



UNIVERSITAT POLITÈCNICA DE CATALUNYA

Escola Politècnica Superior d'Enginyeria de Manresa

Eines per l'estudi de tecnologies i dispositius LoRa

5 de juliol de 2021

treball de fi de grau que presenta

ENRIC GARCIA DIAZ

en compliment dels requisits per assolir el

GRAU D'ENGINYERIA EN SISTEMES TIC

Direcció:

Jordi Bonet Dalmau

Arnau Arumi Casanovas

Aquesta obra està subjecta a una llicència Attribution-NonCommercial-ShareAlike 3.0 Spain de Creative Commons. Per veure'n una còpia, visiteu <https://creativecommons.org/licenses/by-nc-sa/3.0/es/deed.ca> o envieu una carta a Creative Commons, 171 Second Street, Suite 300, San Francisco, California 94105, USA.

Resum

En aquest treball de final de grau s'estudiaran aspectes sobre les tecnologies i sistemes de comunicacions que utilitzen o incorporen la tecnologia de modulació i comunicació «LoRa» a diferents nivells.

També es treballarà amb solucions per a la interconnexió de dispositius per aplicacions d'Internet de les Coses.

Com a part pràctica, es farà ús de les eines per a la connexió a Internet de dispositius com la tecnologia «LoRaWAN» o la xarxa «The Things Network». D'igual manera, es realitzaran experiments per veure alguns aspectes físics de la modulació i comunicació LoRa.

Aquest treball servirà per a utilitzar la seva informació i parts implementades per aplicacions que puguin utilitzar aquestes tecnologies.

Abstract

This final degree project some aspects of communications technologies and systems that use or incorporate LoRa modulation and communication technology will be studied at different levels.

It will also work with some interconnection for devices and Internet of Things solutions.

As a practical part, It will use some tools for the connection of devices to Internet. Such as LoRaWAN technology or the « The Things Network ». Furthermore, There will be some experiments to study some physical aspects and characteristics of LoRa modulation and communication.

This project will use its information and implemented parts to be used in applications that use these technologies.

Índex

| | |
|--|-----------|
| Resum | i |
| Abstract | i |
| | |
| I. Memòria | 1 |
| 1. Introducció | 3 |
| 1.1. Resum | 3 |
| 1.2. Objectius | 3 |
| 2. Antecedents | 5 |
| 2.1. Exemples | 5 |
| 2.2. Eines i entorns utilitzats | 5 |
| 2.2.1. Llenguatges de programació | 5 |
| 2.2.2. Entorns | 5 |
| 3. Introducció a la modulació LoRa | 7 |
| 3.1. Normativa de bandes de freqüència disponibles | 7 |
| 3.1.1. 433 MHz | 7 |
| 3.1.2. 868 MHz | 7 |
| 3.2. LoRa (Long-Range) modulation | 7 |
| 3.2.1. Spread Spectrum | 7 |
| 3.2.2. LoRa Spread Spectrum | 8 |
| 4. Informació sobre les LPWAN | 11 |
| 4.1. Introducció a les LPWAN | 11 |
| 4.1.1. Sistemes de comunicació Short range | 11 |
| 4.1.2. Sistemes de comunicació Cellular | 11 |
| 4.1.3. Sistemes de comunicació Long range | 11 |
| 4.2. Sigfox | 12 |
| 4.2.1. Aspectes tècnics | 12 |
| 4.2.2. Informació rellevant | 13 |
| 4.3. LoRa | 13 |
| 4.3.1. LoRaWAN | 13 |
| 4.3.2. Introducció | 13 |
| 4.3.3. Classificació dels dispositius | 13 |
| 4.3.4. Certificacions | 14 |
| 4.4. NB-IoT | 15 |
| 4.5. Imatge global | 15 |
| 4.5.1. Les LPWAN | 15 |

