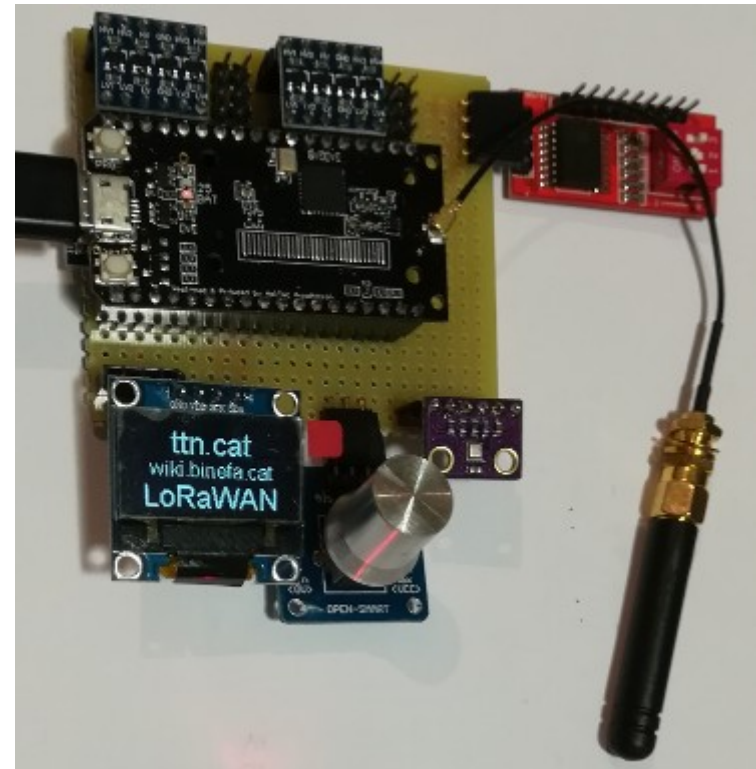


LoRaWAN™



<http://www.thomasclausen.net/wp-content/uploads/2016/09/2016-A-Study-of-LoRa-Long-Range-Low-Power-Networks-for-the-Internet-of-Things.pdf>

Què és?

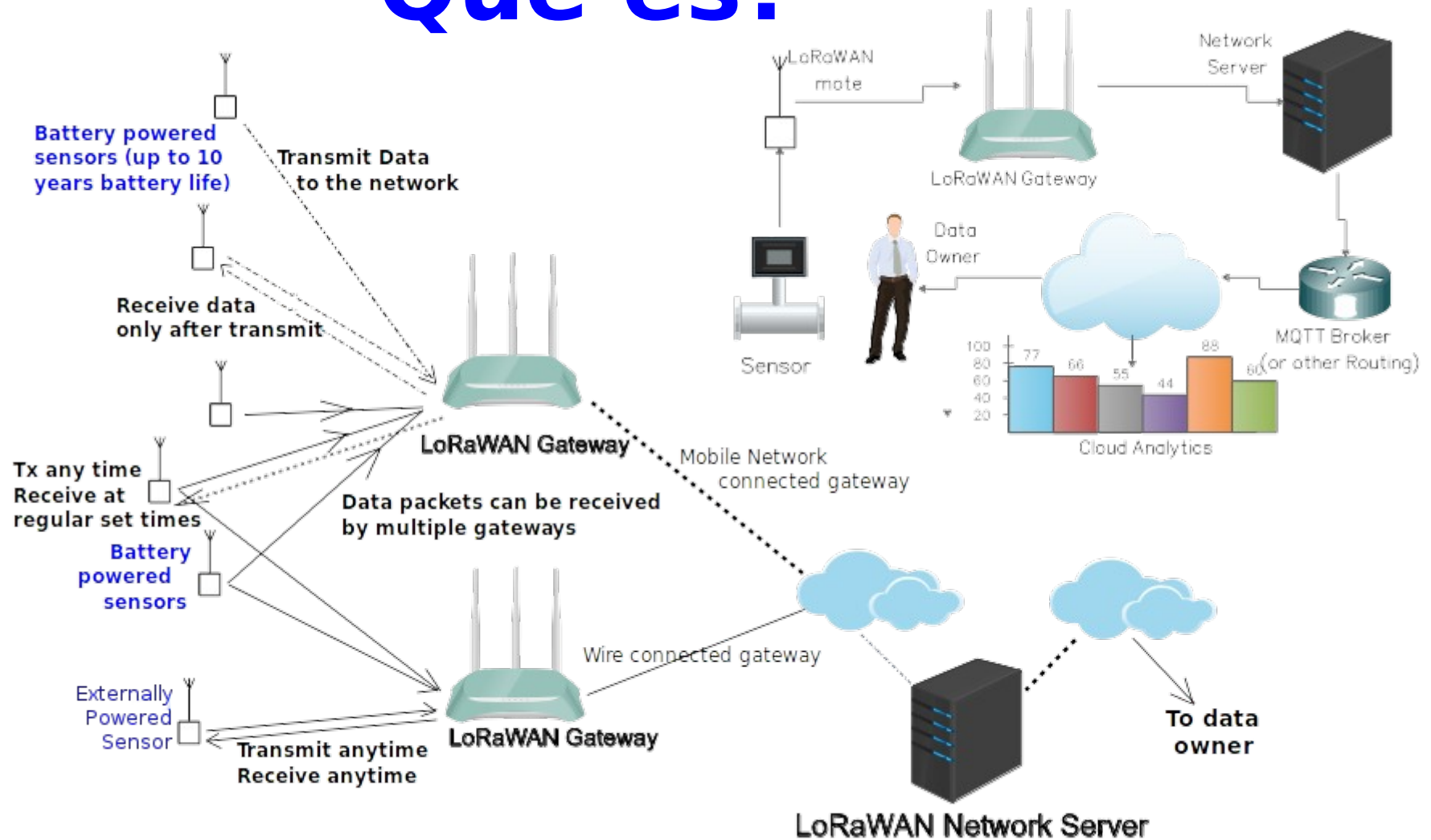
LoRaWAN™ is a Low Power Wide Area Network (**LPWAN**) specification intended for wireless battery operated Things in a regional, national or global network. **LoRaWAN** targets key requirements of **Internet of Things** such as **secure bi-directional communication**, mobility and localization services.

Què és?

LoRaWAN™ és una especificació d'una xarxa **LPWAN** (Low Power Wide Area Network) proposada per la **LoRa Alliance** i pensada per comunicar dispositius de baix cost i baix consum alimentats per bateries.

L'especificació cobreix les **capes PHY i MAC** de la xarxa, deixant les capes superiors a les aplicacions.

Què és?



LoRaWAN™

Terminologia

Terminology

End Device, Node, Mote - an object with an embedded low-power communication device.

Gateway - antennas that receive broadcasts from End Devices and send data back to End Devices.

Network Server - servers that route messages from End Devices to the right Application, and back.

Application - a piece of software, running on a server.

Uplink Message - a message from a Device to an Application.

Downlink Message - a message from an Application to a Device.

