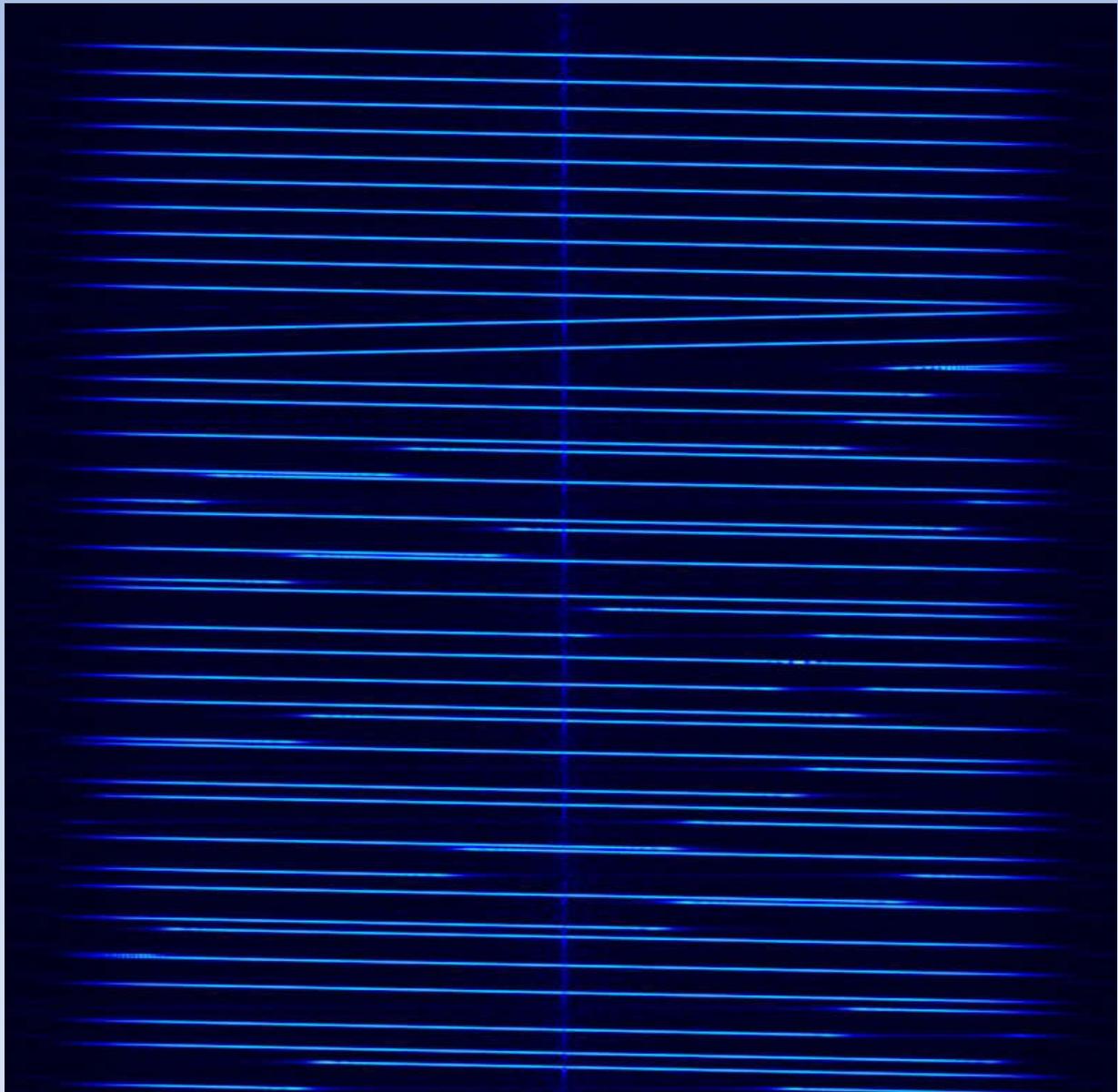


Présentation LoRa APRS 433MHz

433.775 MHz



T



125 kHz

Définition du LoRa

LoRa = **Long Range**

La technologie de modulation propriétaire LoRa est créée en 2009 par la start-up **Grenobloise Cycléo** (rachetée par

Semtech en 2012). Semtech est un fabricant des puces LoRa

Le LoRa est un protocole de communication pour l'Internet

des objets qui utilise une technique de **modulation par étalement de spectre** (chirp spread spectrum)

Bandes radio: 868MHz (EU) - 915MHz (US) - 433MHz (CN)

Principaux avantages du protocole LoRa

Sensibilité:

Décodage des datas en dessous du bruit de bande

Autonomie:

Faible consommation électrique

Coût:

Très peu onéreux

Débit:

Selon Semtech, LoRa devrait atteindre environ 7 à 10 dB plus de sensibilité au même débit de données que le FSK.

Il faut tout de même tenir du compte du fait qu'il faut une ligne de vue assez dégagée entre les 2 stations pour obtenir de bons résultats compte-tenu de la puissance employée (20dBm)

Applications Radioamateur (433MHz)

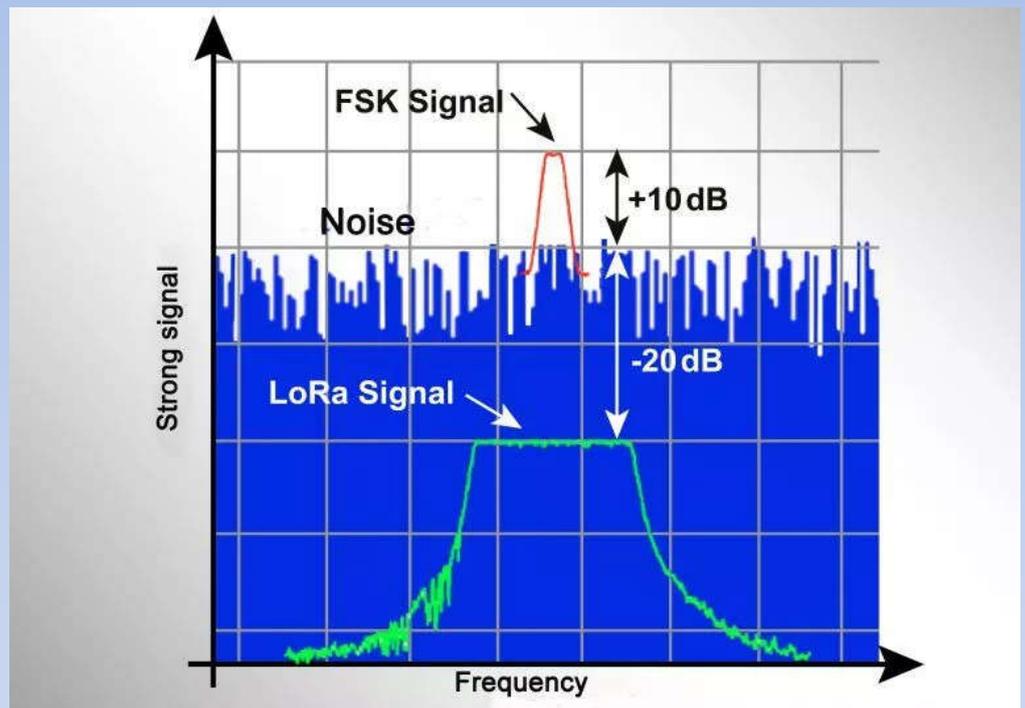
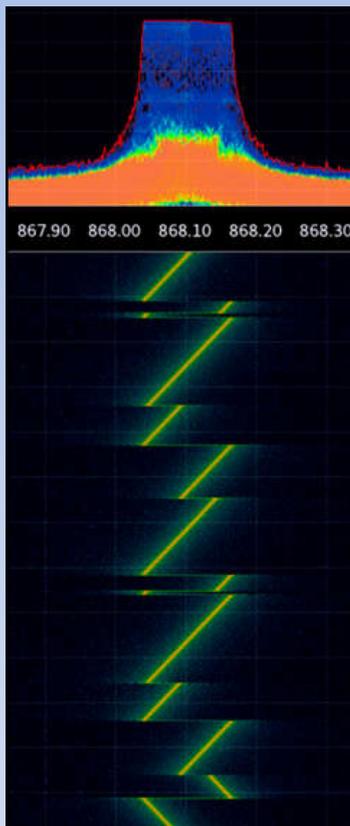
Système APRS alternatif au PACKET1200:

> Localisation de véhicules

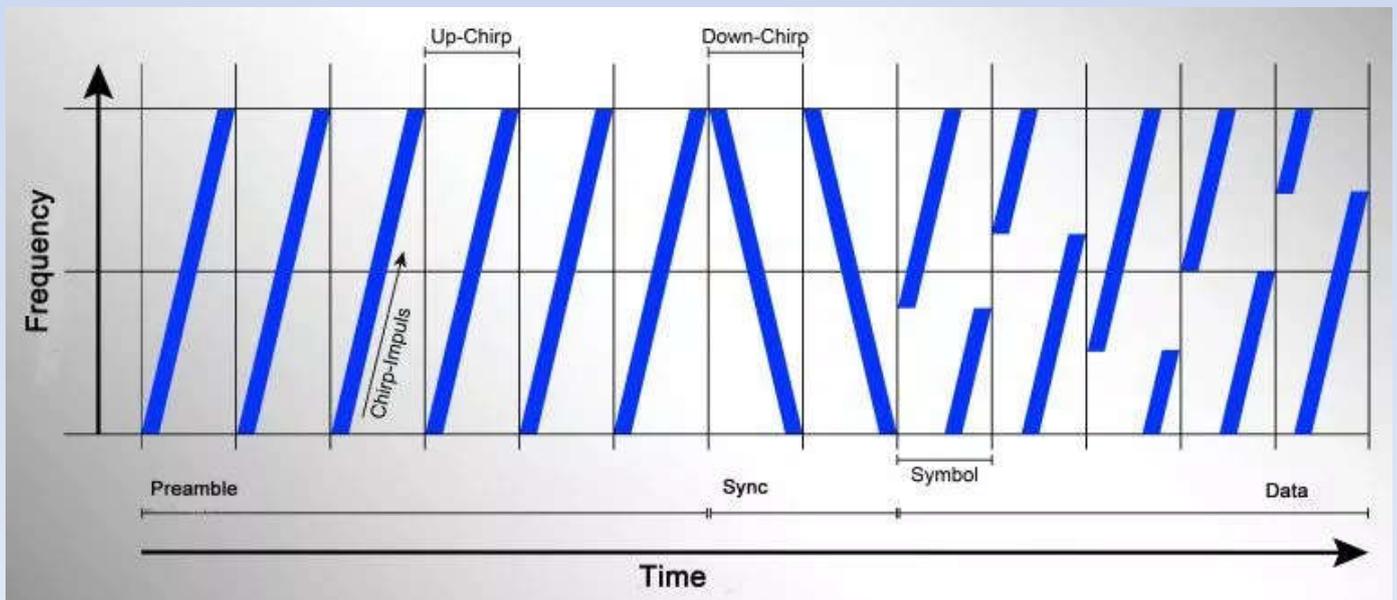
> Mesures de grandeurs dans une station ou un point haut (stations météo - niveau de tension/courant - charge batteries etc..)

Modulation par Etalement de Spectre

Le récepteur LoRa peut toujours recevoir et décoder avec succès un signal LoRa utile jusqu'à 20 dB en dessous du niveau de bruit, ce qui se traduit par une sensibilité du récepteur d'un maximum de -149 dBm.



Source: <https://www.mokosmart.com/fr/lora-frequency/>



Historique du développement HamRadio

Le développement initial des applicatifs radioamateurs utilisant le protocole LoRa vient d'Autriche.

par

Bernd/OE1ACM et **Christoph/OE1CGC**

Extrait traduit de leur site:

« La technologie LoRa a en fait été développée pour le transfert de données pour les appareils IOT dans les bandes ISM (LoraWAN). En raison de la large distribution et donc des grandes quantités, les coûts du matériel requis ont extrêmement baissé et sont abordables et extrêmement intéressants pour les projets du secteur amateur.

La version 433 MHz est particulièrement adaptée aux radioamateurs, car elle peut être utilisée par les amateurs dans la bande 70 cm sans les restrictions habituelles pour l'IOT dans la bande ISM (**rapport cyclique, puissance d'émission**, etc...).

Il y a quelque temps, un groupe d'intérêt s'est formé entre radioamateurs et amis dans le but de construire un réseau national LoRa afin d'échanger diverses données (localisation GPS, télémétrie, valeurs mesurées, messages, télécommande, ...) et ce, dans le réseau APRS existant »

Le code utilisé pour les expérimentations françaises vient de

Peter/OE5BPA ou de **Chris/OE3CJB**

https://github.com/lora-aprs/LoRa_APRS_iGate
<https://github.com/oe3cjb/TTGO-T-Beam-LoRa-APRS>

Produits

Module « Tracker » TTGO T-Beam (30€)

POWER

- 5V
- GND
- 3V3
- DCDC1
- GND
- 21 Wire_5DA
- 22 Wire_SCL
- Lora_ID1
- Lora_ID2

LORA

- TXD 601001 Serial TX
- RXD 601003 Serial RX
- 23 VSPI_MOSI
- 4 GPIO4 ADC10 TOUCH0
- 0 GPIO00 ADC11 TOUCH1
- GND

RST

- 5V
- GND

GPIO

- GPIO36 SVP
- GPIO39 SVN
- RST
- GPIO15 15
- GPIO35 35
- GPIO32 32
- GPIO33 33
- GPIO25 25
- GPIO14 14
- GPIO13 13
- GPIO02 2
- GND