| | |
|---|-----------|
| 5. Solucions pels dispositius LoRaWAN | 17 |
| 5.1. Tipus de solucions o productes | 17 |
| 5.1.1. Xip integrat | 17 |
| 5.1.2. Mòdul | 17 |
| 5.1.3. Dispositiu de funció específica | 18 |
| 5.1.4. Placa de desenvolupament | 18 |
| 5.2. SAM R34/R35 | 18 |
| 5.2.1. LoRaWAN Software API | 18 |
| 5.2.2. Mòdul de desenvolupament pel microcontrolador | 19 |
| 5.3. RN2483 Module | 19 |
| 5.4. Informació extra | 19 |
| 6. The Things Network (V3) | 21 |
| 6.1. Arquitectura | 21 |
| 6.1.1. Network Server | 21 |
| 6.1.2. Gateway Server | 22 |
| 6.1.3. Application Server | 22 |
| 6.1.4. Identity Server | 22 |
| 6.1.5. Join Server | 23 |
| 6.2. Registre de Gateways i dispositius | 23 |
| 6.2.1. Registre de dispositius | 23 |
| 6.3. Aplicacions | 24 |
| 6.3.1. Integracions | 24 |
| 7. Llistat de components triats pel projecte | 25 |
| 7.1. Gateway Microtik Lora 868Mhz | 25 |
| 7.2. Antena pel Gateway de Microtik | 25 |
| 7.3. TTN Node Sensor | 25 |
| 7.4. Heltec ESP32 + LORA SX1276 868Mhz | 26 |
| 8. Preparació i configuració dels dispositius | 27 |
| 8.1. The Things Node | 27 |
| 8.1.1. Preparació de l'entorn de software | 27 |
| 8.1.2. Trobar el EUI del dispositiu | 28 |
| 8.2. Heltec ESP32 + LORA SX1276 868Mhz | 28 |
| 8.2.1. Preparació de l'entorn de software d'Arduino IDE | 28 |
| 8.2.2. Càrrega d'un programa d'exemple | 29 |
| 8.2.3. dispositiu LoRaWAN | 29 |
| 8.2.4. Connexió a la xarxa TTN | 30 |
| 8.2.5. Comunicació LoRa punt a punt entre dos dispositius | 30 |
| 8.2.6. Programació en C | 30 |
| 8.2.7. Pujant un programa o projecte al dispositiu amb ESP-IDF (C Programming Language) | 31 |
| 8.2.8. Resum de les implementacions | 32 |
| 8.3. Gateway | 32 |
| 8.3.1. Instal·lació i configuració inicial | 32 |
| 8.3.2. Configuració LoRaWAN del Gateway | 33 |

| | |
|--|-----------|
| 9. Connexió a la xarxa TTN V3 | 35 |
| 9.1. Registre del Gateway a la xarxa | 35 |
| 9.1.1. Registre del dispositiu The Things Node a la xarxa | 35 |
| 9.1.2. Connexió del dispositiu a la xarxa | 35 |
| 9.2. Registre d'un dispositiu Heltec ESP32 LoRa | 36 |
| 9.3. Aplicacions TTN | 38 |
| 9.3.1. Introducció | 38 |
| 9.3.2. Integració MQTT | 38 |
| 10. Proves amb els dispositius | 41 |
| 10.1. Heltec ESP32 + SX1276 LoRa | 41 |
| 10.1.1. Temps de transmissió | 41 |
| 10.1.2. Jugant amb els paràmetres LoRa | 46 |
| 10.2. Desat de dades dels dispositius a una base de dades amb protocol MQTT | 47 |
| 10.2.1. Mòdul python mqtt_processor | 48 |
| 10.2.2. Mòdul python db_utils | 48 |
| 10.2.3. Format del missatge | 49 |
| 10.2.4. Esquema de la base de dades | 50 |
| 10.3. Enviament i visualització de les dades (DDBB influxDB) | 51 |
| 10.3.1. Visualització de les dades enviades | 51 |
| 10.3.2. Servei del sistema operatiu | 51 |
| 10.3.3. Sistema de callbacks | 52 |
| 10.4. Heltec ESP32 + Sensors externs | 54 |
| 10.4.1. Sensor de Temperatura DS18B20 | 54 |
| 10.4.2. Acceleròmetre ADXL345 | 54 |
| 10.4.3. Modificació de la llibreria LoRaWAN | 55 |
| 10.4.4. Dades enviades | 57 |
| 10.5. Proves de consum amb la placa de desenvolupament Heltec ESP32 + LoRa (LoRaWAN) | 59 |
| 10.5.1. Gràfic de consum d'una transmissió amb SF=7 | 59 |
| 10.5.2. Gràfica de consum amb un SF diferent de 7 | 60 |
| 10.6. Instal·lació del Gateway a l'EPSEM | 61 |
| 11. Conclusions | 63 |
| 11.1. Valoració general | 63 |
| 11.2. Valoració personal | 63 |
| Bibliografia | 65 |
| II. Apèndix | 69 |
| A. The Things Network V2 | 71 |
| A.1. The Things Industry | 71 |
| A.2. Arquitectura de la xarxa | 71 |
| A.2.1. Visió Externa | 71 |
| A.2.2. Part interna o Backend | 72 |