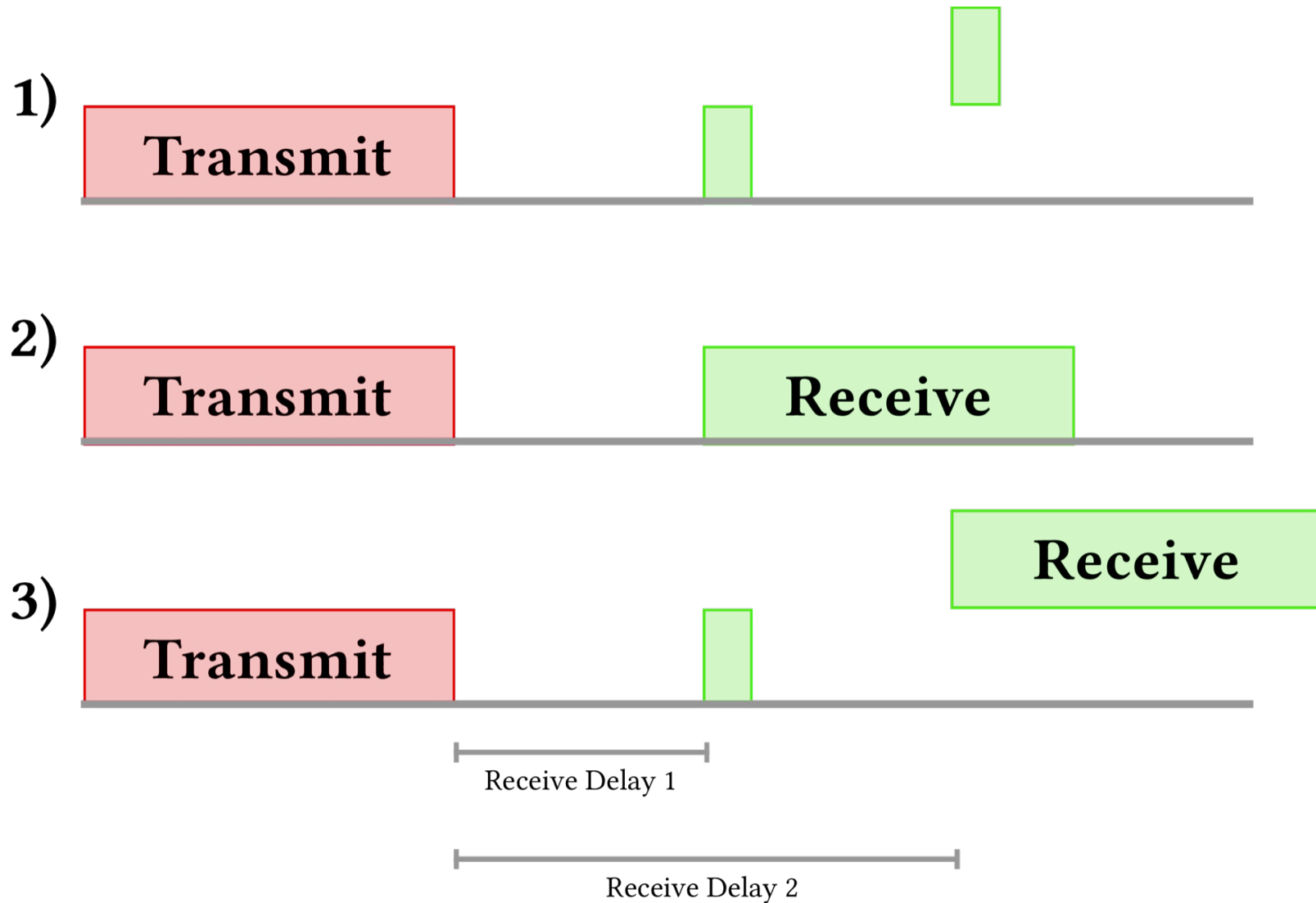
Classes de dispositius finals

LoRaWAN endpoint classes

Accordingly the LoRaWAN supports three classes of endpoints:



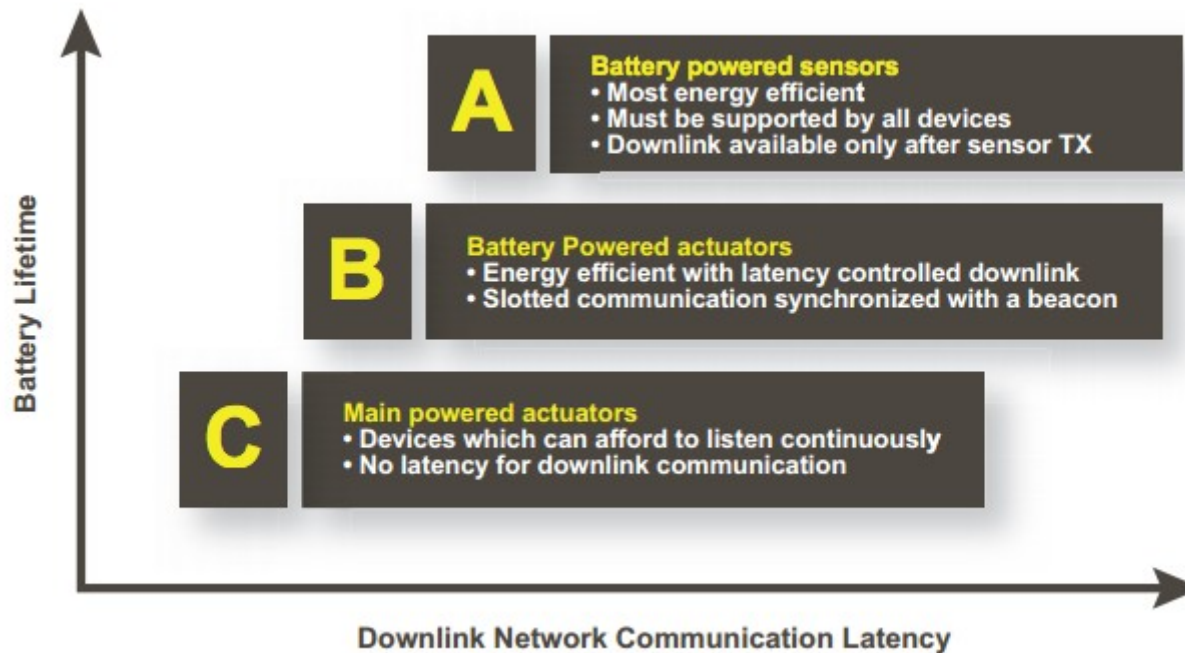
Classe A de dispositius finals



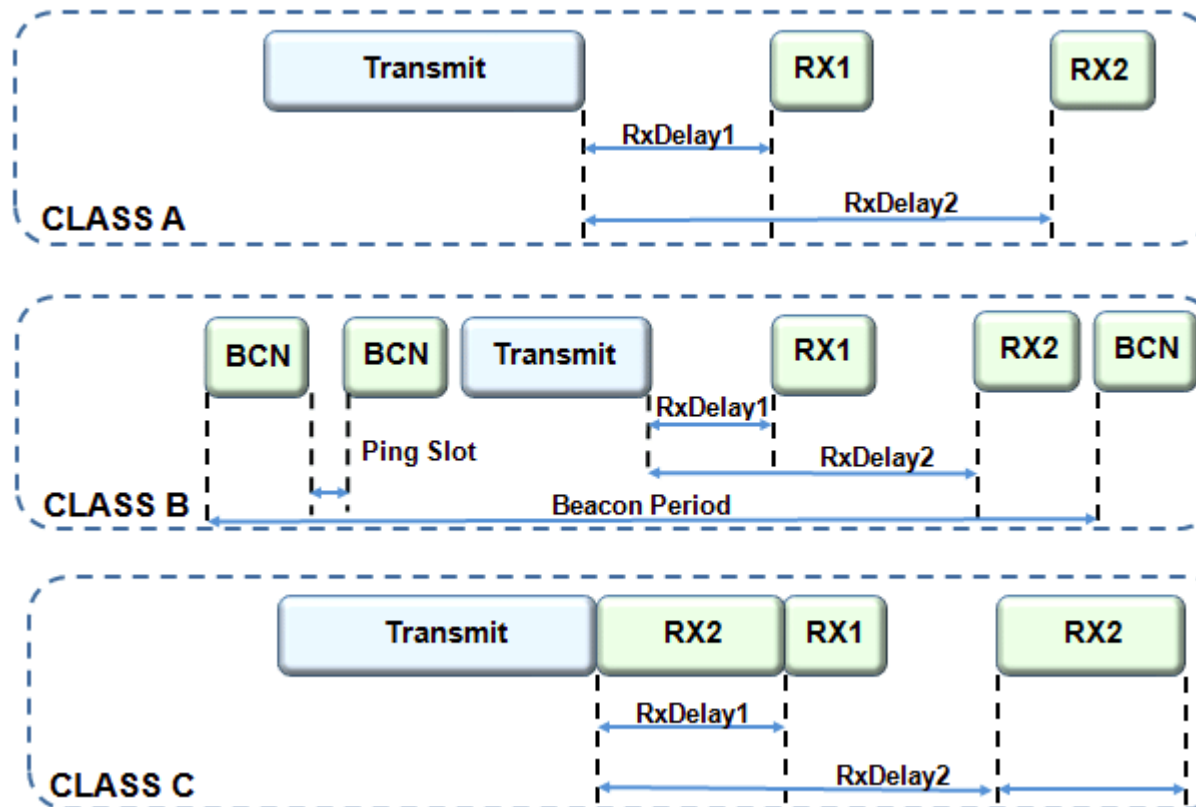
Classes de dispositius finals

CLASS NAME	INTENDED USAGE
A (« all »)	Battery powered sensors , or actuators with no latency constraint Most energy efficient communication class. Must be supported by all devices
B (« beacon »)	Battery powered actuators Energy efficient communication class for latency controlled downlink. Based on slotted communication synchronized with a network beacon.
C (« continuous »)	Mains powered actuators Devices which can afford to listen continuously. No latency for downlink communication.