ADC

- ADC0
- ADC3
- ADC13
- ADC7
- ADC4
- ADC5
- ADC18
- ADC16
- ADC14
- ADC12

TOUCH

- TOUCH3
- TOUCH9
- TOUCH8
- TOUCH6
- TOUCH4
- TOUCH2

DAC

- DAC1

I2C

- SCL 5
- SDA 26

SPI

- MOSI 27
- CS 18
- DIO 26
- RST 23
- MISO 19

NEO GPS

- TX 34
- RX 12

LED Introduction:

- LED1---Charging Indicator
- LED2---GPS Working Indicator

Power Introduction:

Programmable power supply

AXP192

18650 Battery

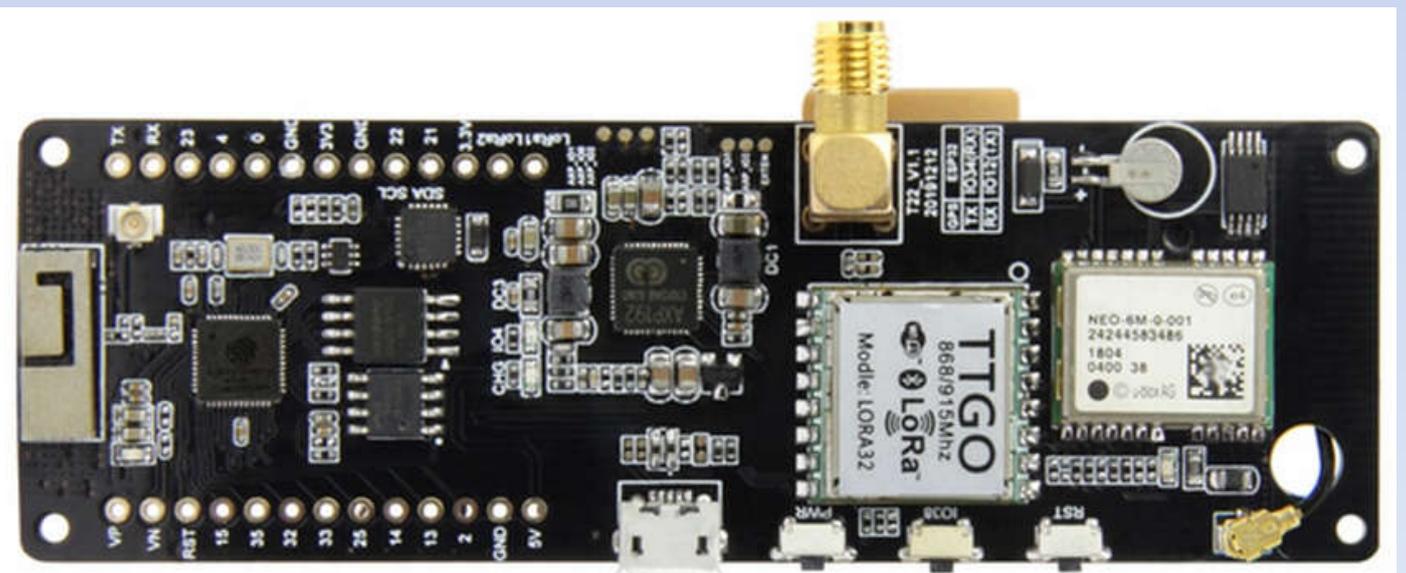
Legend:

- Name
- GND
- Other
- BAT
- ADC
- DAC
- SPI
- Touch

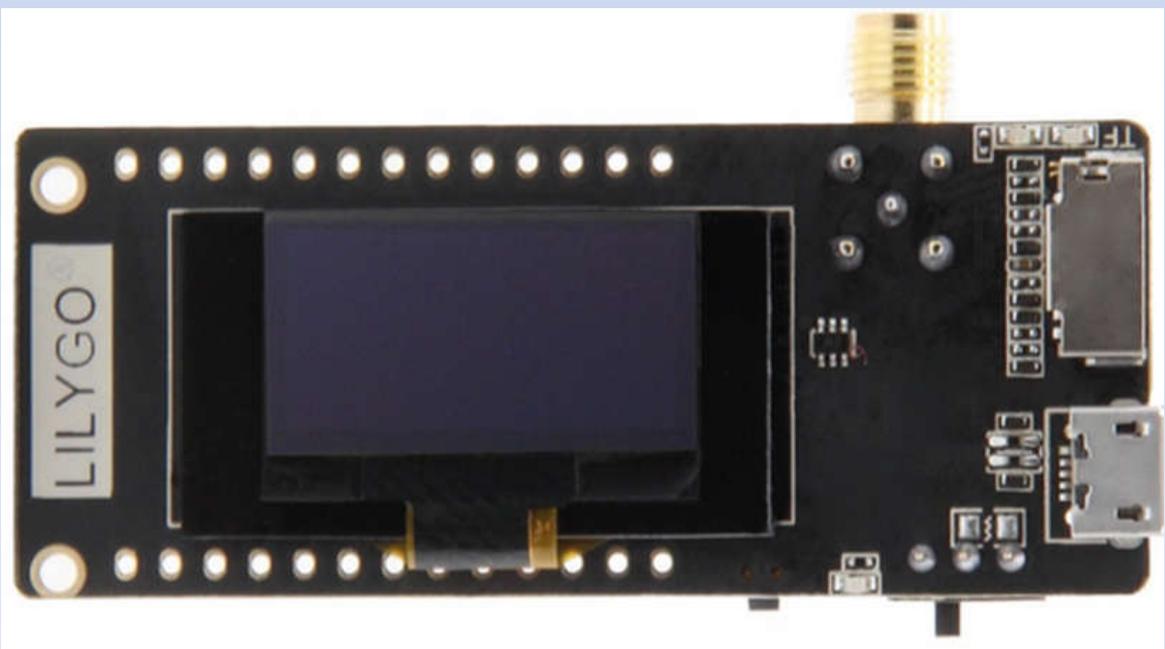
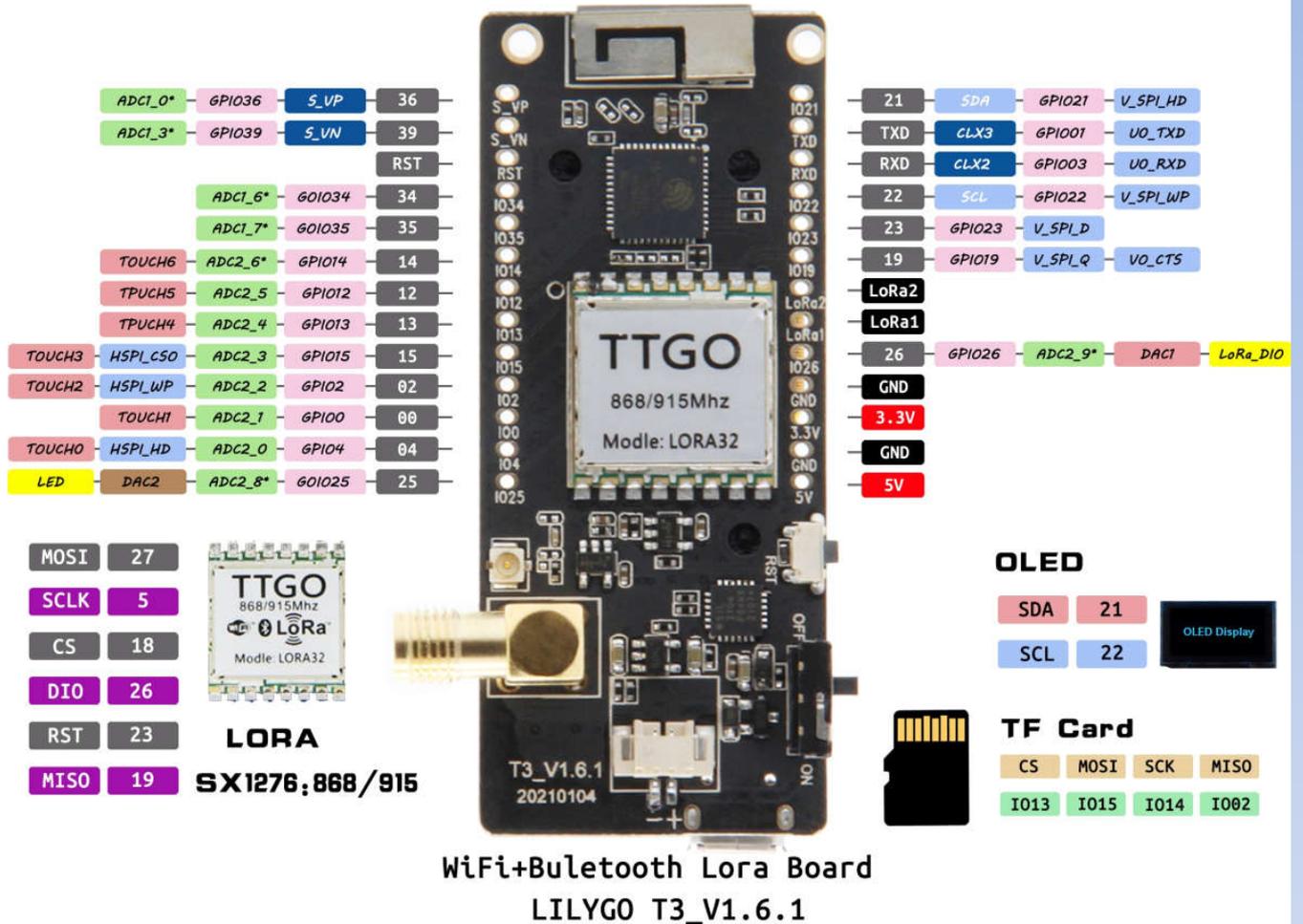
ESP32 (RX)

- TX 34
- RX 12

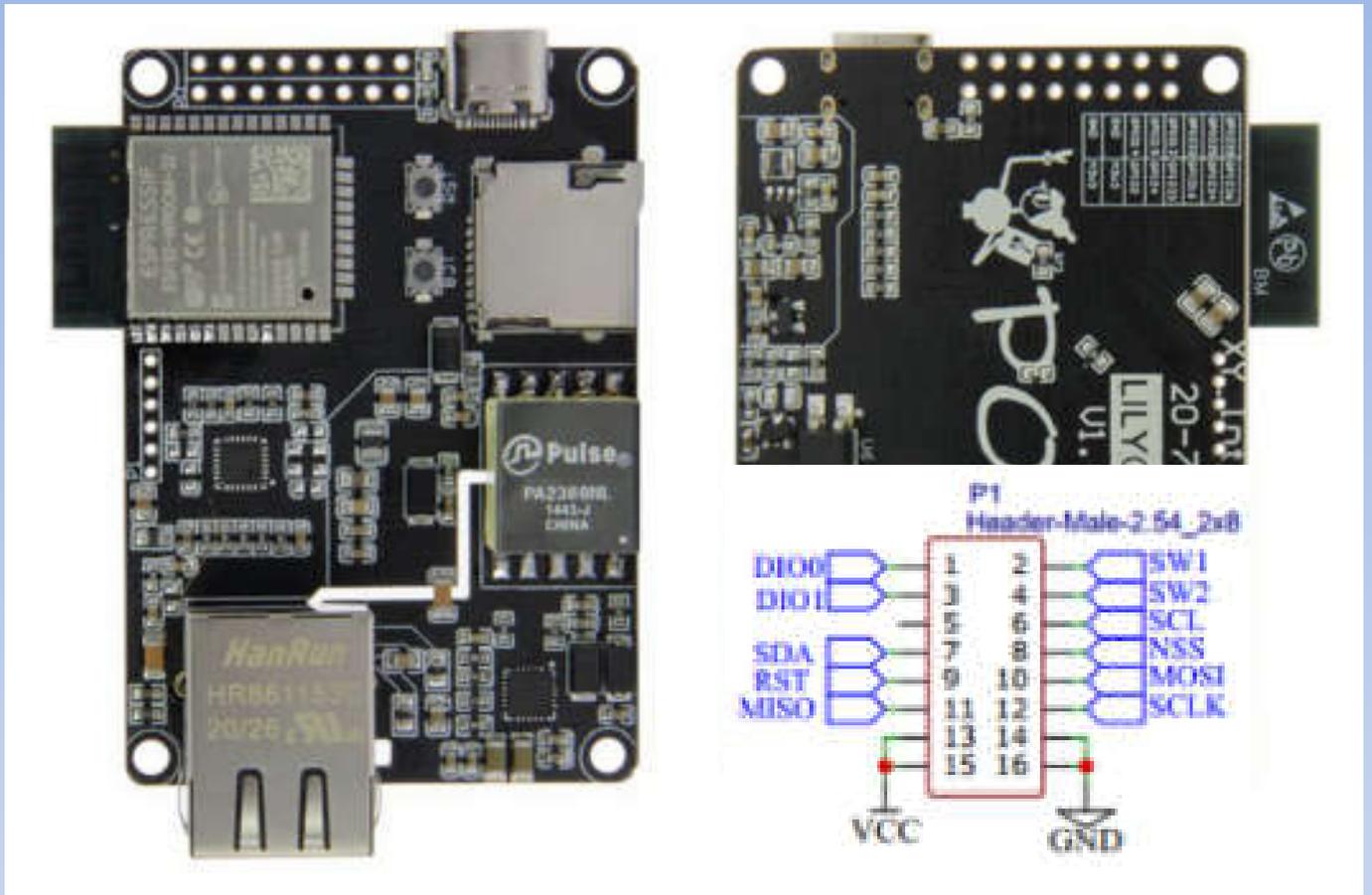
ESP32 (TX)