| | |
|--|----|
| A.2.3. Funcionament del nucli | 73 |
| A.3. Gateways | 73 |
| A.3.1. Registre d'un Gateway | 73 |
| A.3.2. Exemples de Gateways | 74 |
| A.4. Aplicacions | 76 |
| A.4.1. Registre d'aplicació a la xarxa TTN | 76 |
| A.4.2. Build an Application | 77 |
| A.4.3. APIs | 77 |
| A.4.4. SDK o Llibreries | 77 |
| A.4.5. Integracions | 78 |
| A.5. Dispositius | 78 |
| A.5.1. Exemples de dispositius | 78 |

Part I.

Memòria

1. Introducció

1.1. Resum

Aquest projecte de final de grau tindrà com a objectiu principal estudiar amb profunditat diferents característiques de les tecnologies de comunicació de dispositius de baix consum com sensors o dispositius de L'Internet de les Coses. Més concretament, se centrarà en la síntesi de la informació sobre les tecnologies relacionades amb LoRa i LoRaWAN.

La documentació i síntesi de les tecnologies i aplicacions permetrà en última instància desenvolupar una sèrie d'eines que tindran com a objectiu l'agilització a l'hora de treballar i realitzar estudis en el futur relacionats amb les tecnologies esmentades.

També es procedirà a la realització de l'enviament de dades a Internet dels diferents dispositius que es comuniquen amb LoRa. Es realitzarà una connexió LoRaWAN dels dispositius utilitzant la xarxa «The Things Network». Posteriorment, es veurà la manera d'obtenir les dades dels dispositius a través de les diferents interfícies de la xarxa i quines eines es poden utilitzar.

Altrament, es realitzaran una sèrie d'experiments de comunicació utilitzant la modulació LoRa amb l'objectiu d'avaluar quines són les característiques de la comunicació LoRa.

1.2. Objectius

De manera resumida, els objectius d'aquest projecte seran els següents:

- Recerca relacionada amb dispositius, xarxes i comunicacions amb modulació LoRa.
- Recerca sobre la tecnologia i arquitectura LoRaWAN i els seus tipus de dispositius.
- Recerca sobre la xarxa «The Things Network».
- Connexió de dispositius i Gateway LoRaWAN a la xarxa TTN.
- Connexió d'aplicacions externes a la xarxa mitjançant les seves interfícies per a recollir les dades dels dispositius.
- Visualització de les dades rebudes dels dispositius des de la DDBB.
- Realització d'experiments per veure aspectes de la comunicació entre dos dispositius amb modulació LoRa.
- Instal·lació d'un Gateway a l'EPSEM.

2. Antecedents

En aquest capítol, es nomenaran algun dels treballs previs que s'han realitzat relacionats amb el tema d'aquest projecte de final de grau. També, es mostraran les eines, llenguatges i entorns utilitzats per al desenvolupament del projecte.

2.1. Exemples

Entre els diferents treballs i projectes elaborats, s'han trobat exemples d'aplicacions per casos o estudis pràctics en un àmbit més específic. Un exemple seria [57], on es parla de l'ús de tecnologies com LoRa o LoRaWAN per a la gestió de l'aigua en els entorns rurals.