Classes de dispositius finals



Classes de dispositius finals



Modulació CHIRP

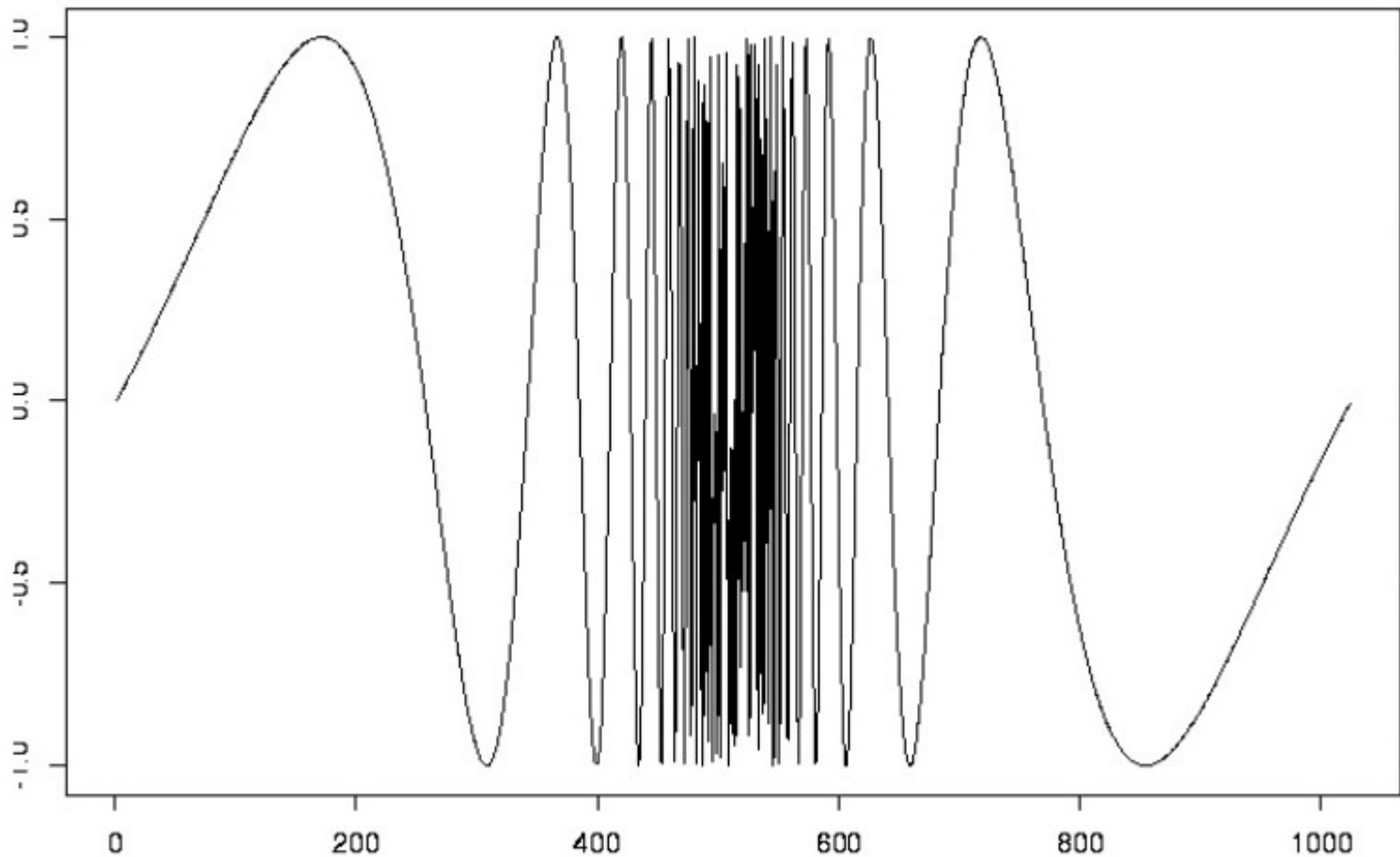
Chirp

- A **chirp** is a signal in which the frequency increases ('up-chirp') or decreases ('down-chirp') with time.
- In some sources, the term **chirp** is used interchangeably with **sweep signal**.
- It is commonly used in sonar and radar, but has other applications, such as in spread spectrum communications.
- In spread spectrum usage, SAW devices such as RACs are often used to generate and demodulate the chirped signals.
- In optics, ultrashort laser pulses also exhibit chirp, which, in optical transmission systems interacts with the dispersion properties of the materials, increasing or decreasing total pulse dispersion as the signal propagates.

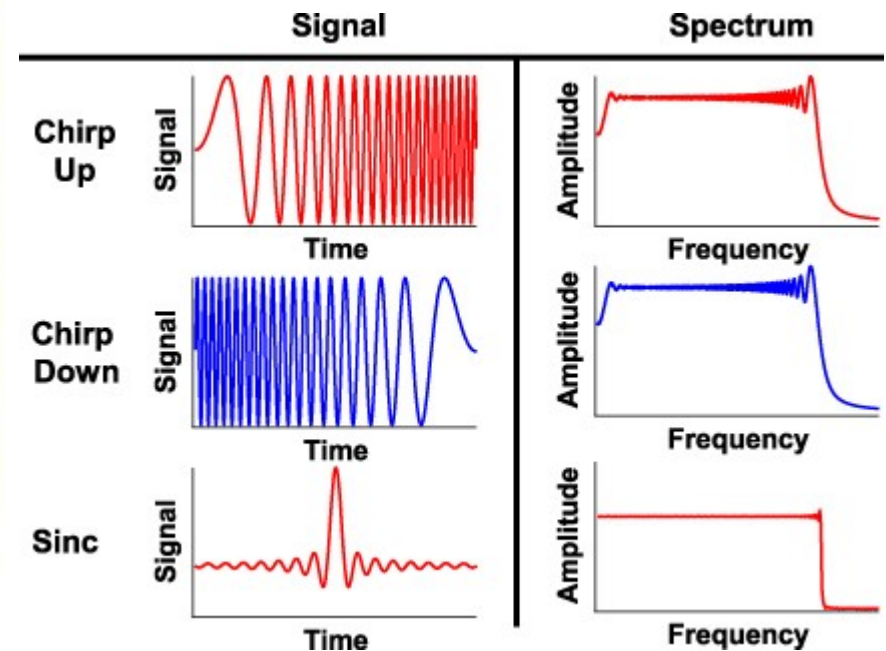
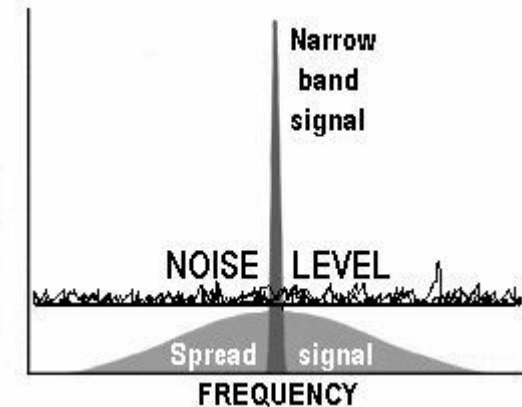
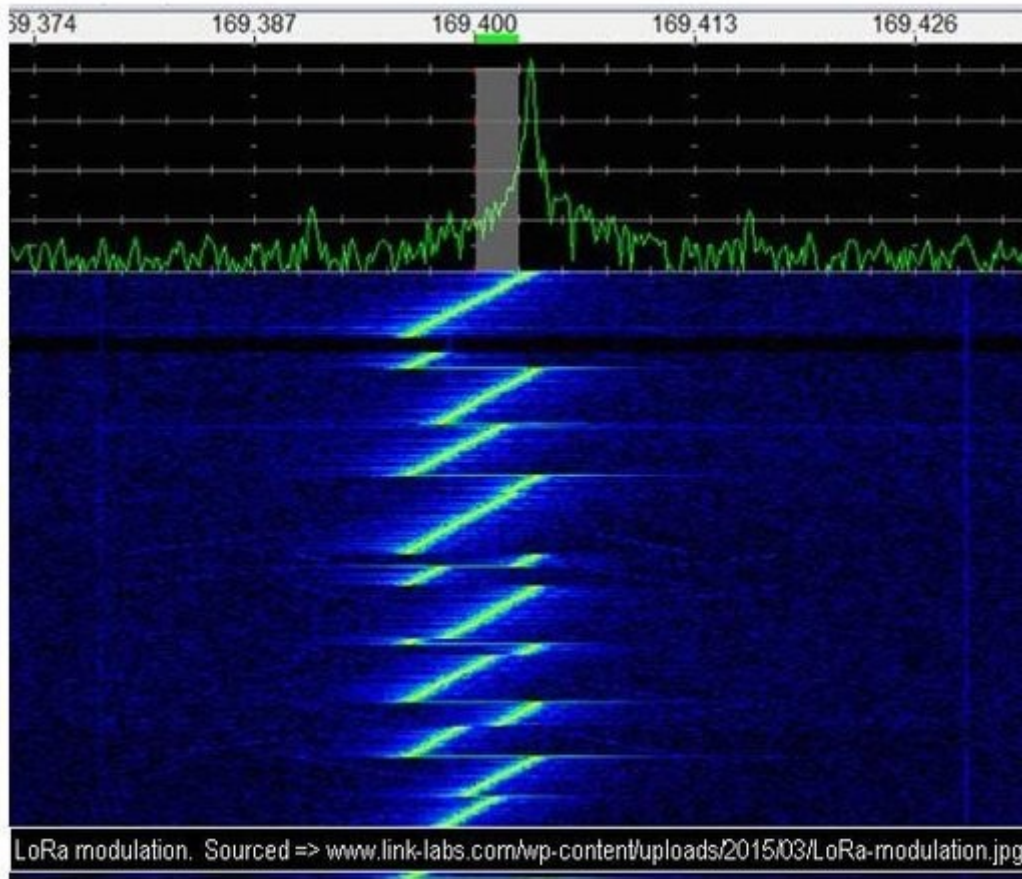
<https://www.slideshare.net/ajal4u/chirps>

