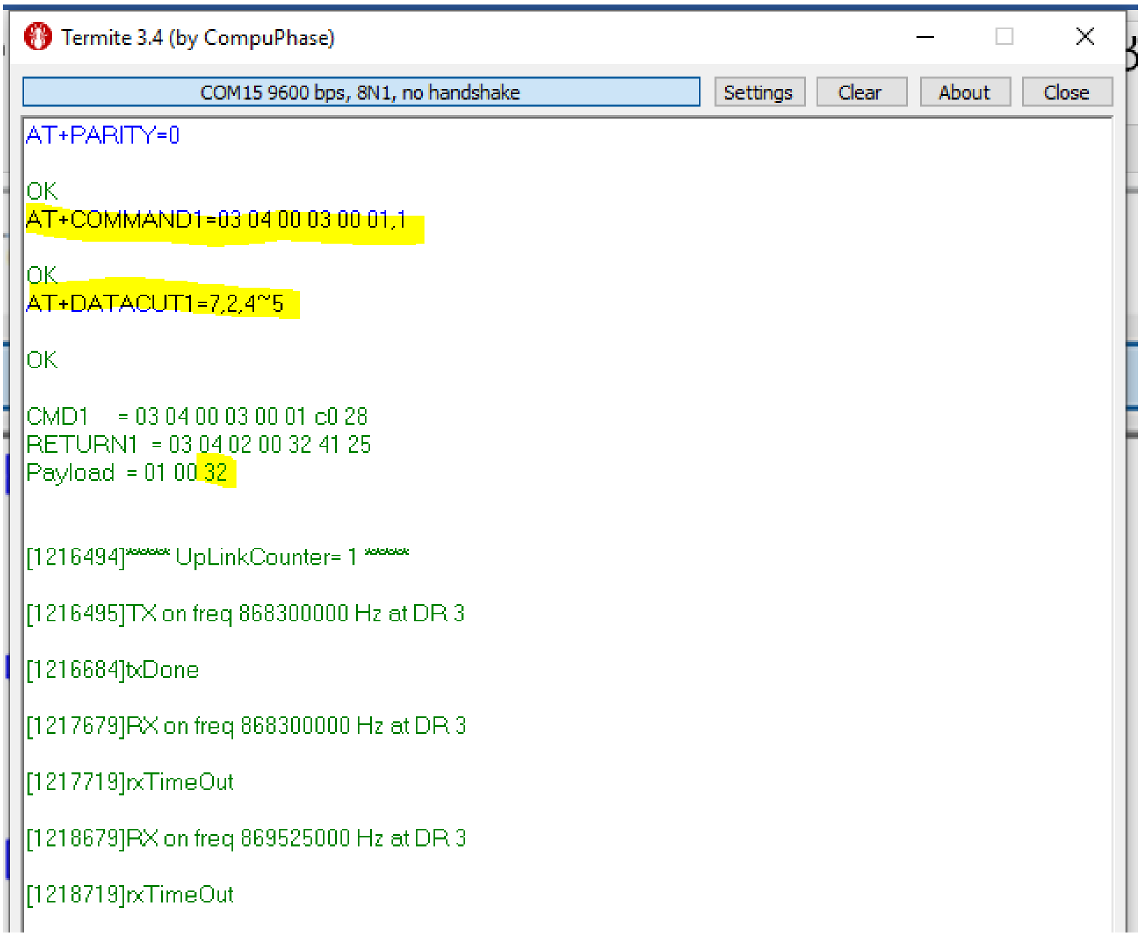
m: 0: no CRC, 1: add CRC-16/MODBUS in the end of this command

AT+DATACUT1=7,2,4~5



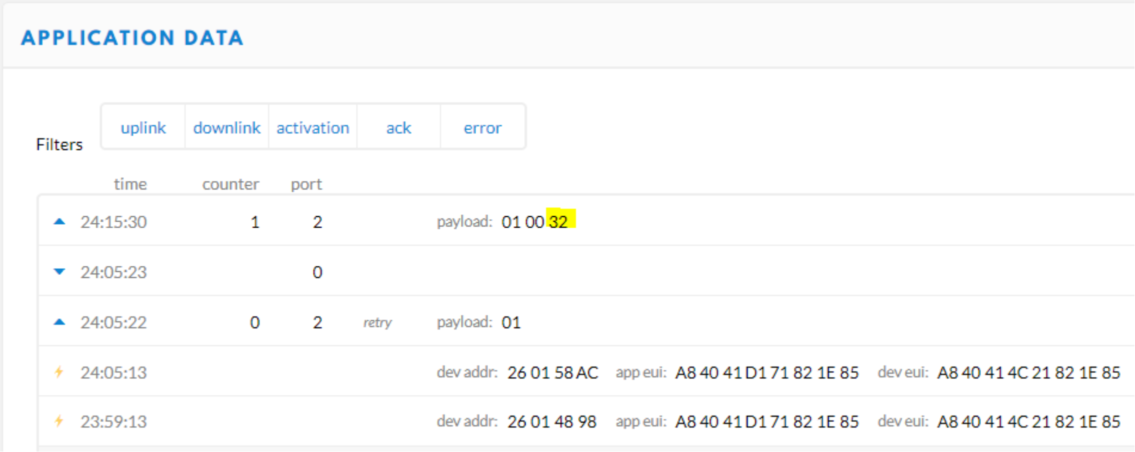
AT+DATACUTx=a,b,c

- **a: length for the return of AT+COMMAND**
- **b:1: grab valid value by byte, max 6 bytes. 2: grab valid value by bytes section, max 3 sections.**
- **c: define the position for valid value.**


Termite 3.4 (by CompuPhase)
COM15 9600 bps, 8N1, no handshake
Settings
Clear
About
Close

```
AT+PARITY=0

OK
AT+COMMAND1=03 04 00 03 00 01,1

OK
AT+DATACUT1=7,2,4~5

OK

CMD1     = 03 04 00 03 00 01 c0 28
RETURN1  = 03 04 02 00 32 41 25
Payload  = 01 00 32


[1216494]***** UpLinkCounter= 1 *****

[1216495]TX on freq 868300000 Hz at DR 3

[1216684]txDone

[1217679]RX on freq 868300000 Hz at DR 3

[1217719]rxTimeOut

[1218679]RX on freq 869525000 Hz at DR 3

[1218719]rxTimeOut
```

It Works!

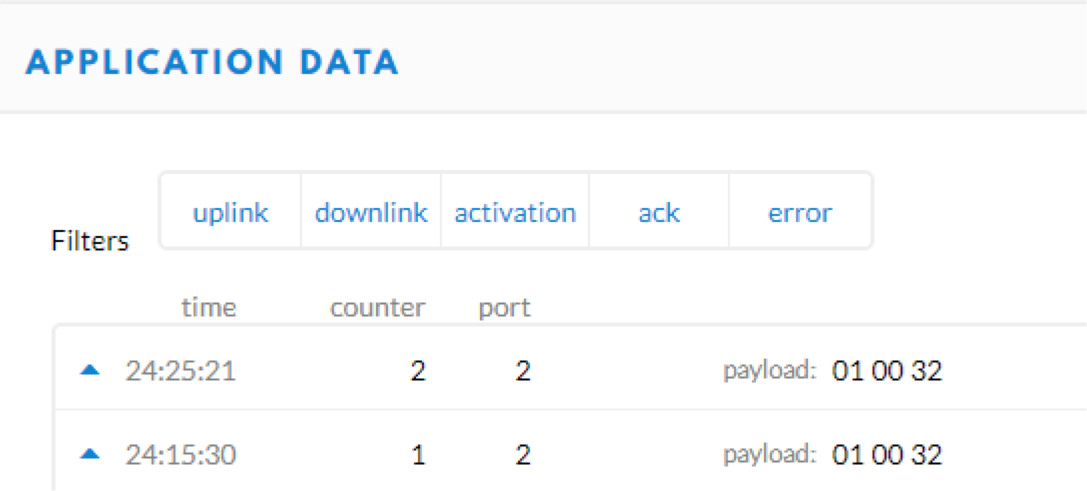


APPLICATION DATA

Filters: uplink | downlink | activation | ack | error

	time	counter	port				
▲	24:15:30	1	2		payload: 01 00 32		
▼	24:05:23		0				
▲	24:05:22	0	2	retry	payload: 01		
⚡	24:05:13				dev addr: 26 01 58 AC	app eui: A8 40 41 D1 71 82 1E 85	dev eui: A8 40 41 4C 21 82 1E 85
⚡	23:59:13				dev addr: 26 01 48 98	app eui: A8 40 41 D1 71 82 1E 85	dev eui: A8 40 41 4C 21 82 1E 85