Un altre exemple, és el de l'article [58] on parla de la implementació d'un sistema de monitoratge de la qualitat de l'aire utilitzant LoRa i LoRaWAN.

2.2. Eines i entorns utilitzats

2.2.1. Llenguatges de programació

Els llenguatges de programació utilitzats per aquest projecte han sigut principalment python3 i C.

- python: S'ha utilitzat python per al programa que s'encarrega de recollir les dades dels dispositius de la xarxa «The Things Network».
- C: s'ha utilitzat per a programar i modificar les llibreries que es carregaran al dispositiu. També ha sigut necessari per a l'estudi i lectura del codi de les llibreries LoRaWAN dels dispositius.

2.2.2. Entorns

Els entorns que s'han utilitzat, han sigut els següents:

- Sistema Operatiu - GNU/Linux: El sistema operatiu on s'ha desenvolupat les parts pràctiques del projecte ha sigut del tipus GNU/Linux.
- emacs: com a entorn de programació principal del projecte.
- Arduino IDE: entorn per a programar els dispositius LoRa.
- ESP-IDF: entorn per a programar els dispositius amb microcontroladors ESP32 i compilar programes escrits en C.
- Bases de dades: s'han utilitzat bases de dades del tipus MySQL i influxDB.

- Grafana: plataforma de visualització gràfica de les dades de dispositius o sensors.
- The Things Network: S'ha utilitzat com a servei principal LoRaWAN pels dispositius.

3. Introducció a la modulació LoRa

3.1. Normativa de bandes de freqüència disponibles

3.1.1. 433 MHz

Les bandes de freqüència disponibles i classificades com a bandes d'ús comú i d'ús especial són les que van de 433.05 - 434,79 MHz. Aquestes són bandes de freqüència designades i disponibles per aplicacions industrials, científiques o mèdiques. [1] pàg. 242.

Aquestes bandes estan descrites com a bandes per aplicacions ICM. Són bandes que estan ocupades per aquestes aplicacions i qualsevol servei de comunicacions de radiofreqüències que comparteixi aquestes freqüències, haurà de tenir-ho en compte.

3.1.2. 868 MHz

També hi ha les bandes 868 - 870 MHz. Estan destinades per aplicacions de baixa potència. Aquestes dues bandes de freqüència són les que se solen utilitzar per dispositius que utilitzen LoRa. [1] pàg. 244.

Per a la utilització d'aquestes bandes de freqüències s'han de complir una normativa de potència de senyal radiada i màxim cicle de treball.

3.2. LoRa (Long-Range) modulation

És una tècnica de modulació que es pot utilitzar en les xarxes low-power wide-area (LPWAN). És a dir, és una tècnica de modulació que permet el desenvolupament de xarxes que poden englobar una àrea considerablement àmplia (de l'ordre de kilòmetres). També es considera com a low-power pel poc consum dels dispositius que formen part de la xarxa. LoRa és una tècnica de modulació de radiofreqüència del tipus chirp spread spectrum Propietat de Semtech [2].

Tots aquests aspectes característics són deguts al baix bit rate amb el que aquests tipus de xarxes treballen.

3.2.1. Spread Spectrum

LoRa és una tècnica de modulació que entra dins del conjunt que realitzen Spread Spectrum. Aquest, té com a objectiu principal repartir l'espectre del senyal de dades en el domini freqüencial repartint-lo en una amplada de banda major.

Amb això s'aconsegueix que un cop modulada la senyal i transmesa, sigui més resistent a les interferències o al soroll.

Dos exemples de derivats del concepte de Spread Spectrum:

- Direct Sequence Spread Spectrum (DSSS).
- Chirp Spread Spectrum (CSS).

Direct Sequence Spread Spectrum

Consisteix a aplicar una seqüència digital multiplicant-la als bits de dades de manera que el senyal queda repartit en una amplada de banda més repartit al llarg de l'espectre.

El receptor, després de desmodular, haurà d'aplicar la mateixa seqüència digital al senyal rebut per a recuperar les dades. Un dels inconvenients principals d'aquests sistemes és la complexitat de sincronització respecte el senyal amb la seqüència que s'ha d'aplicar per a descodificar la informació.