Modulació CHIRP

Simulated chirp signal



Modulació CHIRP



The Things Network

https://www.thethingsnetwork.org

70%

Cerca



[COMMUNITIES](#) [LABS](#) [LEARN](#) [SUPPORT](#) [FORUM](#) [SHOP](#)

[SIGN UP](#)

[LOGIN](#)



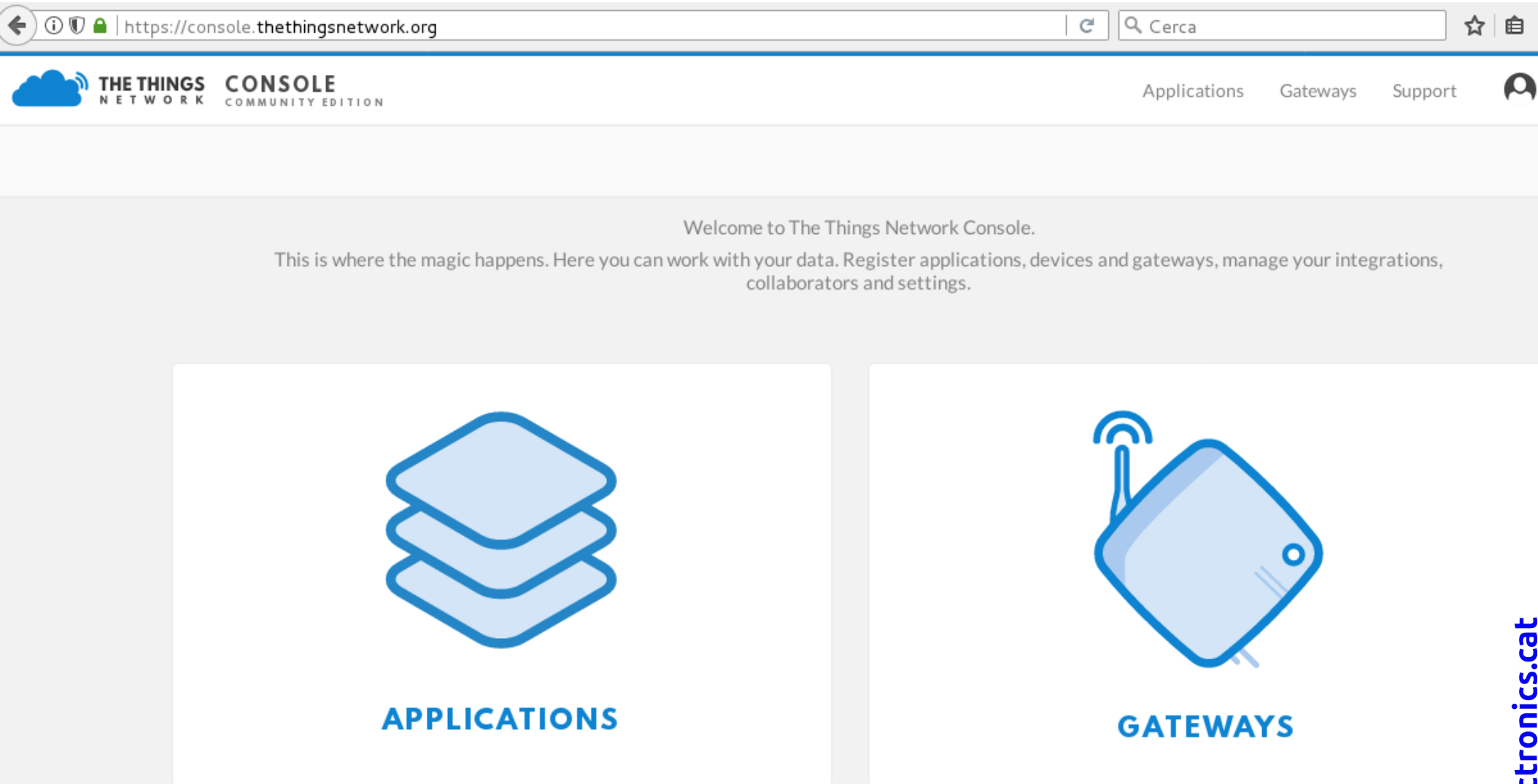
**BUILDING A GLOBAL INTERNET OF
THINGS NETWORK TOGETHER.**

[Learn More →](#)

**BUILDING A FULLY DISTRIBUTED INTERNET OF THINGS DATA
INFRASTRUCTURE.**

<https://www.thethingsnetwork.org/>

La cònsola de The Things Network



Nova aplicació a The Things Network

← ⓘ 🔒 | https://console.thethingsnetwork.org/applications | 🔍 Cerca ☆ 📄

 **THE THINGS NETWORK** **CONSOLE** COMMUNITY EDITION Applications Gateways Support 👤

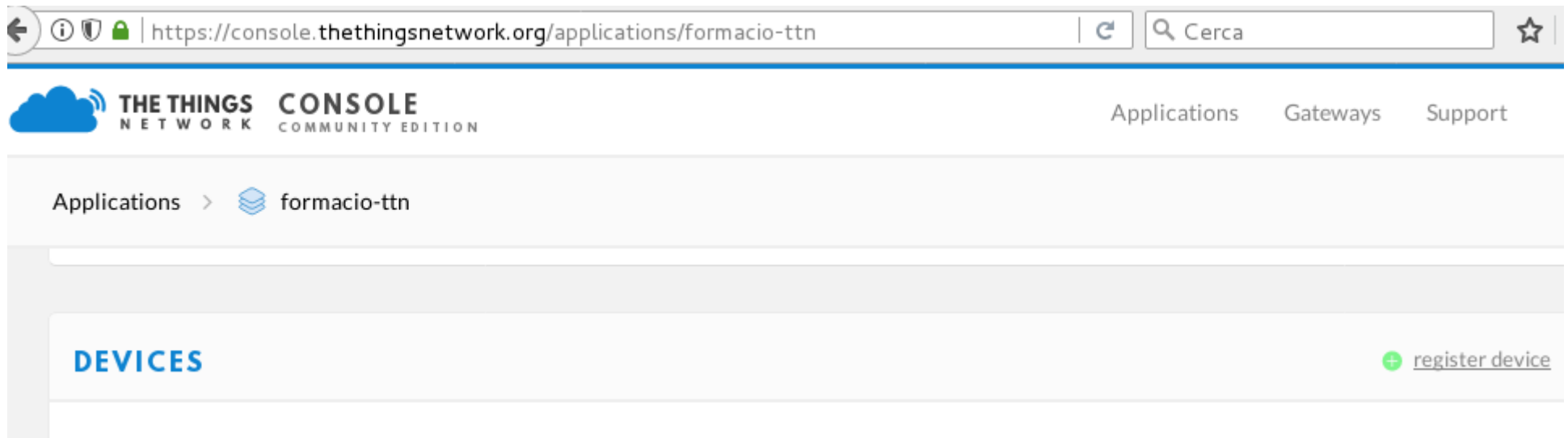
Applications

APPLICATIONS [+ add application](#)

APPLICATIONS [+ add application](#)

formacio-ttn	Formació sobre The Things Network	ttn-handler-eu	70 B3 D5 7E
---------------------	-----------------------------------	----------------	-------------

Registre de dispositiu a l'aplicació a The Things Network



Registre de dispositiu a l'aplicació a The Things Network (TTN)

← ⓘ 🔒 | https://console.thethingsnetwork.org/applications/formacio-ttn/devices/register | 🔍 Cerca ☆ 📄

Applications >  formacio-ttn > Devices

Overview

Devices

Payload Formats

Integrations

Data

Settings

REGISTER DEVICE

[bulk import devices](#)

Device ID

This is the unique identifier for the device in this app. The device ID will be immutable.

esp32-rfm95-07 

Device EUI

The device EUI is the unique identifier for this device on the network. You can change the EUI later.

 65 43 61 74 30 30 30 37  8 bytes

App Key

The App Key will be used to secure the communication between you device and the network.



this field will be generated

Cancel

Register

Registre de dispositiu a l'aplicació a TTN

Applications >  formacio-ttn > Devices >  esp32-rfm95-07

DEVICE OVERVIEW

Application ID formacio-ttn

Device ID esp32-rfm95-07

Activation Method OTAA

Device EUI <> ↔ 65 43 61 74 3 00 00 00 📋

Application EUI <> ↔ 70 B3 D5 7E D0 00 00 00 📋

App Key <> ↔ 👁 📋

Status ● never seen

Frames up 0 [reset frame counters](#)

Frames down 0

OTAA i ABP

OTAA (Over The Air Authentication): Aquest mètode consisteix en que un node prova de connectar-se a una passarel·la LoRaWAN emprant les claus **APPEUI** i **APPKEY**. Si les claus són correctes, la passarel·la contestarà al node amb un missatge d'acceptació (join) i des d'aquest moment el node pot rebre i trametre des de la passarel·la.