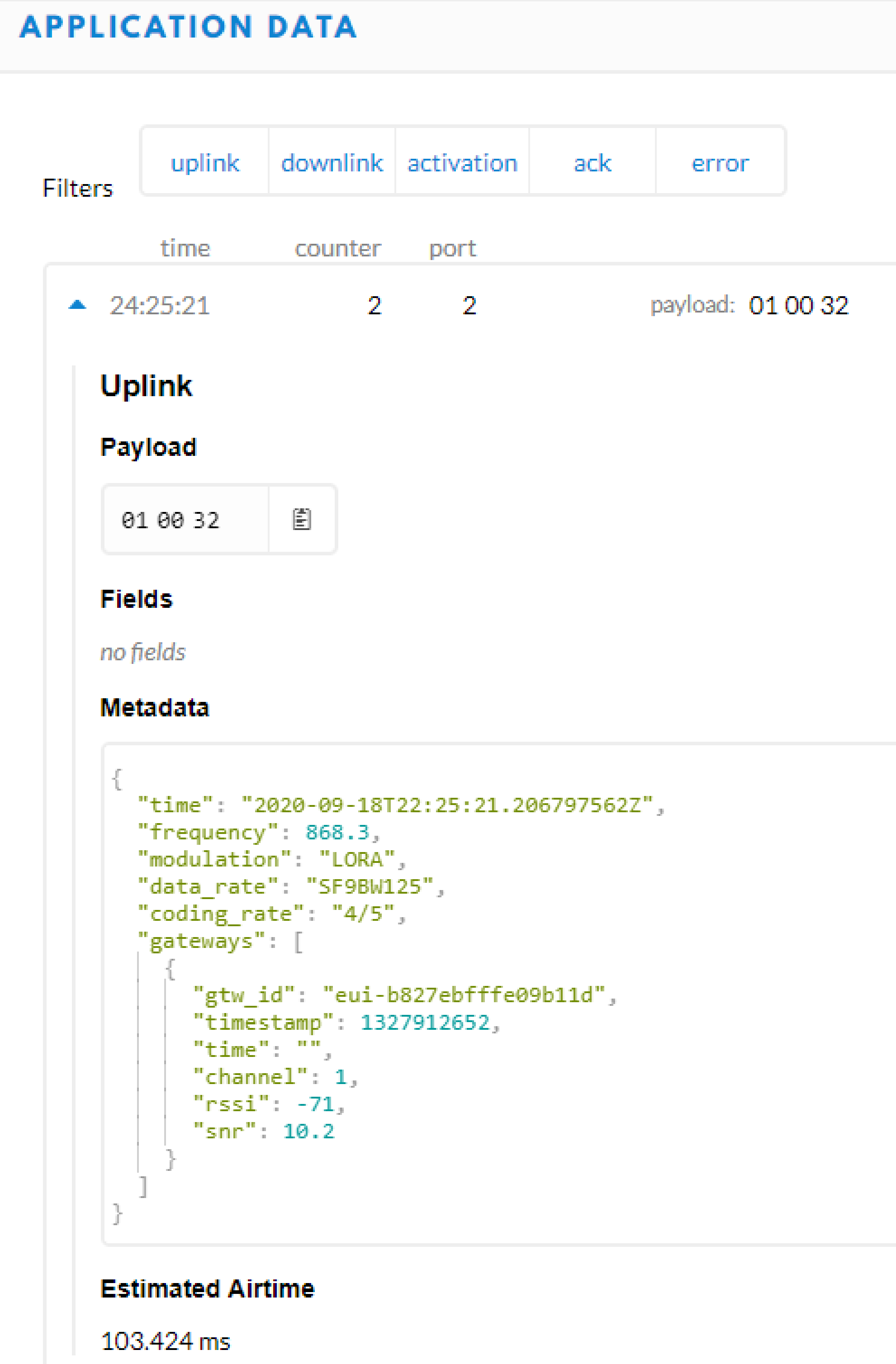
And even gets periodically each 10 minutes



APPLICATION DATA

Filters: uplink | downlink | activation | ack | error

	time	counter	port	
▲	24:25:21	2	2	payload: 01 00 32
▲	24:15:30	1	2	payload: 01 00 32



time	counter	port	
24:25:21	2	2	payload: 01 00 32

Uplink

Payload

```
01 00 32
```

Fields

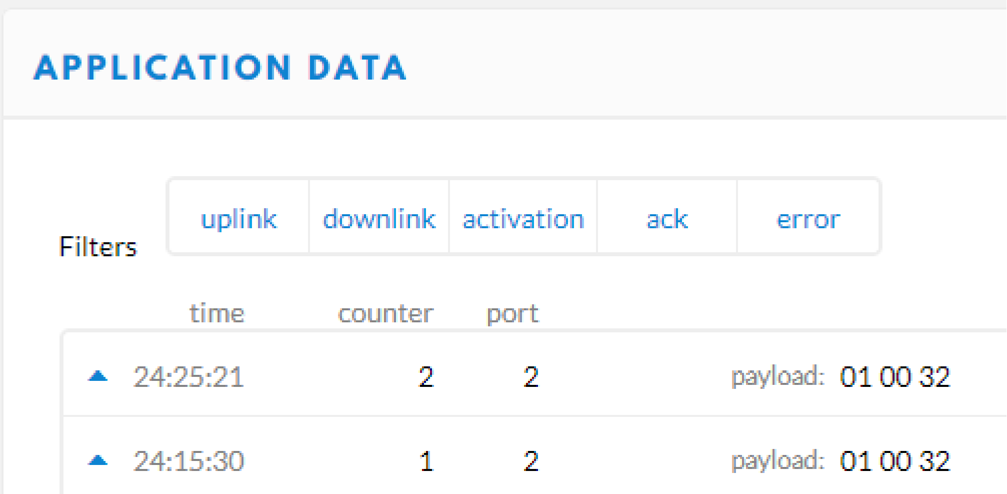
no fields

Metadata

```
{
  "time": "2020-09-18T22:25:21.206797562Z",
  "frequency": 868.3,
  "modulation": "LORA",
  "data_rate": "SF9BW125",
  "coding_rate": "4/5",
  "gateways": [
    {
      "gtw_id": "eui-b827ebfffe09b11d",
      "timestamp": 1327912652,
      "time": "",
      "channel": 1,
      "rssi": -71,
      "snr": 10.2
    }
  ]
}
```

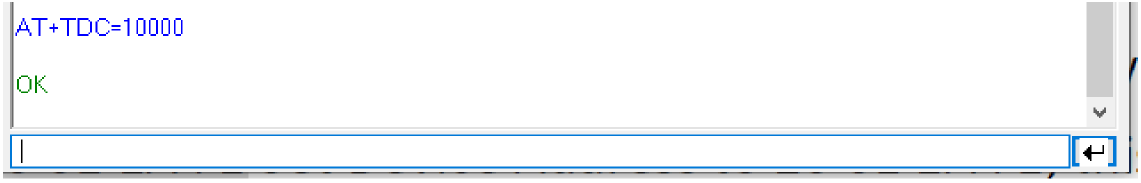
Estimated Airtime

103.424 ms



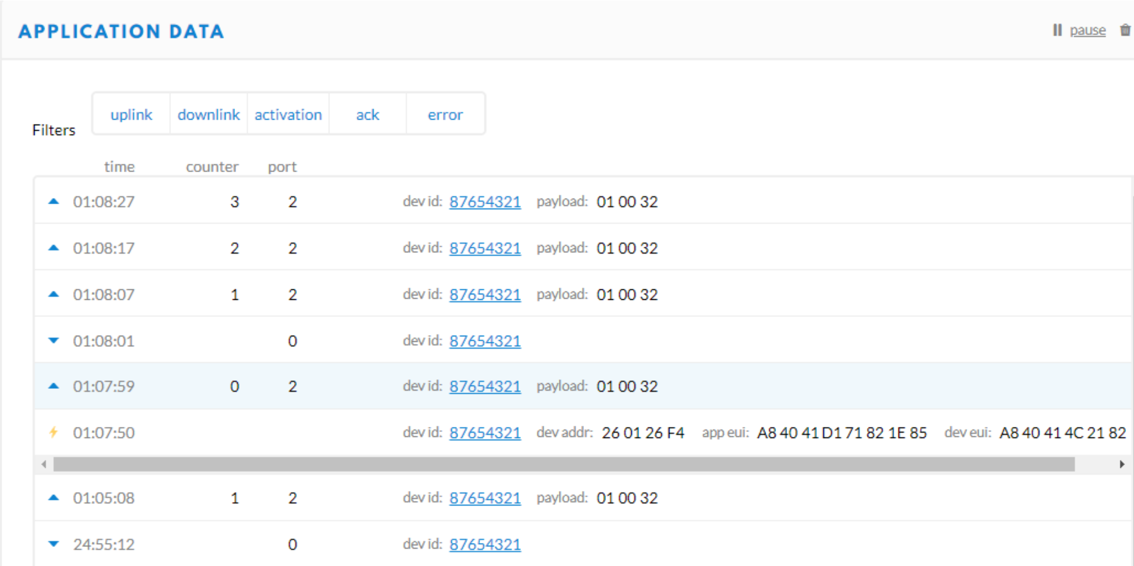
	time	counter	port	
▲	24:25:21	2	2	payload: 01 00 32
▲	24:15:30	1	2	payload: 01 00 32

Now let’s try to change the period from 60 seconds (default) to 10 seconds
From the program terminal



```
AT+TDC=10000

OK
```

Nothing changes
We have to reset or power off and on the Dragino device
Not we have a periodo f 10 seconds