Això fa que els receptors tinguin uns costos econòmics i consums més elevats.

Chirp Spread Spectrum

Consisteix en aplicar «chirp pulses» al senyal de dades per codificar la informació. Aquests «chirp pulses» són senyals que incrementen o decreixen la seva freqüència respecte el temps. LoRa, està basat en aquest mètode.

És un esquema que redueix els costos i consums dels receptors respecte al DSSS.

3.2.2. LoRa Spread Spectrum

És un esquema de modulació propietat de Semtech que posa com a focus principal, millorar aspectes de consum i cost general dels dispositius. S'aconsegueix mitjançant l'aplicació d'un «chirp signal» que varia la freqüència de manera contínua. Amb això s'aconsegueix reduir la complexitat dels receptors.

Paràmetres de modulació

Existeixen certs paràmetres que es poden aplicar a la configuració d'una comunicació entre dos dispositius LoRa.

- Spreading factor (SF): 7..12
- Code rate (CR): 1..4
- Modulation bandwidth (BW)

El Spreading factor permet regular l'extensió en freqüència del senyal (spread spectrum). De manera que com més alt, més es redueix els bit-rates però més millora la sensibilitat del receptor (es veu menys afectat per l'atenuació o interferències).

Pel que fa al Code-rate, és un valor que permet modificar la configuració de la codificació per a la gestió dels errors.

Per últim, el BW, fa referència a l'amplada de banda del senyal sense aplicar l'extensió del seu espectre. Com més gran sigui majors bit-rates es poden aconseguir.

4. Informació sobre les LPWAN

En aquest apartat es determinaran aspectes i característiques sobre els diferents tipus de comunicacions denominades com Low-Power Wide-Area Network [5]. Aquests tipus de xarxes poden ser útils per als conjunts de sensors i dispositius que tenen com a característiques o objectius principals el seu baix consum, una comunicació sense fils de gran abast i baix cost econòmic.

4.1. Introducció a les LPWAN

Com ja s'ha dit, les principals característiques d'aquest tipus de comunicació és afavorir els consums dels seus dispositius o nodes a l'hora que s'augmenta la distància màxima de cobertura entre dispositius que s'estan comunicant.

Per a aconseguir aquests objectius pel sistema de comunicació, s'ha de sacrificar els «data rates». Aquests s'hauran de reduir, que vol dir que els dispositius s'han d'ajustar a aquestes característiques.

4.1.1. Sistemes de comunicació Short range

Aquests són sistemes de comunicació que es caracteritzen principalment per tenir una àrea de cobertura reduïda. Això fa que els dispositius que es comuniquin hagin d'estar dissenyats o pensats perquè estiguin propers durant la comunicació.

Alguns exemples serien el Bluetooth, Zigbee o NFC

4.1.2. Sistemes de comunicació Cellular

Aquests sistemes permeten tenir un «data rate» elevat perquè estan pensats per a l'oferiment de serveis de connectivitat a Internet pels usuaris. A més a més, solen estar pensats per a poder oferir aquests serveis amb una àrea de cobertura considerable.

El principal problema d'aquests tipus de sistemes de comunicació és la necessitat d'una infraestructura desenvolupada i integradora de diferents tipus de tecnologies. És per això que sol ser considerablement costós el desenvolupament d'aquests sistemes.

4.1.3. Sistemes de comunicació Long range

Aquest és el conjunt de sistemes de comunicació que conté les LPWAN. Es caracteritzen per tenir un rang d'abast superior a les anteriors mencionades. Algun exemple d'aquest tipus de sistema és el Sigfox, LoRa o NB-IoT.

4.2. Sigfox

És un operador de xarxa tipus LPWAN que ofereix connectivitat per aplicacions en IoT. Utilitza les seves pròpies tecnologies per a poder oferir els seus serveis. Es basa en una infraestructura de «Base Station» pròpies que permeten als dispositius de la xarxa connectar-se a servidors via Protocol d'Internet (IP) [6].