ABP (Authentication By Personalisation): L'autenticació per personalització consisteix en que les claus de xifratge es configuren manualment al dispositiu, permetent la comunicació directa a les passarel·les LoRaWAN sense necessitat de demanar permís.

Registre de dispositiu a l'aplicació a TTN

Applications >  formacio-ttn > Devices >  esp32-rfm95-07 > Settings

DEVICE SETTINGS

[General](#)

[Location](#)

SETTINGS

Description
A human-readable description of the device
 ✓

Device EUI
The serial number of your radio module, similar to a MAC address

 65 43 61 74 30 30 30 37 ✓ 8 bytes

Application EUI
 ⇅

Activation Method

☐ OTAA ☒ ABP

Device Address

Registre de dispositiu a l'aplicació a TTN

Applications > formacio-ttn > Devices > esp32-rfm95-07 > Settings

Activation Method

☐ OTAA ☒ ABP

Device Address

The device address will be assigned by the network server

Network Session Key

Network Session Key will be generated

App Session Key

App Session Key will be generated

Frame Counter Width

☐ 16 bit ☒ 32 bit

☐ **Frame Counter Checks**

Disabling frame counter checks drastically reduces security and should only be used for development purposes

Delete Device Cancel Save

Registre de dispositiu a l'aplicació a TTN

Applications >  formacio-ttn > Devices >  esp32-rfm95-07

DEVICE OVERVIEW

Application ID formacio-ttn

Device ID esp32-rfm95-07

Description Placa ESP32-LoRa-07 (ESP32 i placa RFM95)

Activation Method ABP

Device EUI <> ↔ 65 43 61 74 3 00000000 📋

Application EUI <> ↔ 70 B3 D5 7E D 00000003 📋

Device Address <> ↔ 26 01 00000 📋

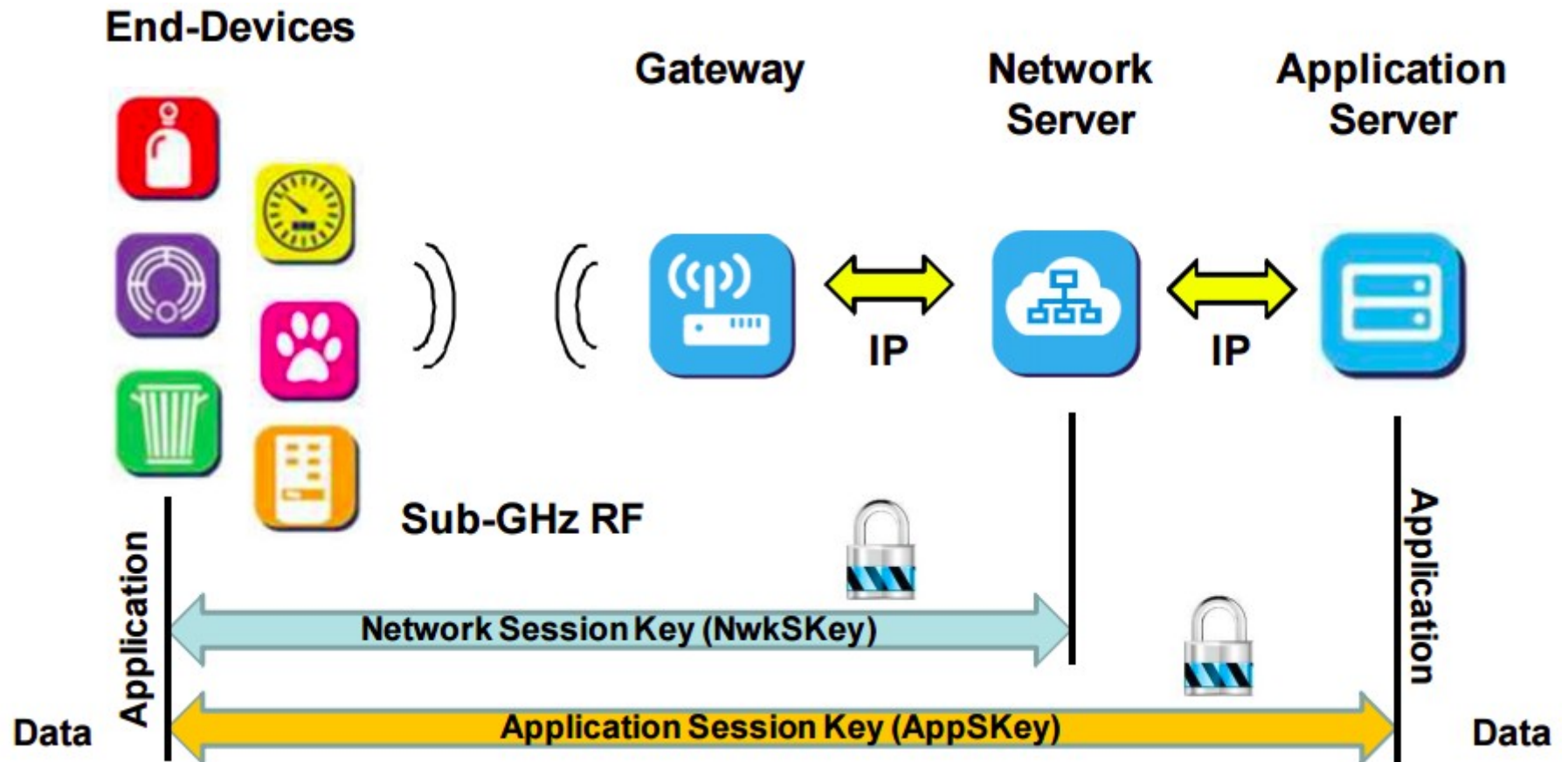
Network Session Key <> ↔ 👁 📋

App Session Key <> ↔ 👁 📋

Status ● never seen

Frames up 0 [reset frame counters](#)

Frames down 0



Registre de dispositiu a l'aplicació a TTN

Applications > formacio-ttn > Devices > esp32-rfm95-07

DEVICE OVERVIEW

Application ID formacio-ttn

Device ID esp32-rfm95-07

Description Placa ESP32-LoRa-07 (ESP32 i placa RFM95)

Activation Method ABP

Device EUI <> ⇅ 65 43 61 74 3 [icon]

Application EUI <> ⇅ 70 B3 D5 7E [icon]

Device Address <> ⇅ 26 01 1 [icon]

Network Session Key <> ⇅ [icon] msb { 0xB7, 0x30, 0xAF, 0x7B, 0xEE, 0x7A, 0x20, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00 } [icon]

App Session Key <> ⇅ [icon] msb { 0x12, 0x5C, 0x41, 0x0C, 0x7E, 0x30, 0x8F, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00 } [icon]

Status ● never seen

Frames up 0 [reset frame counters](#)

Frames down 0

Registre de dispositiu a l'aplicació a TTN

lorawan_ttnEsp32_rfm95_01_multichannel-esp32-rfm95-07 - credentials.h | Arduino 1.8.5

Fitxer Edita Esbós Eines Ajuda



lorawan_ttnEsp32_rfm95_01_multichannel-esp32-rfm95-07

credentials.h

pinMapping.h

```
#define LORA_MODE_ABP 0
#define LORA_MODE_OTAA 1

#define LORA_MODE LORA_MODE_ABP

#if LORA_MODE == LORA_MODE_ABP
    static const PROGMEM u1_t NWKSKEY[16] = { 0xB7, 0x30, 0xAF, 0x7B, 0xEE, 0x7A, 0x20, 0x11, 0x11, 0x11, 0x11, 0x11, 0x11, 0x11, 0x11, 0x11 };
    static const u1_t PROGMEM APPSKEY[16] = { 0x12, 0x5C, 0x41, 0x0C, 0x7E, 0x30, 0x8F, 0x11, 0x11, 0x11, 0x11, 0x11, 0x11, 0x11, 0x11, 0x11 };
    static const u4_t DEVADDR = 0x26011111 ;
#elseif
// De moment no és operatiu a aquesta versió
const char *appEui = "";
const char *appKey = "";
#endif
```

Uplink des del node a l'aplicació a TTN

Applications > formacio-ttn > Devices > esp32-rfm95-07 > Data

APPLICATION DATA

Filters: uplink downlink activation ack error

	time	counter	port	
▲	11:12:40	1	1	payload: 00 00 00 01
▲	11:11:57	0	1	retry payload: 00 00 00 00