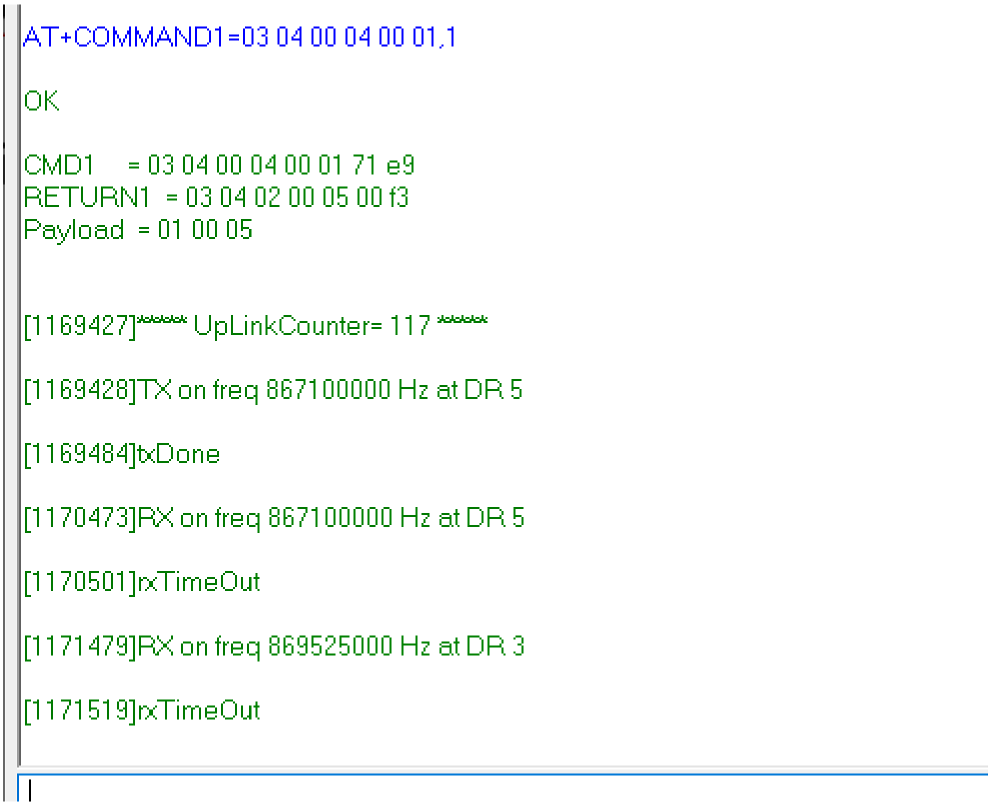


	time	counter	port				
▲	01:08:27	3	2	dev id: 87654321	payload: 01 00 32		
▲	01:08:17	2	2	dev id: 87654321	payload: 01 00 32		
▲	01:08:07	1	2	dev id: 87654321	payload: 01 00 32		
▼	01:08:01		0	dev id: 87654321			
▲	01:07:59	0	2	dev id: 87654321	payload: 01 00 32		
⚡	01:07:50			dev id: 87654321	dev addr: 26 01 26 F4	app eui: A8 40 41 D1 71 82 1E 85	dev eui: A8 40 41 4C 21 82
▲	01:05:08	1	2	dev id: 87654321	payload: 01 00 32		
▼	24:55:12		0	dev id: 87654321			

Let’s seehow to change the programmed transmisión
Now we will try read on register 4 (This is the preset trip current)

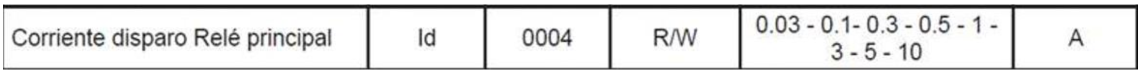
From serial Terminal
AT+COMMAND1=03 04 00 04 00 01,1

Yes



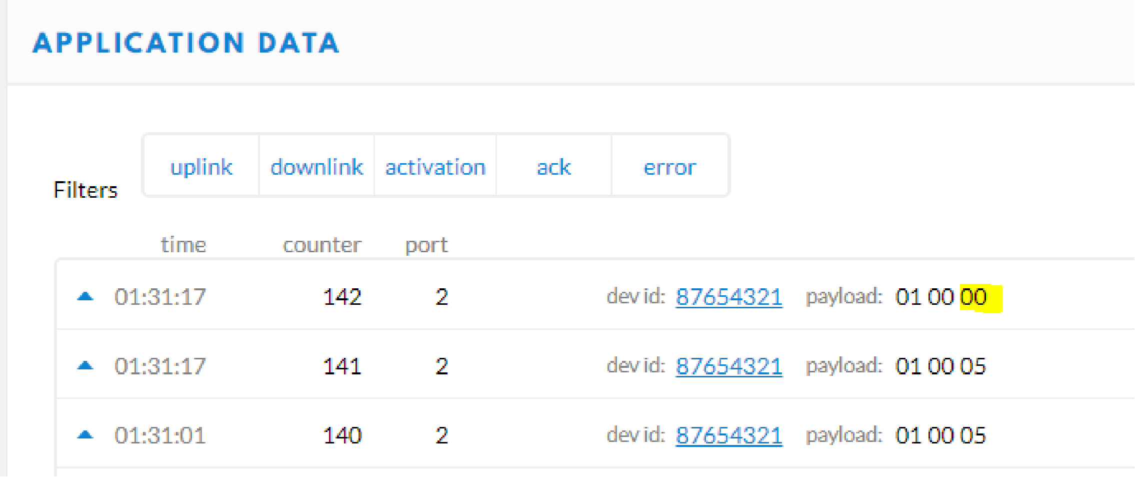
```
AT+COMMAND1=03 04 00 04 00 01,1

OK

CMD1     = 03 04 00 04 00 01 71 e9
RETURN1  = 03 04 02 00 05 00 f3
Payload  = 01 00 05


[1169427]***** UpLinkCounter= 117 *****

[1169428]TX on freq 867100000 Hz at DR 5

[1169484]txDone

[1170473]RX on freq 867100000 Hz at DR 5

[1170501]rxTimeOut

[1171479]RX on freq 869525000 Hz at DR 3

[1171519]rxTimeOut
```

Yes now we get 5 corresponding to 3.0 A
Since the posible values are



Corriente disparo Relé principal	Id	0004	R/W	0.03 - 0.1- 0.3 - 0.5 - 1 - 3 - 5 - 10	A

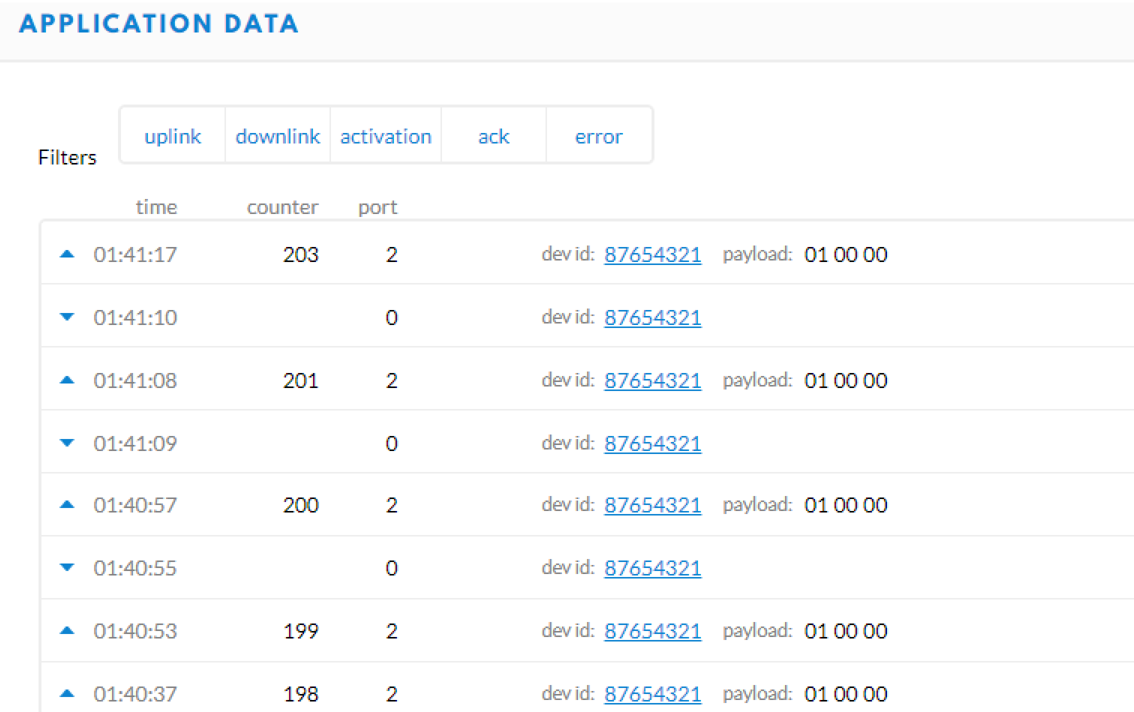
Now we change the preset on the leak circuit breaker relay to 0.03 (30mA)
Yes, 0 is 30mA!



APPLICATION DATA

time	counter	port	
▲ 01:31:17	142	2	dev id: 87654321 payload: 01 00 00
▲ 01:31:17	141	2	dev id: 87654321 payload: 01 00 05
▲ 01:31:01	140	2	dev id: 87654321 payload: 01 00 05

Finally we disconnect the RS485 cable from The leakage realy RGU10 and we get the following result

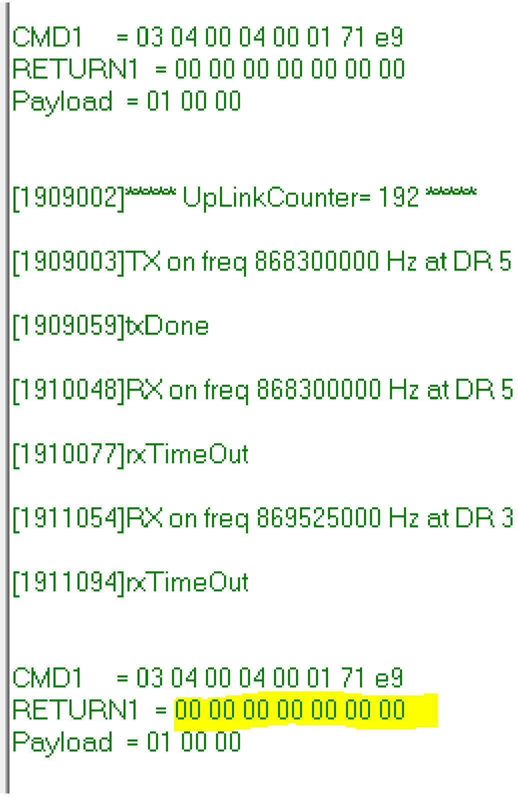


APPLICATION DATA

	time	counter	port	
▲	01:41:17	203	2	dev id: 87654321 payload: 01 00 00
▼	01:41:10		0	dev id: 87654321
▲	01:41:08	201	2	dev id: 87654321 payload: 01 00 00
▼	01:41:09		0	dev id: 87654321
▲	01:40:57	200	2	dev id: 87654321 payload: 01 00 00
▼	01:40:55		0	dev id: 87654321
▲	01:40:53	199	2	dev id: 87654321 payload: 01 00 00
▲	01:40:37	198	2	dev id: 87654321 payload: 01 00 00