4.2.1. Aspectes tècnics

En aquest apartat es descriuran els aspectes tècnics de la comunicació entre els dispositius i les «Base Station» (BS).

Tècnica de modulació:

La tècnica de modulació que s'utilitza per a les comunicacions entre dispositius en una xarxa Sigfox pel «Up-Link» (UL) és el Differential Binary Phase-Shift keying (DBPSK) que és una variant de BPSK. En el cas del «Down-Link» (DL) s'utilitza (GFSK) [8].

Binary Phase-Shift Keying (BPSK): És una variant de BPSK que és una tècnica de modulació que assigna l'estat d'un sol bit d'informació per una fase. És a dir, s'utilitzen dues fases per a modular els bits d'informació, per tant, únicament es pot enviar un bit per símbol. Els consums són baixos gràcies al treball amb bit rates baixos.

L'atribut Differential, indica que en comptes d'assignar una fase concreta a cada estat de bit ('0' o '1'), s'assigna una suma de fase respecte a l'anterior. És a dir, per a modular se li suma l'angle que té assignat cada estat a la fase del símbol anterior [7].

Bandes de freqüència i aspectes de la comunicació

En aquest apartat es descriuran les especificacions de radiofreqüència per a la comunicació entre dispositiu i BS en una xarxa Sigfox [8].

- Es defineixen dos macro-channels un pel «Down-Link» (DL) i l'altre pel «Up-Link» (UL).
- S'utilitzen les unlicensed ISM bands de cada regió.
- L'amplada de banda de cada macro-channel és de 192kHz.
- Les velocitats de transmissió de la comunicació poden ser de 100bps o 600bps en Europa (RC1).
- La sensibilitat del receptor és de -126dBm.
- S'ha de complir amb les regulacions «ETSI 300 220» [9] «(Annex B)» per a poder utilitzar la banda de freqüència sense llicència a Europa. 25 mW e.r.p i ocupació de canal menor al 0.1% o 1% del cicle de treball.

| Freqüència (MHz) | Europa | RC2 (EUA and Mèxic) | RC4 (Amèrica Llatina) |
|--------------------|---------|---------------------|-----------------------|
| UL central | 868.130 | 902.200 | 920.800 |
| DL central | 869.525 | 905.200 | 922.300 |
| Separació de canal | 1.395 | 3.000 | 1.500 |

4.2.2. Informació rellevant

La mateixa xarxa et limita el nombre de missatges que pots transmetre al dia. Has de pagar pel servei i alguns aspectes de millora del servei es veurà reflectit en el cost d'aquest.

4.3. LoRa

És un sistema de comunicació que es defineix fins a la capa física on s'integra amb una tècnica de modulació patentada. Utilitza les bandes ISM sense llicència: 868 MHz en Europa. Aquesta forma de modulació es basa en l'extensió de l'espectre per a augmentar la robustesa i el SNIR del senyal.

Una xarxa de dispositius que utilitzen la capa física LoRa per a desenvolupar una xarxa LPWAN rep el nom de LoRaWAN. Sol ser una bona opció gràcies al baix cost dels transceptors LoRa que comporta una facilitat econòmica de desenvolupament de la xarxa.

4.3.1. LoRaWAN

LoRaWAN és un estàndard o especificació definit per la LoRa-Alliance que defineix un protocol o arquitectura de comunicació que utilitza i es basa en LoRa. [11]

LoRaWAN entra dins del grup de xarxes o protocols Low Power wide Area Network. On garanteix la connectivitat i la seguretat dels dispositius que formen part de la xarxa. Sobretot se centra a proporcionar solucions per a aplicacions de l'internet de les coses (IoT).

4.3.2. Introducció

L'arquitectura de les LoRaWAN té com a objectiu facilitar o aconseguir la comunicació entre els dispositius i un servidor central a través de IP. Principalment, els dispositius es poden comunicar amb el servidor a través d'un Gateway.

El Gateway és el pont entre els dispositius i el servidor. Aquest, té la capacitat d'establir comunicacions amb els dispositius utilitzant la capa Física LoRa. A més, també pot establir connexions amb servidors a través del Protocol d'Internet.