Fitxer Edita Esbós Eines Ajuda

lorawan_ttnEsp32_rfm95_01_multichar

```
#define LORA_MODE_ABP 0
#define LORA_MODE_OTAA 1

#define LORA_MODE LORA_MODE_ABP

#if LORA_MODE == LORA_MODE_ABP
static const PROGMEM u1_t NWKSKEY[]
static const u1_t PROGMEM APPSKEY[]
static const u4_t DEVADDR = 0x2601
#endif

#if LORA_MODE == LORA_MODE_OTAA
// De moment no és operatiu a aql
const char *appEui = "";
const char *appKey = "";
```

```
[MAIN] Startup
[RFM95] Packet queued
135590: [RFM95] EV_TXCOMPLETE (includes waiting for RX windows)
TXRX_ACK confirmed UP frame was acked

TX complete .....
0 0

[RFM95] Packet queued
2839919: [RFM95] EV_TXCOMPLETE (includes waiting for RX windows)
TXRX_ACK confirmed UP frame was acked

TX complete .....
0 0
```

Downlink des de l'aplicació TTN al node

Applications >  formacio-ttn > Devices >  esp32-rfm95-07

DOWNLINK

Scheduling

[replace](#) [first](#) [last](#)

FPort

1  

☒ **Confirmed**

Payload

[bytes](#) [fields](#)

01 65 43 61 74 02  6 bytes

[Send](#)

Downlink des de l'aplicació TTN al node

Applications > formacio-ttn > Devices > esp32-rfm95-07 > Data

APPLICATION DATA

Filters: uplink downlink activation ack error

	time	counter	port	
▲	11:16:54	6	0	payload: [not provided]
✓	11:16:54		1	confirmed ack appid: formacio-ttn
▼	11:16:49		1	confirmed payload: 0165 43 61 74 02
▲	11:16:49	5	1	payload: 0000 00 05
▼	11:15:47		1	confirmed payload: 0165 43 61 74 02
▲	11:15:47	4	1	payload: 0000 00 04
▼	11:15:13		1	scheduled confirmed payload: 0165 43 61 74 02
▲	11:14:45	3	1	payload: 0000 00 03

```
#define LORA_MODE_OTAA 1
#define LORA_MODE LORA_MODE_ABP

#if LORA_MODE == LORA_MODE_ABP
    static const PROGMEM u1_t MNKSKEY[
    static const u1_t PROGMEM APPSKEY[
    static const u4_t DEVADDR = 0x2601;
#endif

#if LORA_MODE == LORA_MODE_OTAA
    // De moment no és operatiu a aq
    const char *appEui = "";
    const char *appKey = "";
#endif

[RFM95] Packet queued
[RFM95] Pending message
18295111: [RFM95] EV_TXCOMPLETE (incl
TXRX_ACK confirmed UP frame was acked
TX complete .....
6 1
1654361742
[RFM95] Received 6 bytes of payload
01 65 43 61 74 02
18678004: [RFM95] EV_TXCOMPLETE (incl
TXRX_ACK confirmed UP frame was acked
TX complete .....
0 0
```

Adaptació del node de multicanal a monocanal

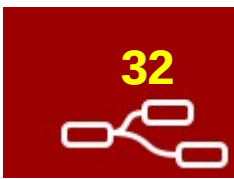
```
LMIC_setupChannel(0, 868100000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENTI); // g-band
/*
LMIC_setupChannel(1, 868300000, DR_RANGE_MAP(DR_SF12, DR_SF7B), BAND_CENTI); // g-band
LMIC_setupChannel(2, 868500000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENTI); // g-band
LMIC_setupChannel(3, 867100000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENTI); // g-band
LMIC_setupChannel(4, 867300000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENTI); // g-band
LMIC_setupChannel(5, 867500000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENTI); // g-band
LMIC_setupChannel(6, 867700000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENTI); // g-band
LMIC_setupChannel(7, 867900000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENTI); // g-band
LMIC_setupChannel(8, 868800000, DR_RANGE_MAP(DR_FSK, DR_FSK), BAND_MILLI); // g2-band
*/
LMIC_setupChannel(1, 868100000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENTI); // g-band
LMIC_setupChannel(2, 868100000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENTI); // g-band
LMIC_setupChannel(3, 868100000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENTI); // g-band
LMIC_setupChannel(4, 868100000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENTI); // g-band
LMIC_setupChannel(5, 868100000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENTI); // g-band
LMIC_setupChannel(6, 868100000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENTI); // g-band
LMIC_setupChannel(7, 868100000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENTI); // g-band
LMIC_setupChannel(8, 868100000, DR_RANGE_MAP(DR_SF12, DR_SF7), BAND_CENTI);
```

Enllaç al [codi multicanal](#)
Enllaç al [codi monocanal](#)
Biblioteca [arduino-lmic](#)

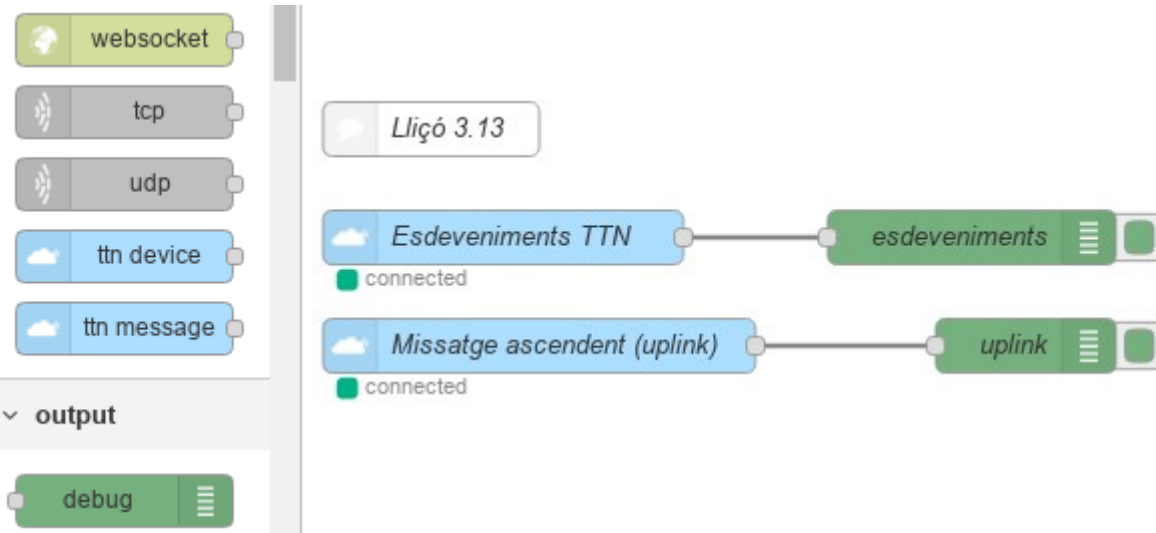


Node-RED

LoRaWAN



Nodes 'ttn device' i 'ttn message'



Edit debug node

Delete Cancel Done

Output complete msg object

to debug tab

Name esdeveniments

Edit ttn device node

Delete Cancel Done

Name Esdeveniments TTN

App formacio-ttn

Device ID

Event activations

Edit ttn message node

Delete Cancel Done

Name Missatge ascendent (uplink)

App formacio-ttn

Device ID

Field

Codi Node-RED: https://binefa.cat/IoT/nodeRed/03_nodeRed_13.txt

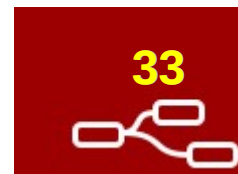
Codi d'Arduino IDE per a l'ESP32: https://binefa.cat/IoT/esp32/lorawan_ttnEsp32_rfm95_01_multichannel-llico_3_13.tar.gz



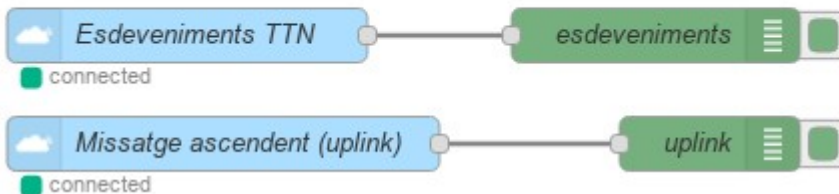
Node-RED

LoRaWAN

App ID



Lliçó 3.13



Applications > formacio-ttn

ttn device > Edit ttn app node

Delete

Cancel

Update

App ID formacio-ttn

Region or Broker eu

Access Key

Overview

APPLICATION OVERVIEW

Application ID formacio-ttn

Description Formació sobre The Things Network

Created 7 days ago

Handler ttn-handler-eu (current handler)

Codi Node-RED: https://binefa.cat/loT/nodeRed/03_nodeRed_13.txt

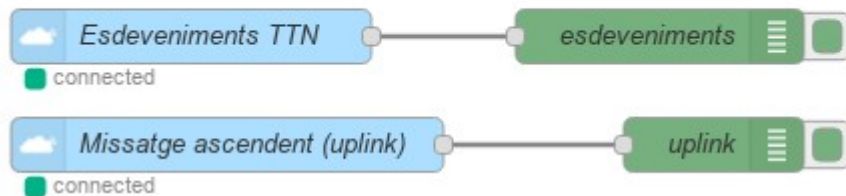
Codi d'Arduino IDE per a l'ESP32: https://binefa.cat/loT/esp32/lorawan_ttnEsp32_rfm95_01_multichannel-llico_3_13.tar.gz