Obvious since there is no one listening at address 3

But the payload is made of bytes 4 and 5

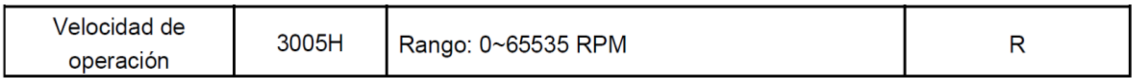


```
CMD1    = 03 04 00 04 00 01 71 e9
RETURN1  = 00 00 00 00 00 00 00
Payload  = 01 00 00


[1909002]***** UpLinkCounter= 192 *****

[1909003]TX on freq 868300000 Hz at DR 5

[1909059]txDone

[1910048]RX on freq 868300000 Hz at DR 5

[1910077]rxTimeOut

[1911054]RX on freq 869525000 Hz at DR 3

[1911094]rxTimeOut


CMD1    = 03 04 00 04 00 01 71 e9
RETURN1  = 00 00 00 00 00 00 00
Payload  = 01 00 00
```

Variable Speed drive and Dragino RS485 to Lora for Monitoring

Salicru CV30 VFD

Now we want to read the speed of a VFD
Our VFD i son address 1
9600 bauds
No Parity, 8 bits, 1 stop bit: N,8,1
Reading the speed on register:



Velocidad de operación	3005H	Rango: 0~65535 RPM	R

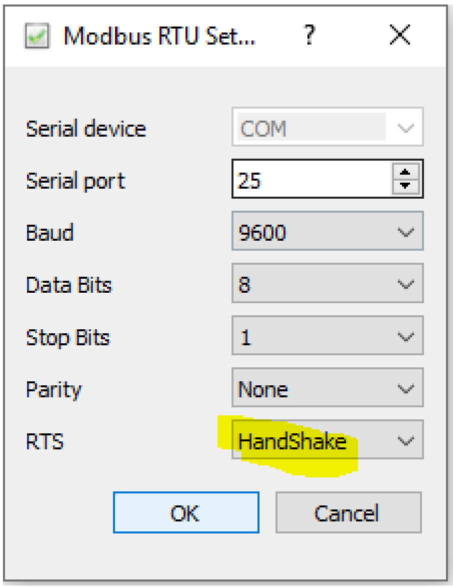
Parameter 12293 in decimal is 3005 in Hex

First we try with qModMaster terminal, in order to find out the right parameters

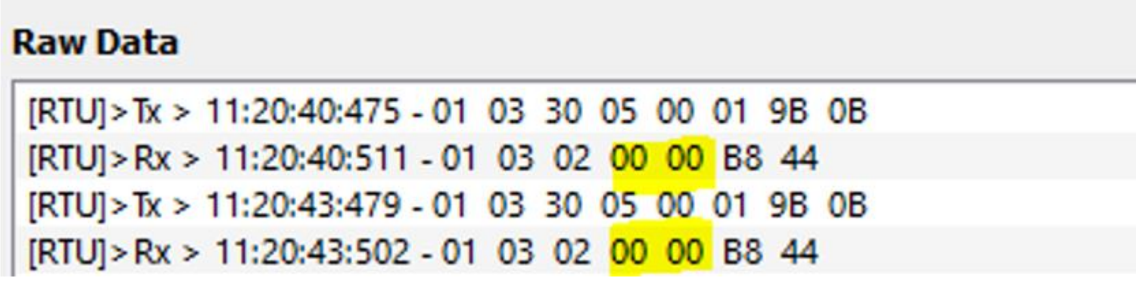
Connecting the VFD to the computer with a RS-485 to USB converter

Yes, we have the VFD stopped, with 0 speed as we see here 00 00 RPM

If yo do not set the RTS to Handshake you will get an error



Serial device	COM
Serial port	25
Baud	9600
Data Bits	8
Stop Bits	1
Parity	None
RTS	HandShake



Raw Data

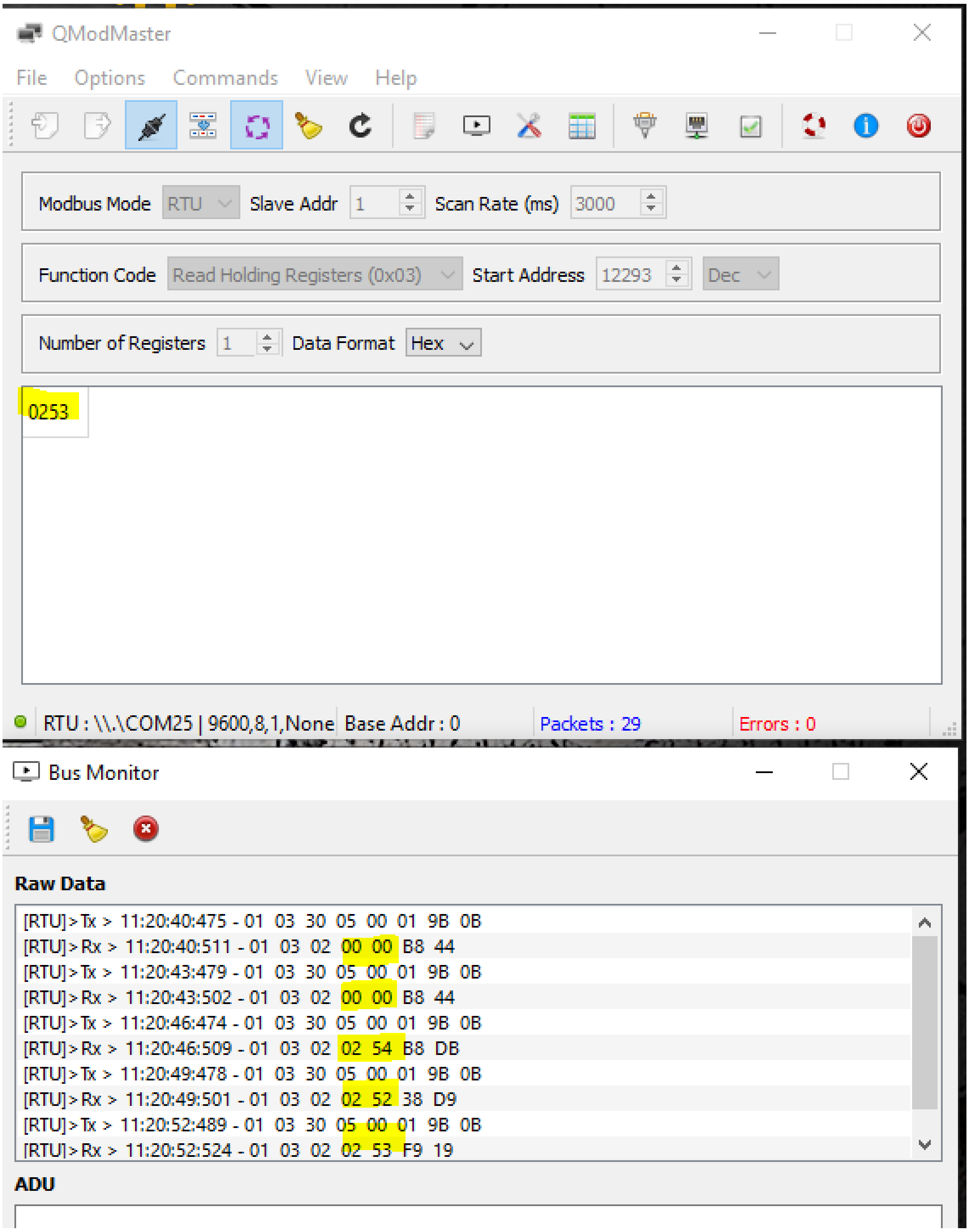
```
[RTU]>Tx > 11:20:40:475 - 01  03  30  05  00  01  9B  0B
[RTU]>Rx > 11:20:40:511 - 01  03  02  00  00  B8  44
[RTU]>Tx > 11:20:43:479 - 01  03  30  05  00  01  9B  0B
[RTU]>Rx > 11:20:43:502 - 01  03  02  00  00  B8  44
```

The we start the motor manually with a digital input on the drive (speed is controlled by Modbus)

We see

595 RPM (0253 Hex)

It Works!



