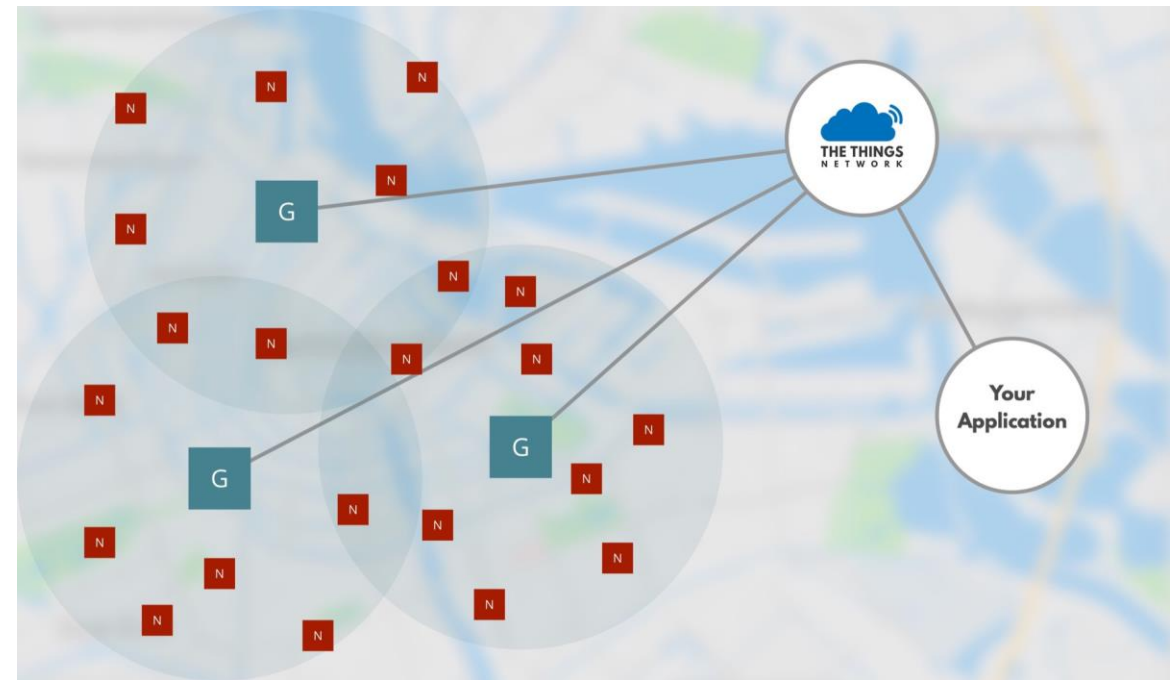


LoRaWan Register a device on TTN

F. Ferrero

LoRaWAN with The Thing Network

- The Things Network is a global, open, crowd-sourced Internet of Things data network.
- **The Things Network Backend** route messages from Nodes to the right Application, and back
- First, you have to register to <https://www.thethingsnetwork.org/> , when it is done, tell me your ID, I will add you as a collaborator on the Polytech' application
- You can also join the [Sophia Antipolis TTN community](#) :



Adding a new device

- Go to « Projets Polytech » application and register device
- For ID and EUI, use the N° 50ff1a00000000XX and just increment XX.
- To remember it : « 50ff1a » is for « SOPHIA »
- It will provide Device EUI, Application EUI and App Key

Activation Method

Device EUI

Application EUI

App Key

Activation by Personalization (ABP)

- Go to settings
- Select ABP and save
- Go back to Overview
- You have now the Device Address and the two 128 AES keys
- You can click on Hex-C Style to have the key in the right format

Activation Method

Device EUI 50 FF 1A 00 00 00 00 01

Application EUI 70 B3 D5 7E D0 00 A3 90

Device Address 26 01 11 44

Network Session Key msb { 0x14, 0x46, 0xEF, 0x00, 0x4B, 0xC9, 0x96, 0x95, 0xFE, 0x6A, 0x6F, 0x9F, 0xAC, 0x00, 0x00, 0x00, 0x00 }