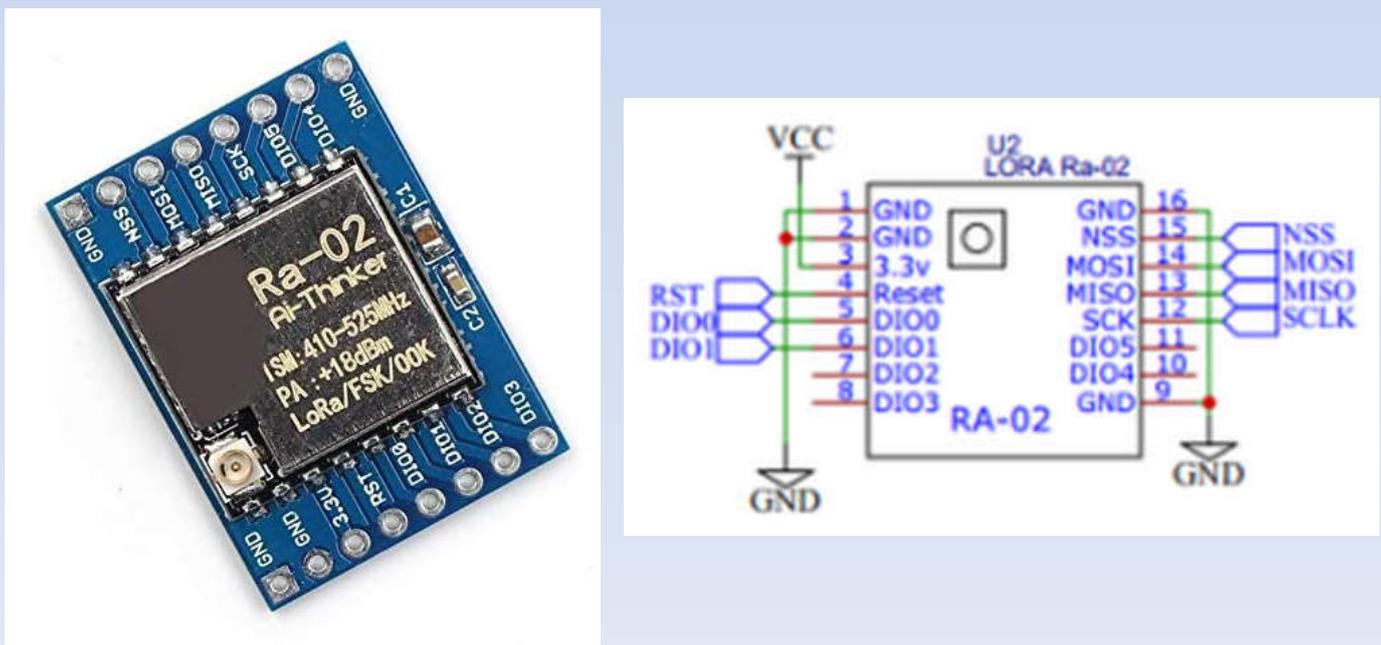
Module « iGate/Digi » TTGO PAXCounter (17€)



Module « Lilygo ESP32 POE » (18€)



Puce LoRa « Ra-02 » (10€)



Tests Terrain

Expériences terrain autour du RCPA

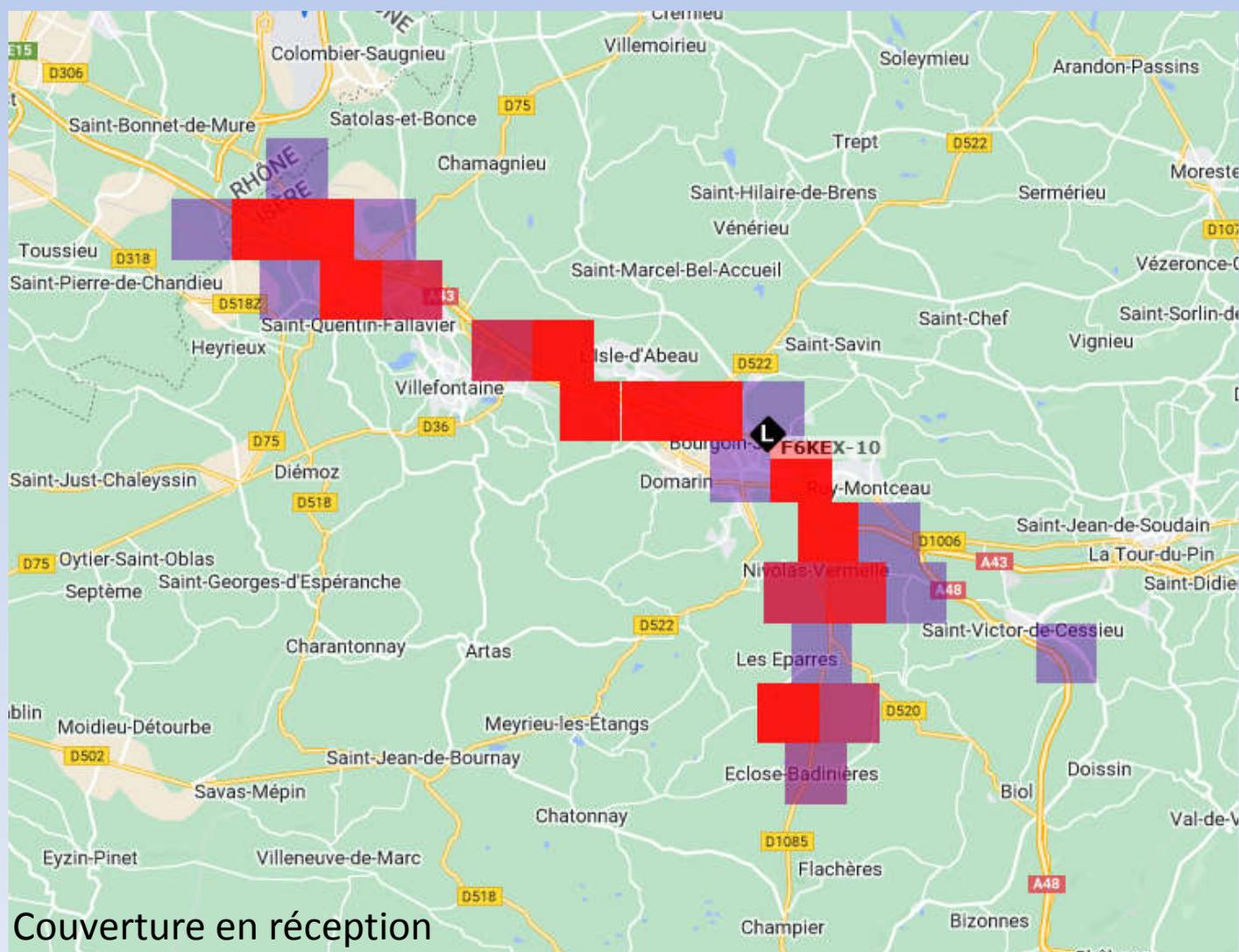
Conditions de test:

F6KEX-10 - IGATE

Matériel: Lilygo **POE** + puce **Ra-02** + pigtail IPEX/N
Antenne: bibande COMET CR8900 (5.5dBi 430MHz)
sur embase magnétique posée sur le shelter du RCPA

F4GFZ/M - TRACKER

Matériel: Lilygo T-Beam 100mW (20dBm)
Antenne: bibande lambda sur embase magnétique au
centre du toit et configuration Smart Beacons



Expériences terrain (2 bonds)

Conditions de test:

F4KIT-4 - DIGIPEATER

Site du Col du Berthiand 01 - pas d'internet sur site

Matériel: Lilygo PAXcounter

Antenne: $\frac{1}{4}$ onde posée sur le shelter du site

F1ZCK-10 - IGATE

Site de Planfoy / Croix de Guizay 42 - lien réseau Hamnet

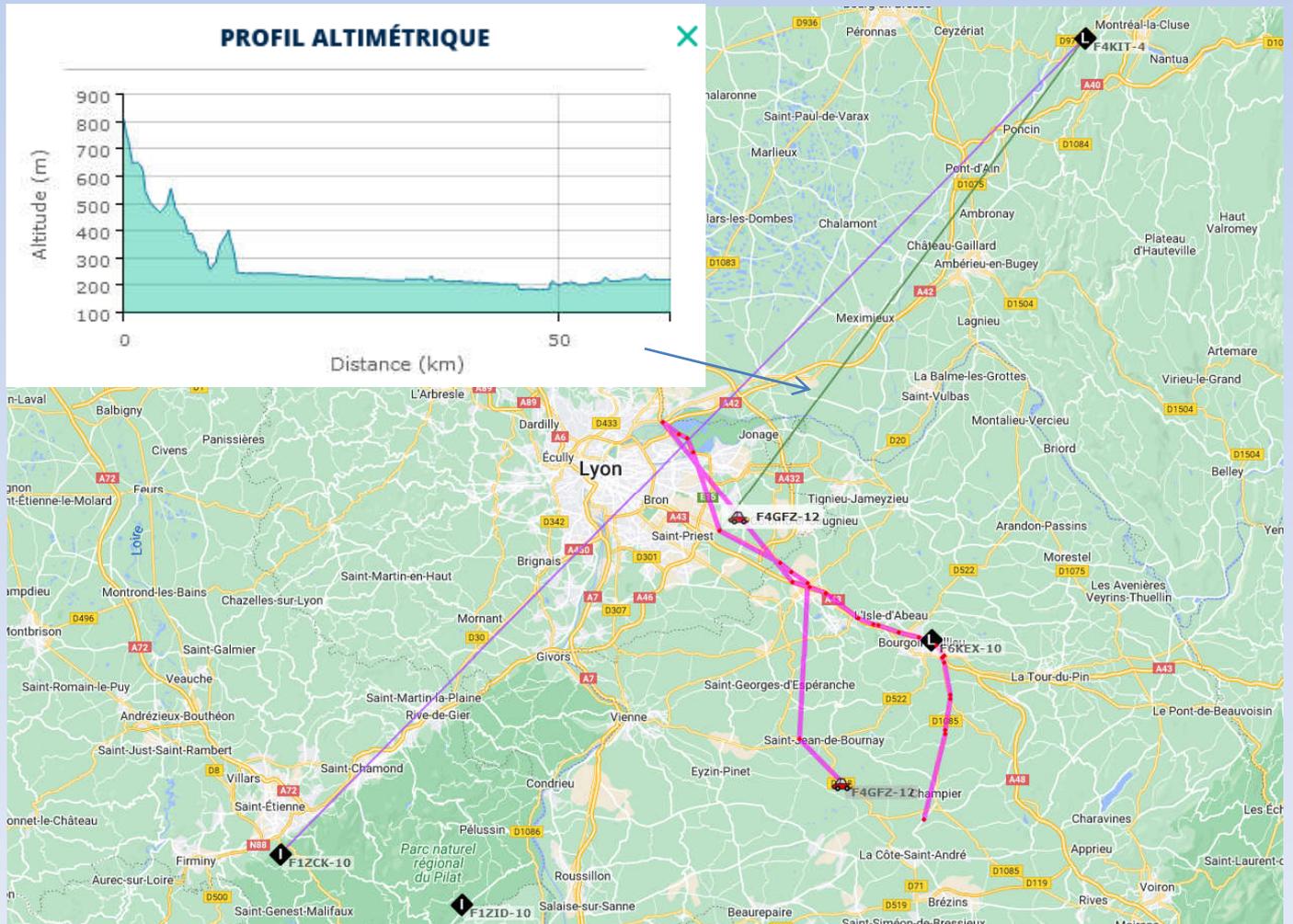
Matériel: Lilygo PAXcounter

Antenne: inconnue

F4GFZ/M - TRACKER

Matériel: Lilygo T-Beam 100mW (20dBm)

Antenne: bibande lambda sur embase magnétique au centre du toit et configuration Smart Beaconsing)



Expériences terrain au sol > **121km DX!** 😊

Conditions de test:

F1ZCK-10 - IGATE

Site de Planfoy / Croix de Guizay 42 - lien réseau Hamnet

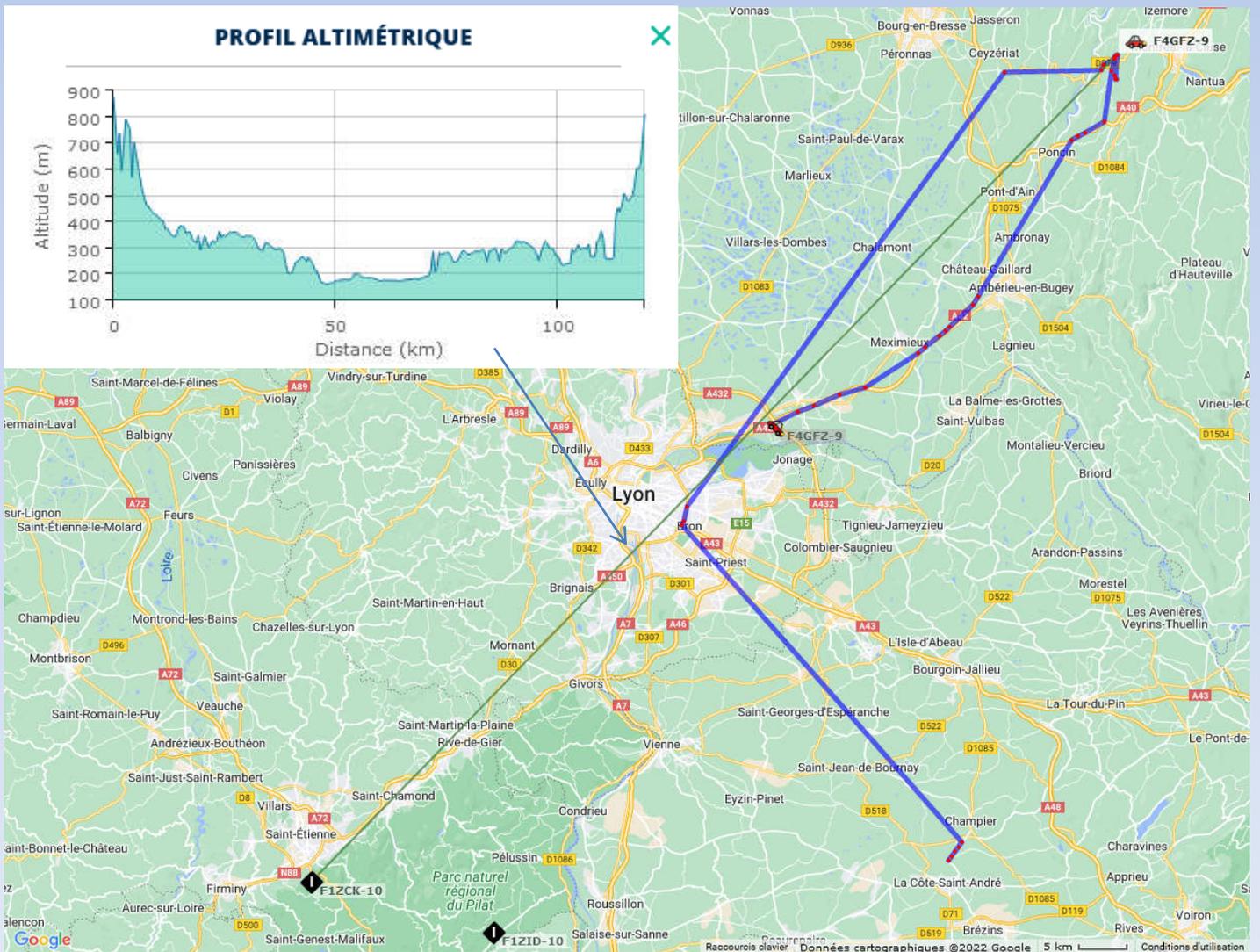
Matériel: Lilygo PAXcounter

Antenne: inconnue

F4GFZ/M - TRACKER

Matériel: Lilygo T-Beam 100mW (20dBm)

Antenne: bibande lambda sur embase magnétique au centre du toit et configuration Smart Beaconsing)



Configuration du tracker F4GFZ-12

Modification aisée de la trame APRS par ajout de variables dans le code source du projet OE5BPA
(informations envoyés toutes les 5 trames APRS dans ma config)

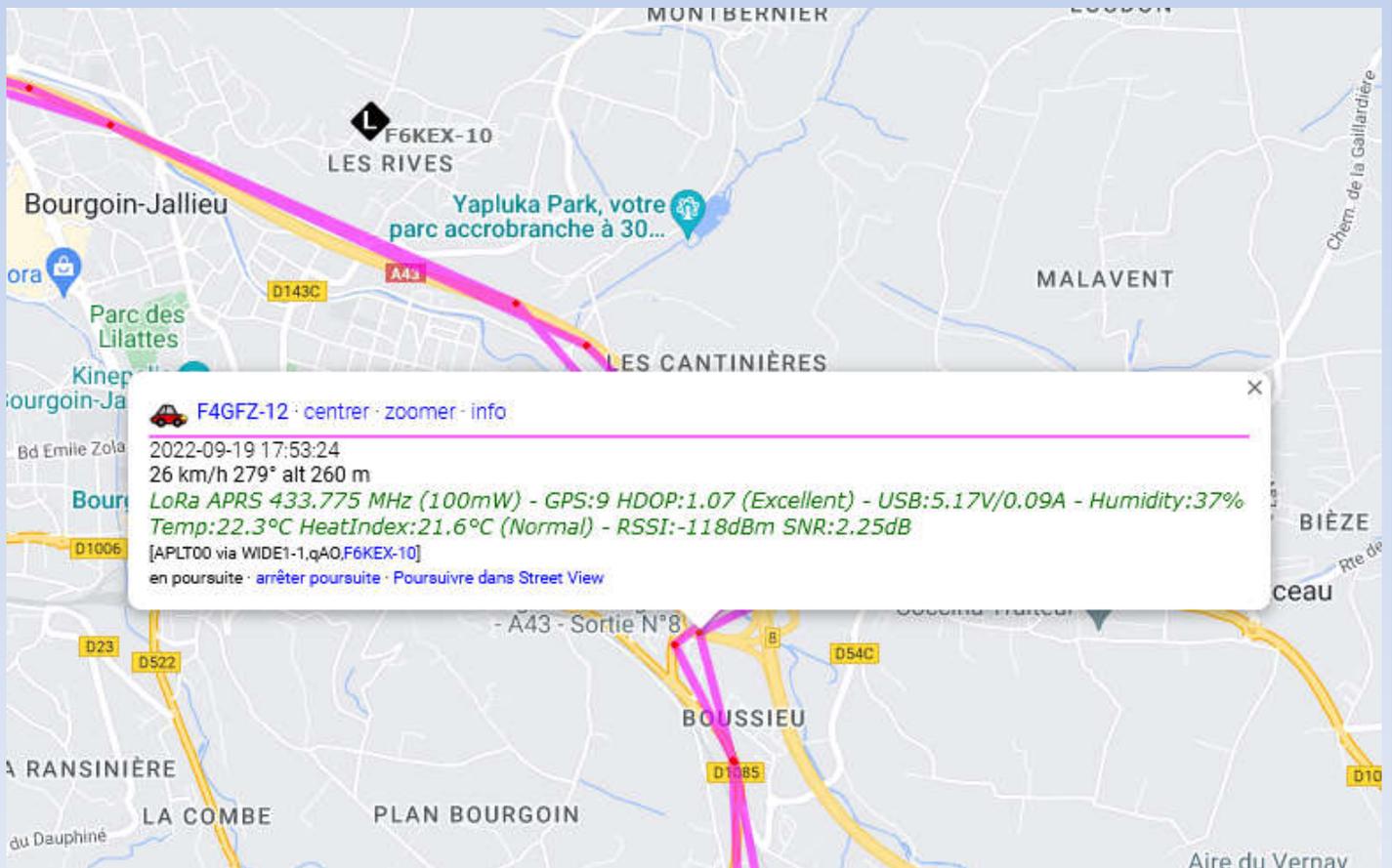
Ajout du nombre de satellites reçus

Ajout du HDOP (précision horizontale GPS)