Modbus Mode RTU Slave Addr 1 Scan Rate (ms) 3000

Function Code Read Holding Registers (0x03) Start Address 12293 Dec

Number of Registers 1 Data Format Hex

0253

RTU : \\.\COM25 | 9600,8,1,None Base Addr : 0 Packets : 29 Errors : 0

Raw Data

```
[RTU]>Tx > 11:20:40:475 - 01  03  30  05  00  01  9B  0B
[RTU]>Rx > 11:20:40:511 - 01  03  02  00  00  B8  44
[RTU]>Tx > 11:20:43:479 - 01  03  30  05  00  01  9B  0B
[RTU]>Rx > 11:20:43:502 - 01  03  02  00  00  B8  44
[RTU]>Tx > 11:20:46:474 - 01  03  30  05  00  01  9B  0B
[RTU]>Rx > 11:20:46:509 - 01  03  02  02  54  B8  DB
[RTU]>Tx > 11:20:49:478 - 01  03  30  05  00  01  9B  0B
[RTU]>Rx > 11:20:49:501 - 01  03  02  02  52  38  D9
[RTU]>Tx > 11:20:52:489 - 01  03  30  05  00  01  9B  0B
[RTU]>Rx > 11:20:52:524 - 01  03  02  02  53  F9  19
```

ADU

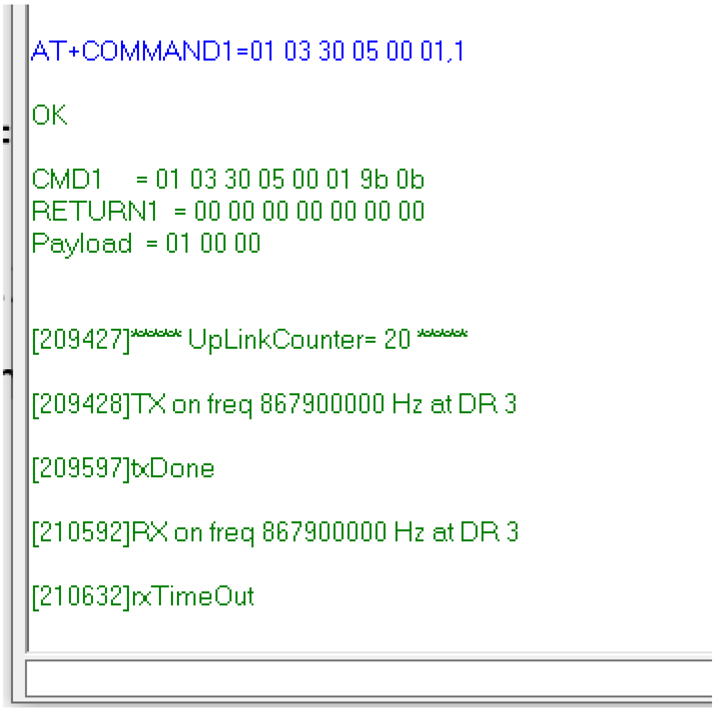
So now we know the right command to programm to The Dragino RS-485 transactions

And this is:

AT+COMMAND1=01 03 30 05 00 01,1

AT+DATACUT1=7,2,4~5 (But this was already programmed on the Dragino so we do not need to resend this AT command)

Yes, we have already programmed the Dragion, but the reponse is still 000000000 since we have not connected the VFD to the Dragino

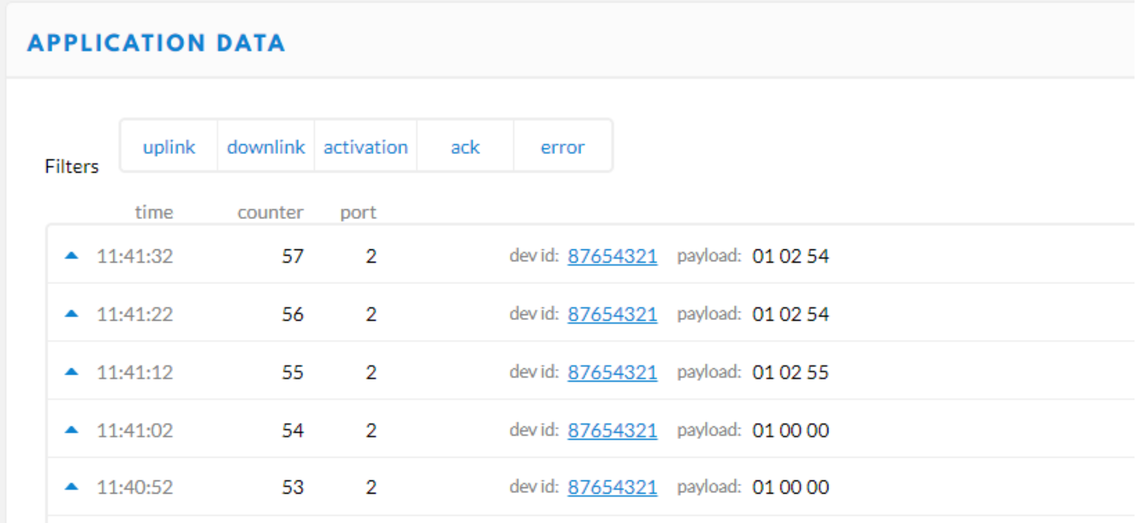


```
AT+COMMAND1=01 03 30 05 00 01,1

OK

CMD1     = 01 03 30 05 00 01 9b 0b
RETURN1  = 00 00 00 00 00 00 00 00
Payload  = 01 00 00


[209427]***** UpLinkCounter= 20 *****

[209428]TX on freq 867900000 Hz at DR 3

[209597]txDone

[210592]RX on freq 867900000 Hz at DR 3

[210632]rxTimeOut
```

Let’s connect the VFD to Dragino RS-485-LN

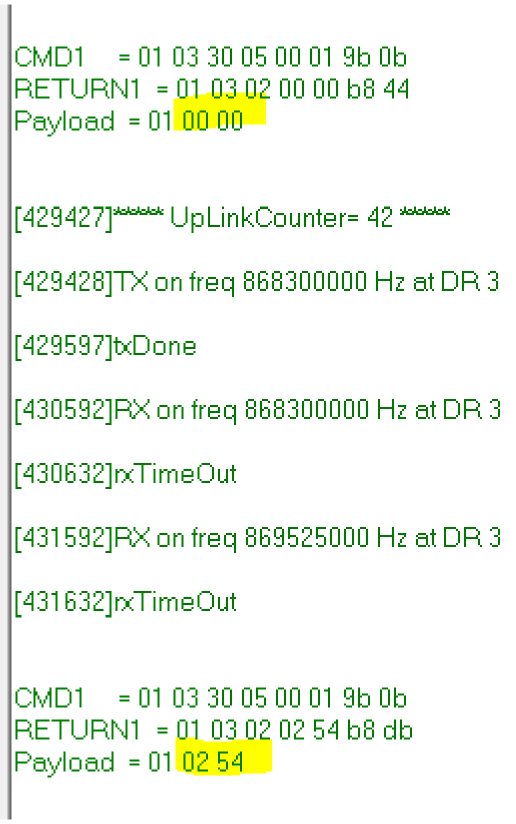
Voilà,

First drive stopped, and the drive started at speed 02 54 in Hex (595 RPM in decimal)

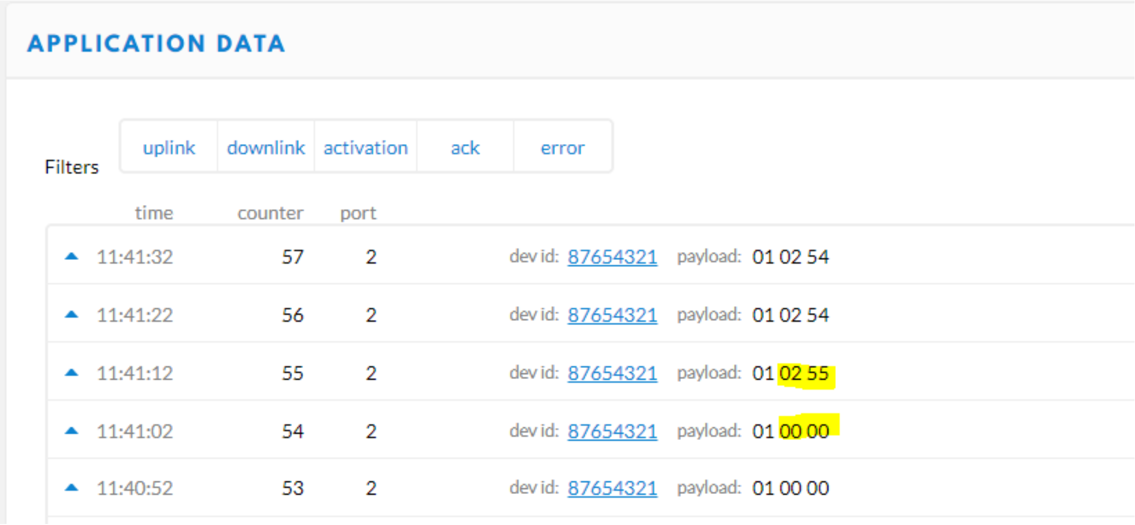


APPLICATION DATA

	time	counter	port		
▲	11:41:32	57	2	dev id: 87654321	payload: 01 02 54
▲	11:41:22	56	2	dev id: 87654321	payload: 01 02 54
▲	11:41:12	55	2	dev id: 87654321	payload: 01 02 55
▲	11:41:02	54	2	dev id: 87654321	payload: 01 00 00
▲	11:40:52	53	2	dev id: 87654321	payload: 01 00 00