App Session Key msb { 0x18, 0xF8, 0xAA, 0x61, 0x31, 0x2C, 0x6C, 0x05, 0x66, 0x4F, 0xCB, 0xE1, 0x81, 0x00, 0x00, 0x00, 0x00 }

Activation by Personalization (ABP)

- Go to my Github : https://github.com/FabienFerrero/UCA_Board
- Download the archive (.zip) and extract the archive
- Copy the file from Arduino_Code/Libraries/ to /Document/Arduino/Libraries/
- Open the code Arduino_Code/LORAWAN/ABP/Basic/UCA-ABP_Basic.ino
- Copy/Paste NWKSKY, APPSKY and DEVADDR with your IDs from TTN

```
// LoRaWAN NwkSKey, network session key
// This is the default Semtech key, which is used by the early prototype TTN
// network.
static const PROGMEM ul_t NWKSKY[16] = { 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00 };

// LoRaWAN AppSKey, application session key
// This is the default Semtech key, which is used by the early prototype TTN
// network.
static const ul_t PROGMEM APPSKY[16] = { 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00 };

// LoRaWAN end-device address (DevAddr)

static const u4_t DEVADDR = 0x00000000;
```

Activation by Personalization (ABP)

- Compile and download the code on your board
- Look at the TTN device overview
- Frames up should increment each minutes as you board is sending an uplink each 60s.
- Have look on data
- For each uplink, you can look many details as RSSI, SNR, airtime, modulation, coding rate, GW ID, etc ...

Status ● 25 seconds ago

Frames up 0 [reset frame counters](#)

Frames down 0

	time	counter	port	
▲	21:45:35	3	1	payload: 48 65 6C 6C 6F 2C 20 77 6F 72 6C 64 21
▲	21:44:29	2	1	payload: 48 65 6C 6C 6F 2C 20 77 6F 72 6C 64 21
▲	21:43:22	1	1	payload: 48 65 6C 6C 6F 2C 20 77 6F 72 6C 64 21
▲	21:42:16	0	1	retry payload: 48 65 6C 6C 6F 2C 20 77 6F 72 6C 64 21

Activation by Personalization (ABP)

Frame counter security

- Now reset you board (click on the red button on the Arduino mini pro)
- TTN is no more receiving the data
- Click on « reset frame counters » and reset you board again
- As you can see, frame counter is a security features to avoid replay attack (done by capturing and re-transmitting the messages)

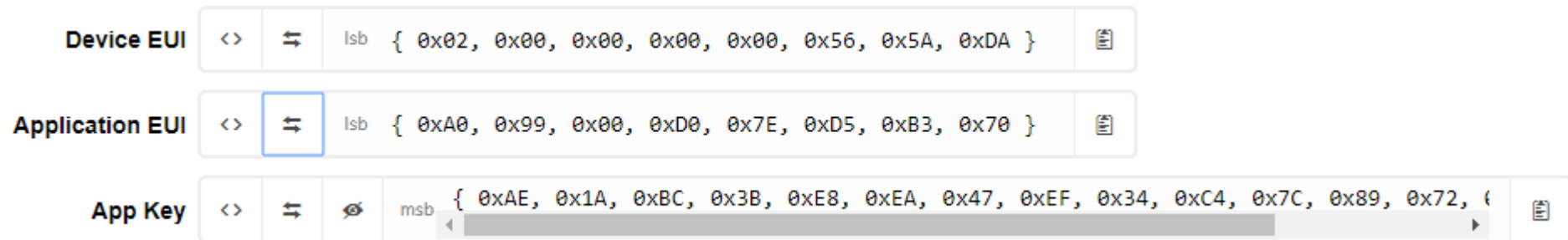
Downlink

- Open your serial monitor
- In TTN overview, go to downling, add a payload like « BABA » and click on send
- After the next uplink, you should see the number of byte received in downlink

```
16608545: EV_TXCOMPLETE (includes waiting for RX windows)
Received
2
bytes of payload
```