Node-RED

LoRaWAN

Access Key

Lliçó 3.13



ttn device > Edit ttn app node

Delete Cancel Update

App ID formacio-ttn

Region or Broker eu

Access Key

Applications > formacio-ttn

ACCESS KEYS

[manage keys](#)

default key

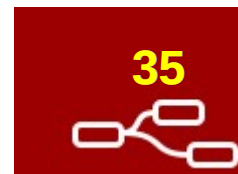
devices messages

..... base64

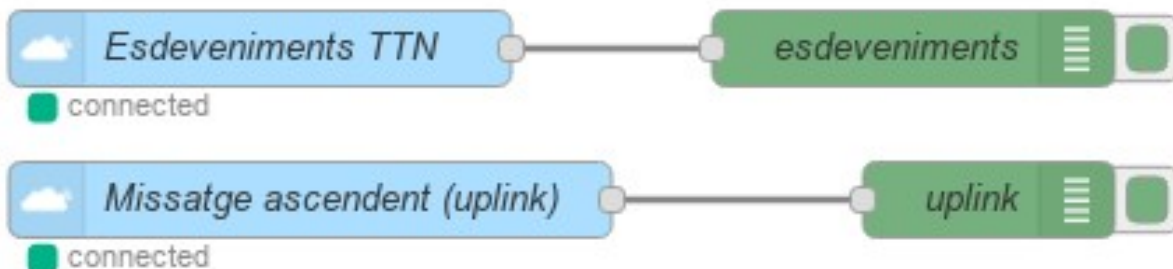


Node-RED

LoRaWAN



Lliçó 3.13



```
/dev/ttyUSB0
```

```
[RFM95] Packet queued
160030409: [RFM95] EV_TXCOMPLETE (includes waiting for RX windows)
TXRX_ACK confirmed UP frame was acked

TX complete .....
0 0

[RFM95] Packet queued
161409360: [RFM95] EV_TXCOMPLETE (includes waiting for RX windows)
TXRX_ACK confirmed UP frame was acked

TX complete .....
0 0
```

☒ Desplaçament automàtic Ambdós NL & CR

```
29/3/2018, 20:59:56 node: uplink
msg: Object
{ app_id: "formacio-ttn", dev_id: "ttgo-01", hardware_serial:
"6543617430303031", port: 1, counter: 38 }

29/3/2018, 21:00:58 node: uplink
msg: Object
object
  app_id: "formacio-ttn"
  dev_id: "ttgo-01"
  hardware_serial: "6543617430303031"
  port: 1
  counter: 39
  payload_raw: buffer[4] raw
    0: 0
    1: 0
    2: 0
    3: 39
  metadata: object
    time: "2018-03-29T19:00:58.270307023Z"
    frequency: 867.9
    modulation: "LORA"
    data_rate: "SF7BW125"
    airtime: 51456000
    coding_rate: "4/5"
    gateways: array[1]
      0: object
        gtw_id: "eui-b827ebfffe875de4"
        timestamp: 4290676260
        time: "2018-03-29T19:00:58.247289Z"
        channel: 7
        rssi: -48
        snr: 9.5
```

Codi Node-RED: https://binefa.cat/loT/nodeRed/03_nodeRed_13.txt

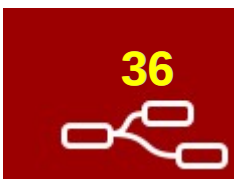
Codi d'Arduino IDE per a l'ESP32: https://binefa.cat/loT/esp32/lorawan_ttnEsp32_rfm95_01_multichannel-llico_3_13.tar.gz



Node-RED

LoRaWAN

Missatge descendent (*downlink*)



Lliçó 3.14

Esdeveniments TTN connected esdeveniments

Missatge ascendent (uplink) connected uplink



```
return {
  dev_id: msg.dev_id,
  port: msg.port,
  schedule: "replace",
  confirmed: false,
  payload: Buffer.from([msg.payload ? 0x71 : 0x51])
}
```

0101 0000	80	50	P	0111 0000	112	70	p
0101 0001	81	51	Q	0111 0001	113	71	q
0101 0010	82	52	R	0111 0010	114	72	r

29/3/2018, 21:30:26 node: uplink

msg : Object

▶ { app_id: "formacio-ttn", dev_id: "ttgo-01", hardware_serial: "6543617430303031", port: 1, counter: 74 ... }

29/3/2018, 21:30:42 node: uplink

msg : Object

/dev/ttyUSBO

TX complete

1 1

51

[RFM95] Received 1 bytes of payload

51

[RFM95] Packet queued

252489096: [RFM95] EV_TXCOMPLETE (includes waiting for RX windows)

TXRX_ACK confirmed UP frame was acked

TX complete

1 1

71

[RFM95] Received 1 bytes of payload

71

☒ Desplaçament automàtic

Ambdós NL & CR

Codi Node-RED: https://binefa.cat/loT/nodeRed/03_nodeRed_14.txt

Codi d'Arduino IDE per a l'ESP32: https://binefa.cat/loT/esp32/lorawan_ttnEsp32_rfm95_01_multichannel-llico_3_13.tar.gz



Node-RED



LoRaWAN / Missatge descendent (*downlink*)



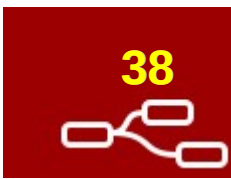
Applications > formacio-ttn > Devices > ttgo-01 > Data				
APPLICATION DATA				
Filters	uplinkdownlinkactivationackerror			
	time	counter	port	
	▲ 21:34:49	79	1	payload: 0000004E
	▲ 21:33:47	78	1	payload: 0000004D
	▲ 21:32:45	77	1	payload: 0000004C
	▲ 21:31:43	76	1	payload: 0000004B
	▼ 21:30:42		1	payload: 71
	▲ 21:30:42	75	1	payload: 0000004A
	▼ 21:30:39		1 scheduled	payload: 71
	▼ 21:30:26		1	payload: 51
	▲ 21:30:26	74	1	payload: 00000049
	▼ 21:30:17		1 scheduled	payload: 51
	▲ 21:29:45	73	1	payload: 00000048



Node-RED

LoRaWAN

Missatge descendent (downlink)



Edit function node

Delete Cancel Done

Name cap a TTN

Function

```
1 - return {  
2   dev_id: msg.dev_id,  
3   port: msg.port,  
4   schedule: "replace",  
5   confirmed: false,  
6   payload: Buffer.from([msg.payload ? 0x71 : 0x51])  
7 }
```

Edit inject node

Delete Cancel Done

Payload 0

Topic

Repeat none

☐ Inject once at start?

Name 0

0101 0000	80	50	P	0111 0000	112	70	p
0101 0001	81	51	Q	0111 0001	113	71	q
0101 0010	82	52	R	0111 0010	114	72	r

DEVICE OVERVIEW

Application ID formacio-ttn

Device ID ttgo-01

Description Placa TTGO-01

Activation Method ABP

Edit ttn send node

Delete Cancel Done

Name Missatge descendent

App formacio-ttn

Device ID ttgo-01

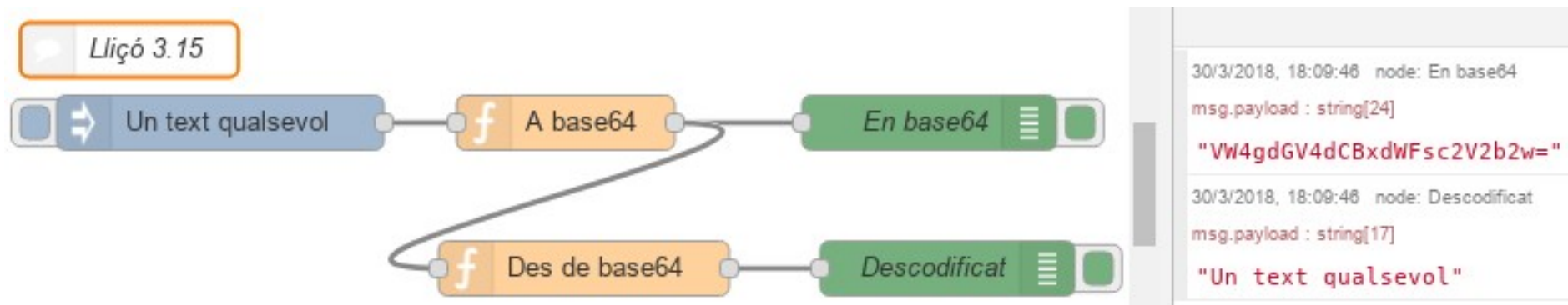
Port 1

Codi Node-RED: https://binefa.cat/loT/nodeRed/03_nodeRed_14.txt

Codi d'Arduino IDE per a l'ESP32: https://binefa.cat/loT/esp32/lorawan_ttnEsp32_rfm95_01_multichannel-llico_3_13.tar.gz