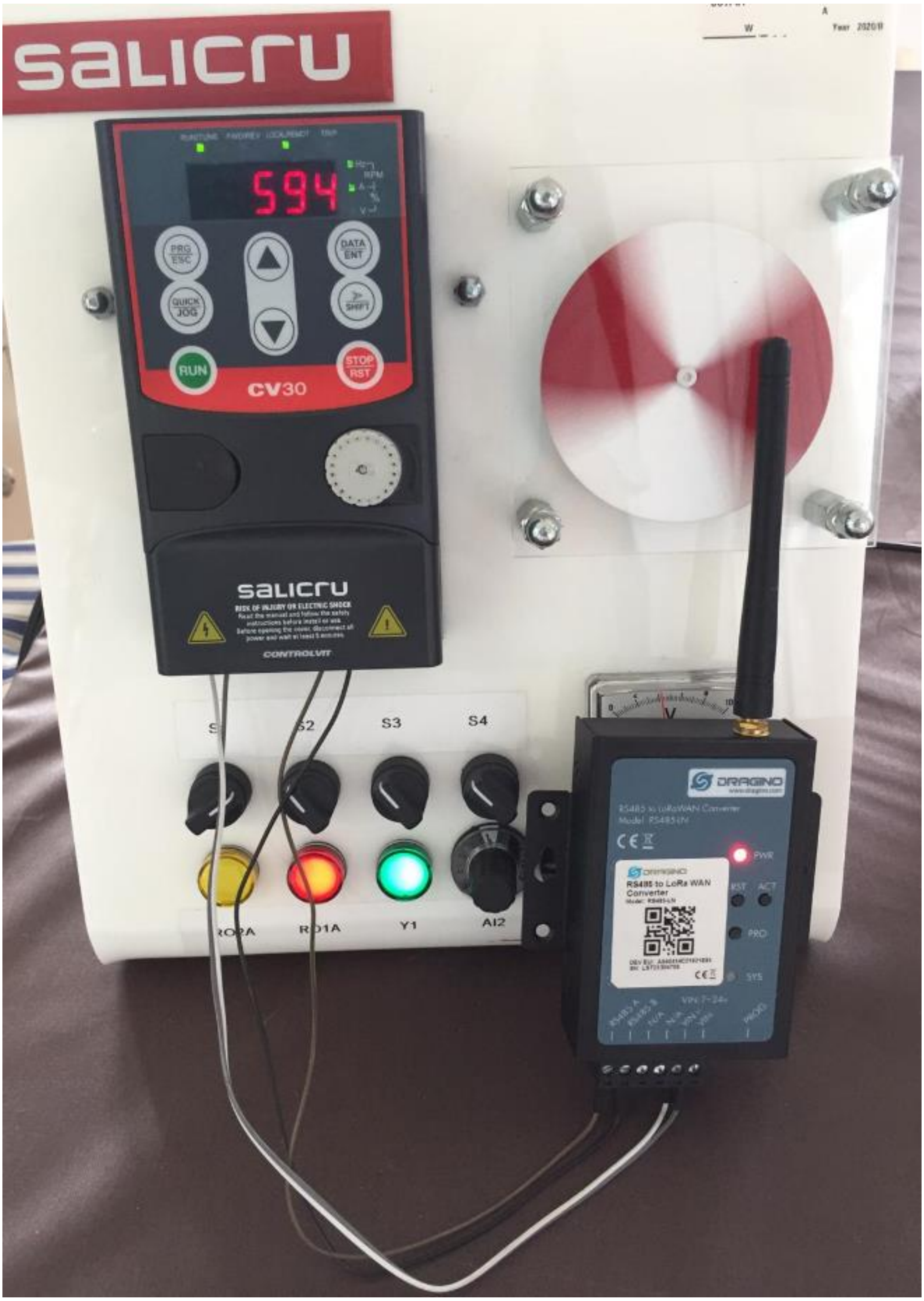


```
CMD1     = 01 03 30 05 00 01 9b 0b
RETURN1  = 01 03 02 00 00 b8 44
Payload  = 01 00 00


[429427]***** UpLinkCounter= 42 *****

[429428]TX on freq 868300000 Hz at DR 3

[429597]txDone

[430592]RX on freq 868300000 Hz at DR 3

[430632]rxTimeOut

[431592]RX on freq 869525000 Hz at DR 3

[431632]rxTimeOut


CMD1     = 01 03 30 05 00 01 9b 0b
RETURN1  = 01 03 02 02 54 b8 db
Payload  = 01 02 54
```

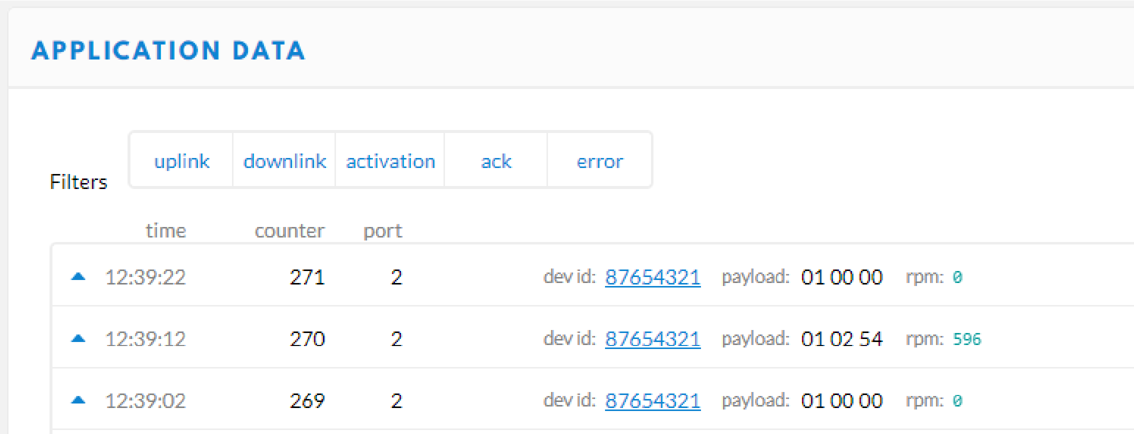


APPLICATION DATA

	time	counter	port		
▲	11:41:32	57	2	dev id: 87654321	payload: 01 02 54
▲	11:41:22	56	2	dev id: 87654321	payload: 01 02 54
▲	11:41:12	55	2	dev id: 87654321	payload: 01 02 55
▲	11:41:02	54	2	dev id: 87654321	payload: 01 00 00
▲	11:40:52	53	2	dev id: 87654321	payload: 01 00 00



But let’s decode the payload since it is in Hex

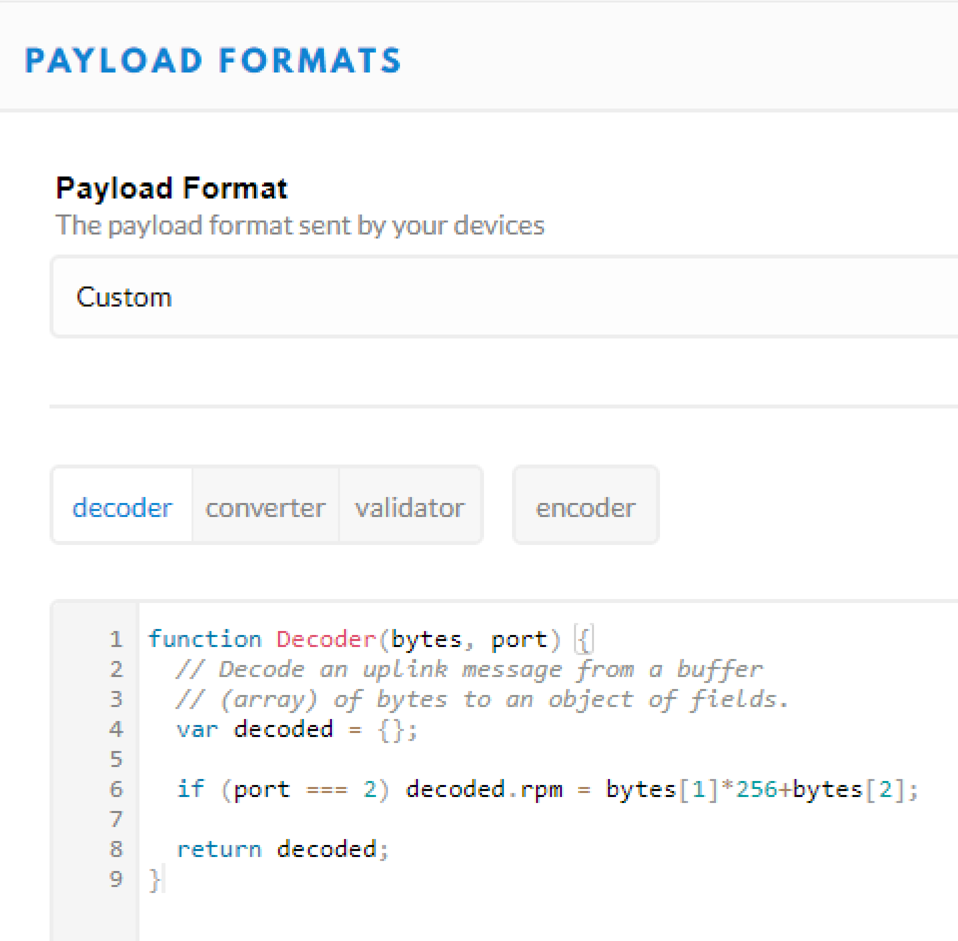


APPLICATION DATA

Filters

	time	counter	port	
▲	12:39:22	271	2	dev id: 87654321 payload: 01 00 00 rpm: 0
▲	12:39:12	270	2	dev id: 87654321 payload: 01 02 54 rpm: 596
▲	12:39:02	269	2	dev id: 87654321 payload: 01 00 00 rpm: 0