Over the Air Activation (OTAA)

- In TTN Settings of your device, select OTAA and save
- Open the code `Arduino_Code/LORAWAN/OTAA/LP_Basic/UCA-OTAA_Basic.ino`
- copy paste after clicking on hexa-style the DEV-EUI, APP-EUI and App Key
- Be carefull !!!
 - Device EUI and Application EUI are **lsb**
 - App Key is **msb**



Over the Air Activation (OTAA)

- Look in data
- You should see a first uplink that request the connection
- And a second packet with the first data
- On the serial monitor you can see the Joining process and then Joined and Tx.
- The device go to sleep after the Tx

COM18

```
Starting
Vbatt : 346.00
69428: EV_JOINING
745031: EV_JOINED
Datarate: SF8
Vbatt : 346.00
BV=346.00
PQ
1187429: EV_TXCOMPLETE (includes waiting for RX windows)
Datarate: SF8
Sleeping for 360 seconds = 45 x 8 + 0 x 4 + 0 x 2 + 0
```

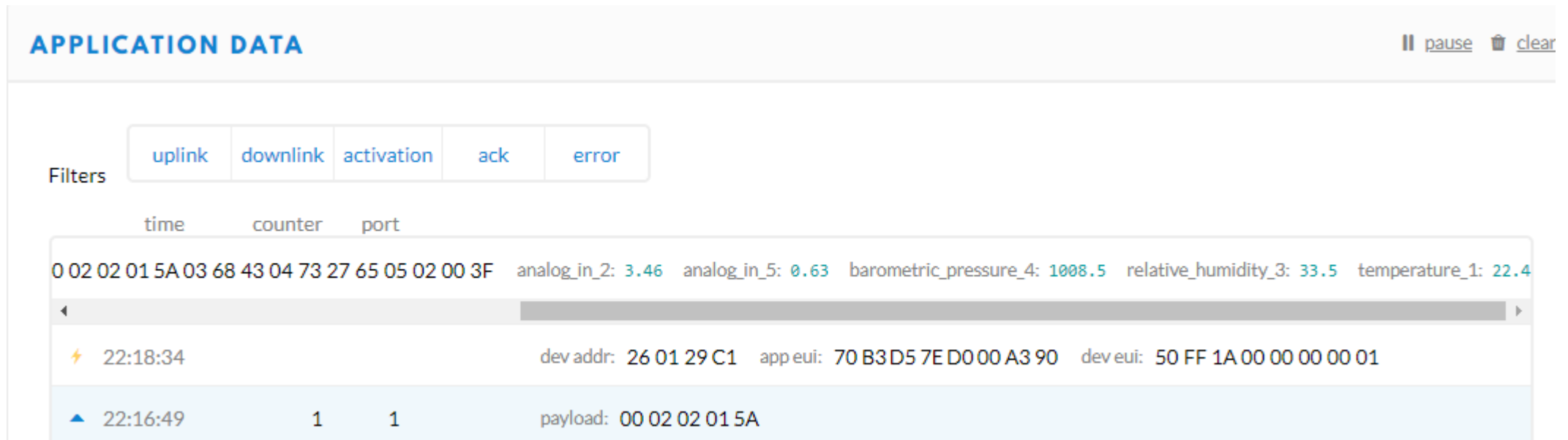
First packet
with data

Board
request for
connection

	time	counter	port	
→	22:10:21	0	1	payload: 00 02 02 01 5A
→	22:10:18			dev addr: 26 01 2A 04 appeui: 70 B3 D5 7E D0 00 A3 90 dev eui: 50 FF 1A 00 00 00 00 01