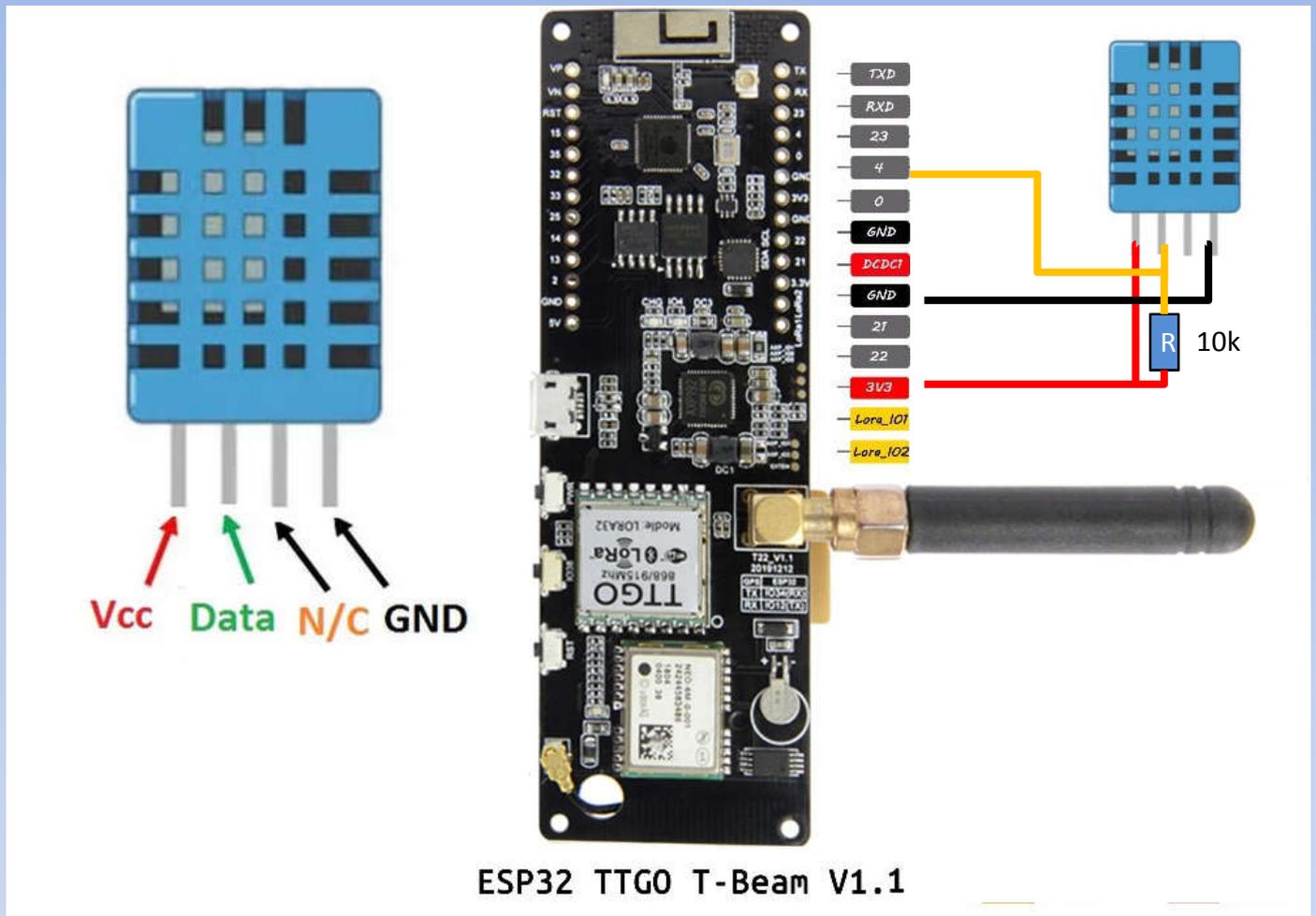
Ajout de la tension et consommation 5V

Ajout de l'Humidité , Température et Indice de Chaleur (sonde DHT11)

En projet, ajout de la Pression Atmosphérique



Câblage de la sonde DHT11



ESP32 TTGO T-Beam V1.1

Déclaration du capteur DHT11

```

15 | #include "DHT.h"
16 |
17 | #define VERSION "22.19.0"
18 | #define DHTPIN 4
19 | #define DHTTYPE DHT11
20 |
21 | DHT dht(DHTPIN, DHTTYPE);
    
```

Initialisation du capteur DHT11

```

67 | void setup() {
68 |     Serial.begin(115200);
69 |     dht.begin();
    
```

Ajout des informations dans la trame APRS

```

aprsmsg += DHTSensorIsConnected ? (String(" - Humidity:") + String(hum,0) + String("% Temp:")
+ String(tem,1) + String("°C HeatIndex:") + String(hic,1) + "°C " + indcha) : "";
    
```

Déclaration des variables de mesures

```

144 | static bool    DHTSensorIsConnected    = true;
145 |
146 | float hum = dht.readHumidity();
147 | float tem = dht.readTemperature();
148 | float hic = dht.computeHeatIndex(tem, hum, false);
149 | if (isnan(hum) || isnan(tem)) {
150 |     DHTSensorIsConnected = false;
151 | } else {
152 |     DHTSensorIsConnected = true;
153 | }
    
```

Correspondance Indices de Chaleur → texte compréhensible

```

294 | // HeatIndex meanings
295 | if (hic < 27) {indcha = "(Normal)"; }
296 | if (hic >= 27 && hic < 32) {indcha = "(Attention)"; }
297 | if (hic >= 32 && hic < 41) {indcha = "(Attention Extreme)"; }
298 | if (hic >= 41 && hic < 54) {indcha = "(Danger)"; }
299 | if (hic >= 54) {indcha = "(Danger Extreme)"; }
    
```