Node-RED

Codificació i descodificació en base64



Edit function node

Delete Cancel Done

Name A base64

Function

```
1 var b = new Buffer(msg.payload);
2 msg.payload = b.toString('base64');
3
4 return msg;
```

Edit function node

Delete Cancel Done

Name Des de base64

Function

```
1 var b = new Buffer(msg.payload, 'base64');
2 msg.payload = b.toString();
3
4 return msg;
```

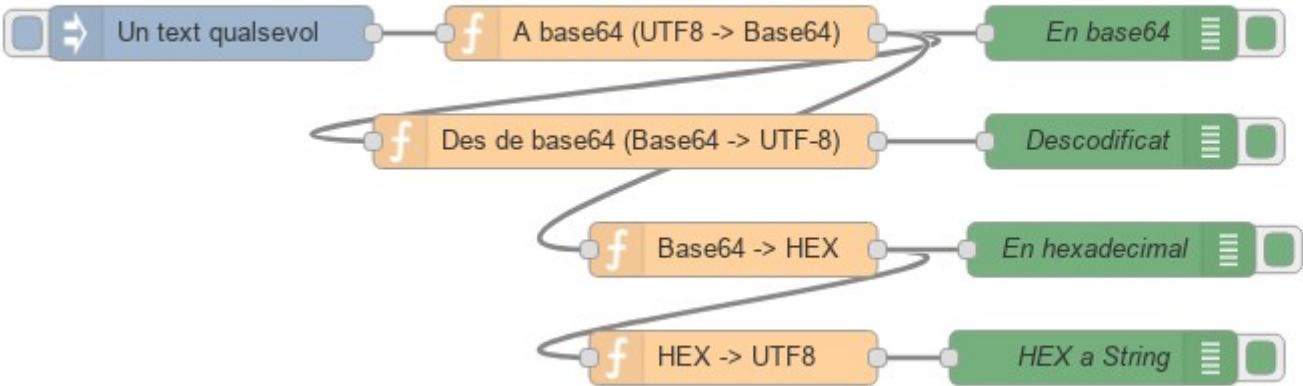


Node-RED

Codificació i descodificació en base64

Presentació en hexadecimal

Lliçó 3.16



```
30/3/2018, 18:35:45 node: En base64
msg.payload : string[24]
"VW4gdGV4dCBxdWFS c2V2b2w="

30/3/2018, 18:35:45 node: Descodificat
msg.payload : string[17]
"Un text qualsevol"

30/3/2018, 18:35:45 node: HEX a String
msg.payload : string[17]
"Un text qualsevol"

30/3/2018, 18:35:45 node: En hexadecimal
msg.payload : string[34]
"556e20746557874207175616c736576666c"
```

Edit function node

Delete Cancel Done

Name Base64 -> HEX

Function

```
1 var b = new Buffer(msg.payload, 'base64')
2 msg.payload = b.toString('hex');
3
4 return msg;
```

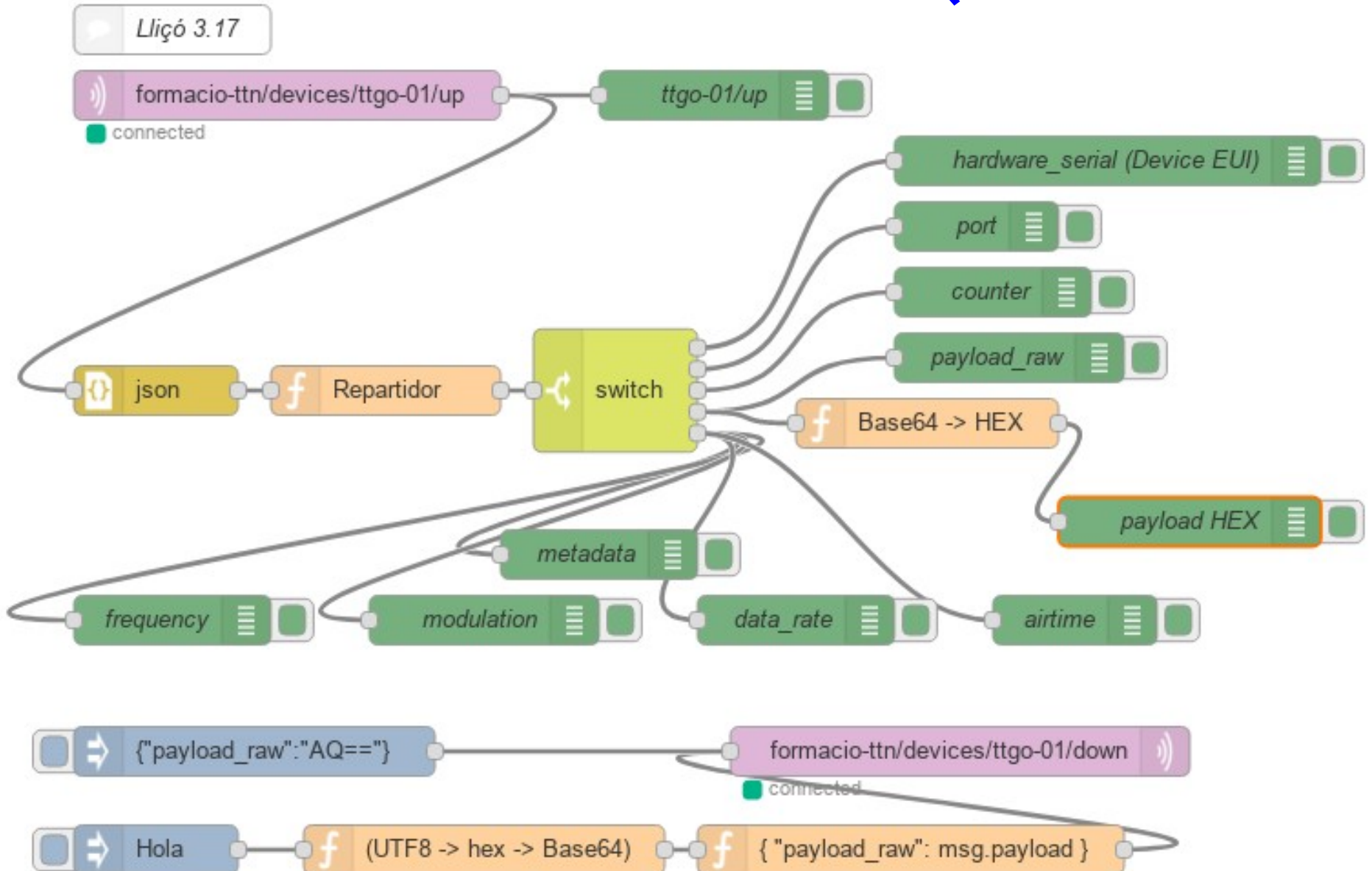
Edit function node

Delete Cancel Done

Name HEX -> UTF8

Function

```
1 var b = new Buffer(msg.payload, 'hex')
2 msg.payload = b.toString('utf8');
3
4 return msg;
```





Node-RED

LoRaWAN

TTN fent servir MQTT



Edit mqtt in node

Delete

Cancel

Done

Server

eu.thethings.network:1883

Topic

formacio-ttn/devices/ttgo-01/up

QoS

2

Name

Name

Edit mqtt out node

Delete

Cancel

Done

Server

eu.thethings.network:1883

Topic

formacio-ttn/devices/ttgo-01/down

QoS

Retain

Name

Name

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

mqtt in > Edit mqtt-broker node

Delete

Cancel

Update

Connection

Security

Birth Message

Will Message

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

mqtt in > Edit mqtt-broker node

Delete

Cancel

Update

Connection

Security

Birth Message

Will Message

Username

formacio-ttn

Password

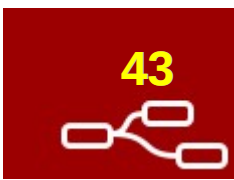
.....



Node-RED

LoRaWAN

TTN fent servir MQTT



Edit function node

Delete Cancel Done

Name Repartidor

Function

```
1 var keys = Object.keys(msg.payload);
2
3 var msgs = keys.map(function(key) {
4   return { topic: key, payload: msg.payload[key] };
5 });
6 return [msgs];
```

Edit function node

Delete Cancel Done

Name (UTF8 -> hex -> Base64)

Function

```
1 var b = new Buffer(msg.payload);
2 msg.payload = b.toString('base64');
3
4 return msg;
```

Edit function node

Delete Cancel Done

Name { "payload_raw": msg.payload }

Function

```
1 msg.payload = { "payload_raw": msg.payload };
2
3 return msg;
```

Edit switch node

Delete Cancel Done

Name

Property msg.topic

==	hardware_serial	→ 1	x
==	port	→ 2	x
==	counter	→ 3	x
==	payload_raw	→ 4	x
==	metadata	→ 5	x

sigfox

Què és?