Over the Air Activation (OTAA) and data

- Try now the code
Arduino_Code/LORAWAN/OTAA/LP_BME280/UCA-BME280.ino
- It use the sensor BME280 that measure T°C, Humidity and Pressure
- The code is using [Cayenne LPP format](#)
- Now you can see sensor data in the uplink packet



The screenshot displays the 'APPLICATION DATA' section of a Cayenne LPP interface. At the top right, there are controls for 'pause' and 'clear'. Below this, a 'Filters' section contains buttons for 'uplink', 'downlink', 'activation', 'ack', and 'error'. The main data area shows a table with columns for 'time', 'counter', and 'port'. A specific data entry is highlighted in light blue, showing a time of 22:16:49, counter 1, and port 1. The payload for this entry is '00 02 02 01 5A'. Above the table, a detailed view of the payload is shown: '0 02 02 01 5A 03 68 43 04 73 27 65 05 02 00 3F', followed by sensor readings: 'analog_in_2: 3.46', 'analog_in_5: 0.63', 'barometric_pressure_4: 1008.5', 'relative_humidity_3: 33.5', and 'temperature_1: 22.4'. Below the detailed view, there is a lightning bolt icon and the time '22:18:34', along with device identifiers: 'dev addr: 26 01 29 C1', 'app eui: 70 B3D5 7E D000 A3 90', and 'dev eui: 50 FF 1A 00 00 00 00 01'.

time	counter	port
22:16:49	1	1

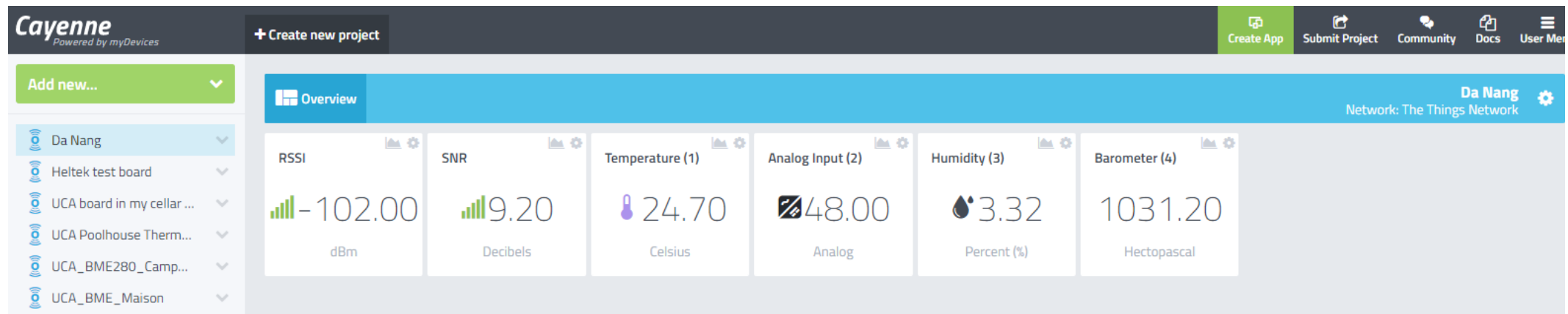
0 02 02 01 5A 03 68 43 04 73 27 65 05 02 00 3F analog_in_2: 3.46 analog_in_5: 0.63 barometric_pressure_4: 1008.5 relative_humidity_3: 33.5 temperature_1: 22.4

⚡ 22:18:34 dev addr: 26 01 29 C1 app eui: 70 B3D5 7E D000 A3 90 dev eui: 50 FF 1A 00 00 00 00 01

▲ 22:16:49 1 1 payload: 00 02 02 01 5A

Using Cayenne to see you data

- Go to <https://mydevices.com/> and create an account
- Add a device by selecting LoRa/TheThingNetwork and Cayenne LPP.
- Just add your device EUI
- You should see your data



Good luck for you projects !