This is the payload decoder



PAYLOAD FORMATS

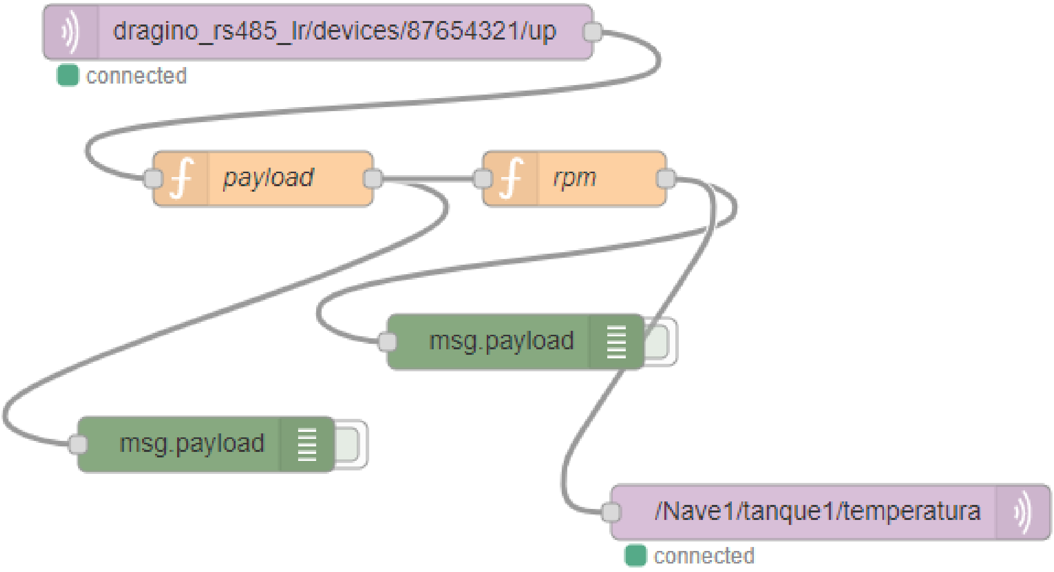
Payload Format

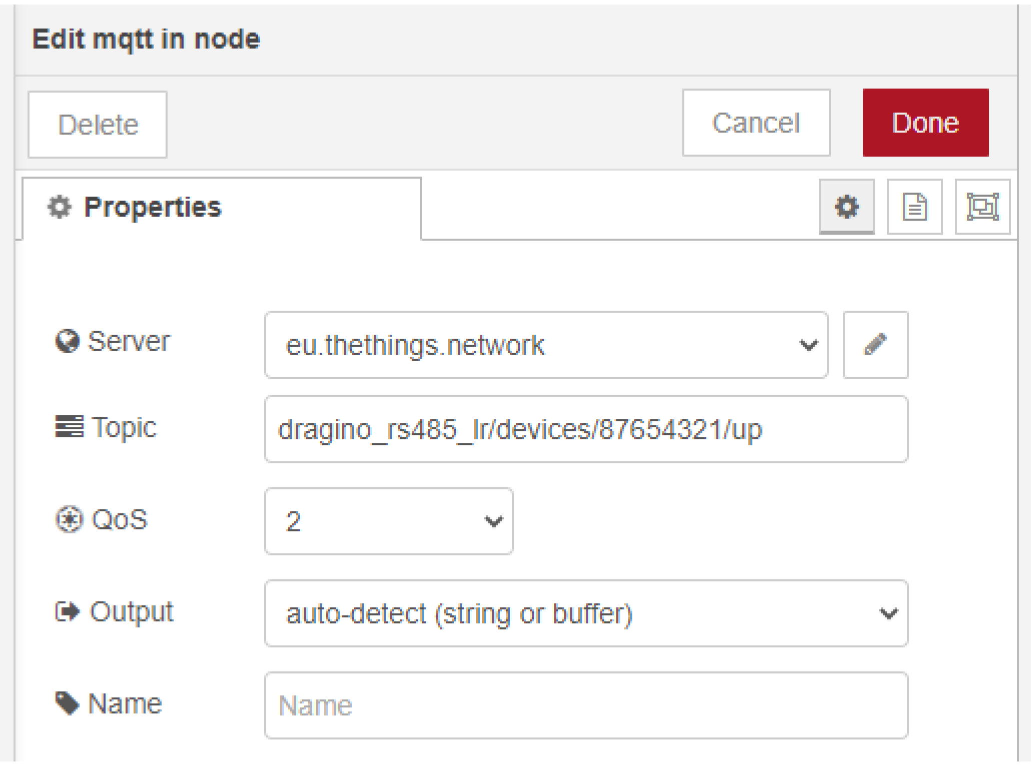
The payload format sent by your devices

Custom

```
function Decoder(bytes, port) {
  // Decode an uplink message from a buffer
  // (array) of bytes to an object of fields.
  var decoded = {};

  if (port === 2) decoded.rpm = bytes[1]*256+bytes[2];

  return decoded;
}
```

How to send the speed values to a mobile phone with “IoT On Off “ App


dragino_rs485_lr/devices/87654321/up
connected
payload
rpm
msg.payload
msg.payload
/Nave1/tanque1/temperatura
connected



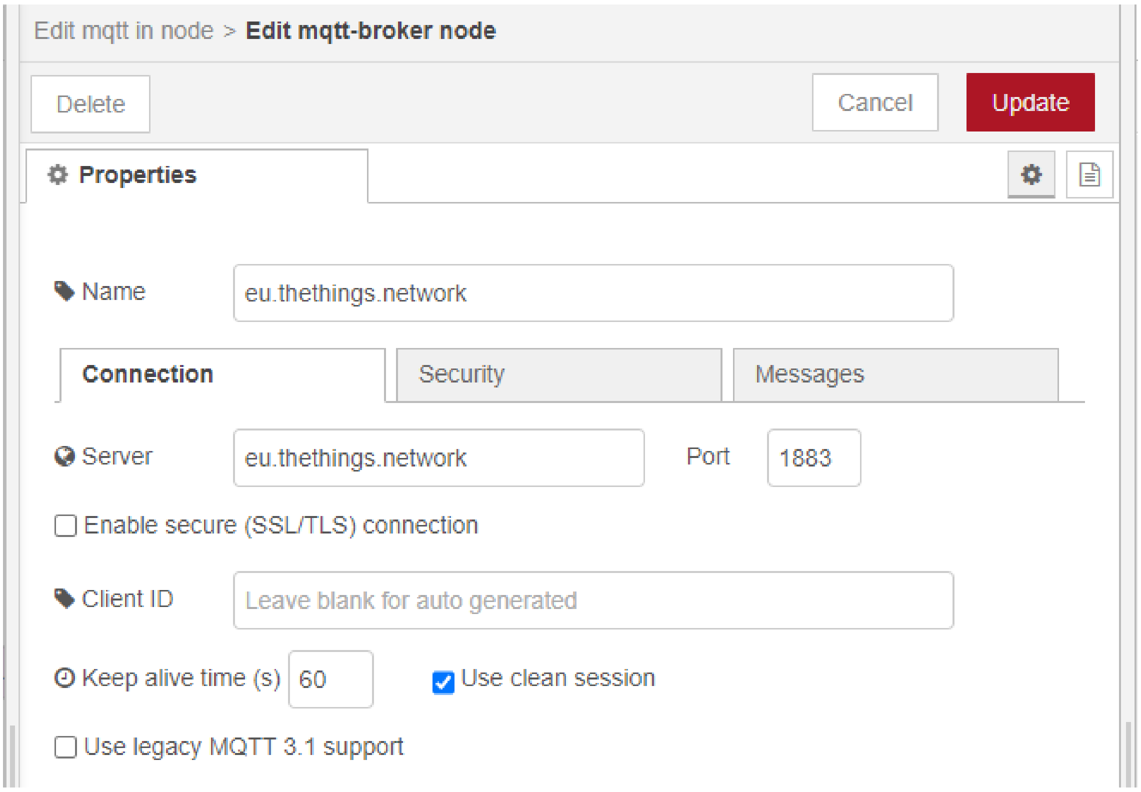
Server eu.thethings.network

Topic dragino_rs485_lr/devices/87654321/up

QoS 2

Output auto-detect (string or buffer)

Name Name



Name: eu.thethings.network

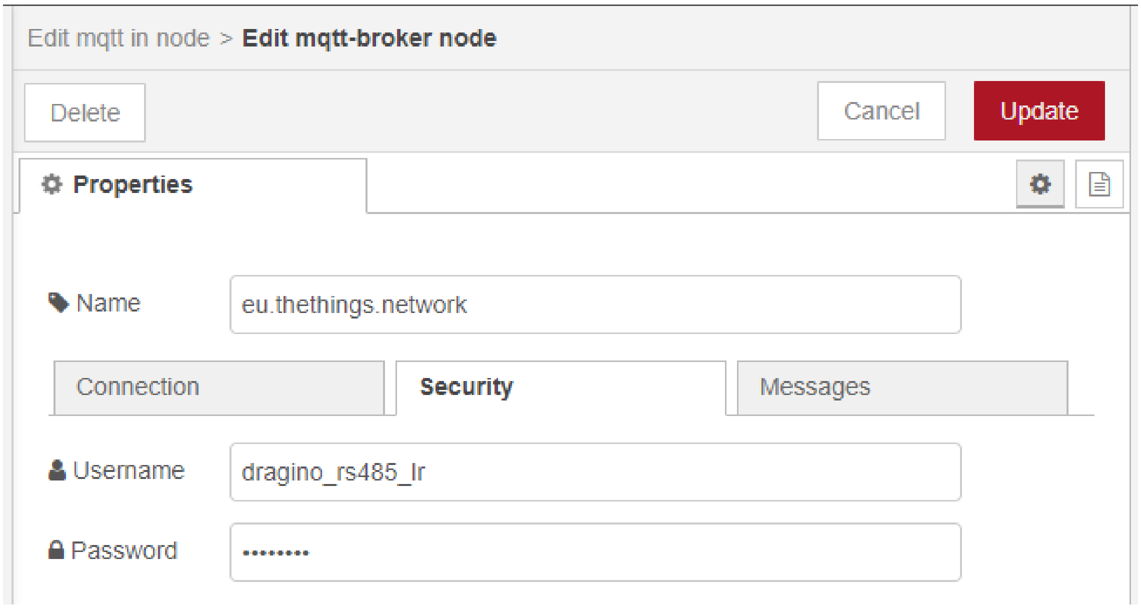
Server: eu.thethings.network Port: 1883

□ Enable secure (SSL/TLS) connection

Client ID: Leave blank for auto generated

Keep alive time (s): 60 ☑ Use clean session

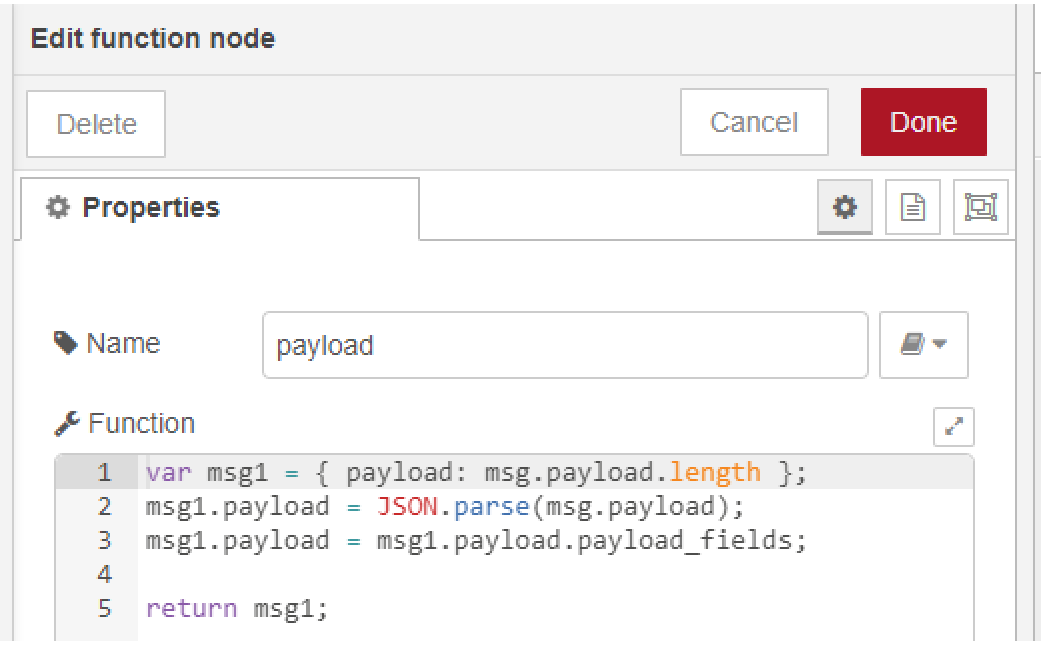
□ Use legacy MQTT 3.1 support



Name: eu.thethings.network

Username: dragino_rs485_lr

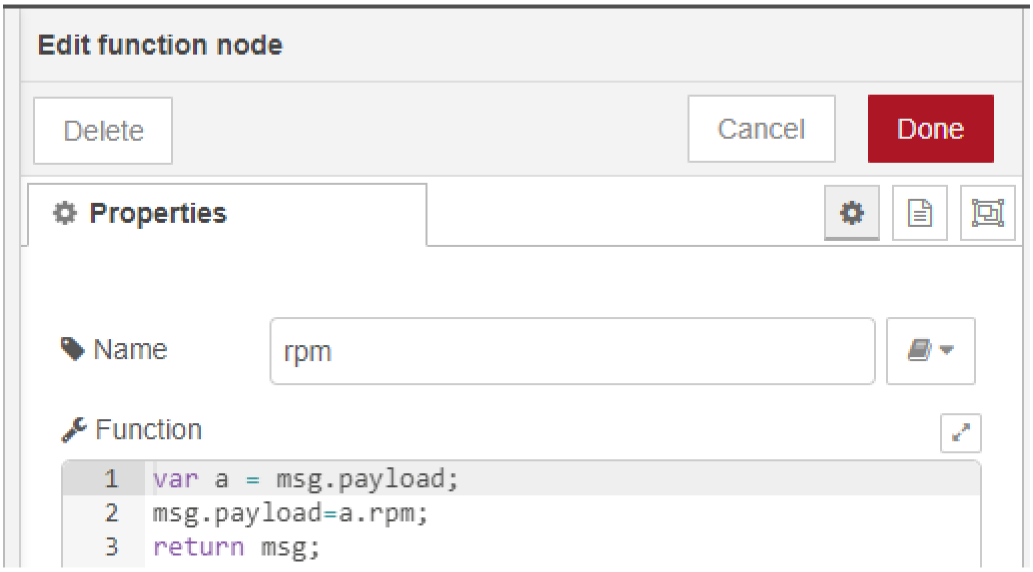
Password: ••••••••



Name: payload

Function

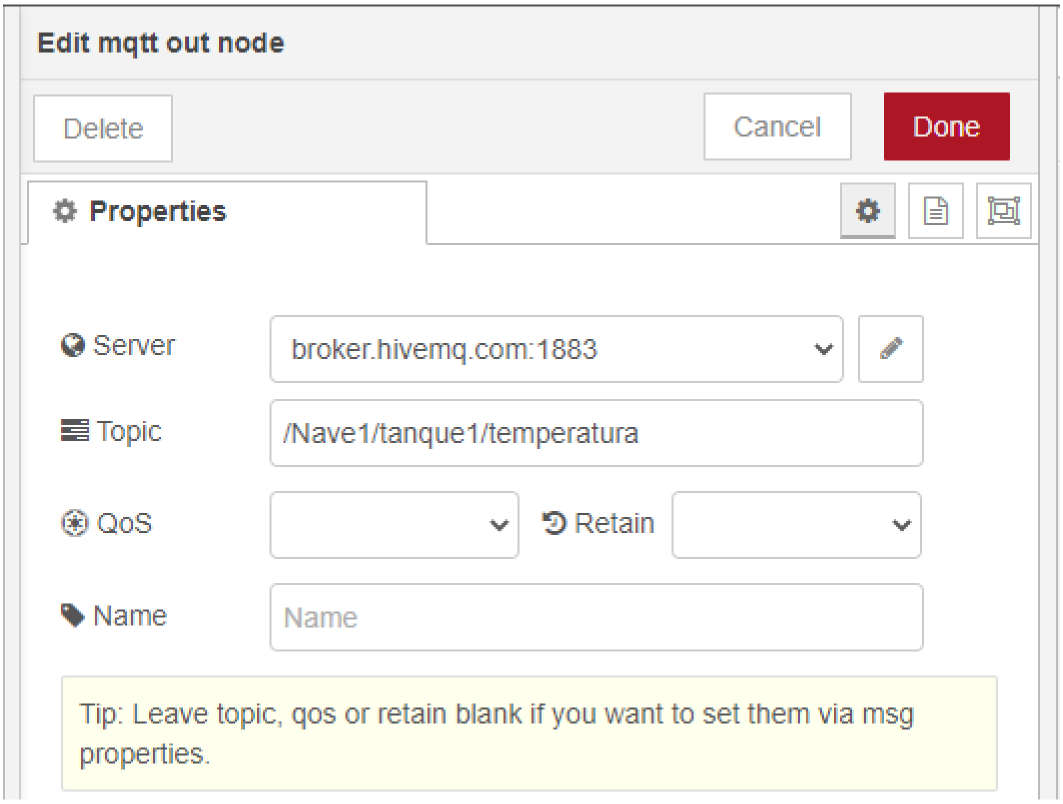
```
var msg1 = { payload: msg.payload.length };
msg1.payload = JSON.parse(msg.payload);
msg1.payload = msg1.payload.payload_fields;

return msg1;
```



Name: rpm

Function

```
var a = msg.payload;
msg.payload=a.rpm;
return msg;
```



Edit mqtt out node

Server: broker.hivemq.com:1883

Topic: /Nave1/tanque1/temperatura

QoS:

Retain:

Name: Name

Tip: Leave topic, qos or retain blank if you want to set them via msg properties.

You can find the Node-RED code here:

https://github.com/xavierflorensa/Salicru-VFD-Dragino-RS485-to-LoRaWAN-to-IOT-OnOff-App-Node-RED-flow

How to change the speed from TTN downlink message injection

Configure AT+COMMANDx or AT+DATACUTx AF MM NN LL XX XX XX XX YY	AF	Dynamic

Type Code 0xAF

0xAF downlink command can be used to set AT+COMMANDx or AT+DATACUTx.

Note: if user use AT+COMMANDx to add a new command, he also need to send AT+DATACUTx downlink.

Format: AF MM NN LL XX XX XX XX YY

Where:

- MM: the ATCOMMAND or AT+DATACUT to be set. Value from 01 ~ 0F,
- NN: 0: no CRC; 1: add CRC-16/MODBUS ; 2: set the AT+DATACUT value.
- LL: The length of AT+COMMAND or AT+DATACUT command
- XX XX XX XX: AT+COMMAND or AT+DATACUT command
- YY: If YY=0, RS485-LN will execute the downlink command without uplink; if YY=1, RS485-LN will execute an uplink after got this command.

Example:

AF 03 01 06 0A 05 00 04 00 01 00: Same as AT+COMMAND3=0A 05 00 04 00 01,1

How to change the speed from a mobile phone