4.3.3. Classificació dels dispositius

L'especificació de per a dispositius d'una xarxa LoRaWAN defineix tres classes de dispositius que pot tenir o suportar el tipus de xarxa.

Class A: Lowers power, bi-directional end-devices

Aquests tipus de dispositius són els que tenen menor consum i els que tenen un major suport en les xarxes LoRaWAN. Principalment, aquests dispositius es caracteritzen per tenir sempre la iniciativa en la comunicació. Posteriorment, poden rebre dues finestres curtes de descàrrega just després del seu missatge previ.

Aquesta iniciativa per part del dispositiu pot ser presa en qualsevol moment de manera asíncrona i en el moment i període que desitgi. Tot això té com a conseqüència que els dispositius d'aquest tipus no necessiten estar pendents de missatges externs i permet obtenir millors consums dels dispositius. Pot estar en mode «sleep» mentre no hagi de comunicar-se per a transmetre la informació.

Com que cada missatge de descàrrega (downlink) ha de ser precedit per un de càrrega (uplink) si el servidor ha de transmetre molts missatges, aquests s'hi hauran d'encuar en un «buffer» de transmissió conforme vagin arribant missatges del dispositiu.

Class B: Bi-directional end-devices with deterministic downlink latency

En aquests tipus de dispositius s'estableix un període de sincronització de la xarxa perquè el servidor pugui enviar dades als dispositius cada cert temps. Això implica que ha d'existir una sincronització de la xarxa sencera (servidor i dispositiu). De manera que s'ha d'assegurar que el sensor estigui escoltant alhora que s'està rebent un missatge de «downlink».

Aquest temps pot anar fins als 128 segons. Com és lògic, aquesta classe de dispositius tenen un consum superior als de classe A. Malgrat això, es poden aconseguir consums molt acceptables.

Aquests tipus de dispositius poden ser útils per si es necessita una comunicació més freqüent amb el servidor que en els de la classe A.

Class C: lowest latency, bi-directional end-devices

Amb aquesta classe de dispositius l'objectiu és el de reduir al màxim la latència en la comunicació. Això s'aconsegueix gràcies al fet que el dispositiu està escoltant mentre no està transmetent. D'aquesta manera es pot assumir que el missatge enviat en qualsevol moment, pel servidor, serà escoltat si el dispositiu està operatiu.

Aquests dispositius estan pensats perquè estiguin alimentats de manera constant de manera externa, sense bateria. Per aquest motiu, aquests tipus de dispositius no estan pensats per a obtenir consums baixos.

Nota: Els tipus de dispositius B i C també utilitzen el protocol de comunicació dels de classe A. D'aquesta manera, es pot implementar els dos comportaments en un dispositiu perquè actui com un de classe A o C en el moment que es desitgi. És a dir, aquest mode pot tenir aplicacions temporals i més específiques com per exemple l'actualització de firmware o configuració remota.

4.3.4. Certificacions

Els dispositius que tenen implementat aquest protocol LoRaWAN han de passar una sèrie de proves de certificació per a poder assegurar que compleixen les especificacions de protocol i normativa. Per descomptat és imprescindible perquè un producte pugui portar l'etiqueta de LoRaWAN. [10]

El programa de certificació està disponible per als dispositius de classe A. Pels gateways encara no n'hi ha.

També hi ha disponible una sèrie de productes que compleixen amb la certificació que compleixen les especificacions de les xarxes de tipus LoRaWAN.

End-device Certification

La certificació té l'àmbit en la capa d'enllaç per a una xarxa d'especificació LoRaWAN [10] i pels paràmetres regionals per les especificacions LoRAWAN [12]. Tot això per als dispositius de classe A.

S'ha de desenvolupar una capa d'aplicació per a poder realitzar els tests de certificació. Aquesta capa treballarà amb el FPort 224 que s'haurà de deshabilitar o eliminar, en cas que el producte no integri capes superiors, per a les aplicacions de producció (és únicament de test).

4.4. NB-IoT

NB-IoT (Narrow Band Internet of Things), Forma part de les xarxes d'abast ampli i de poc consum. L'estàndard va ser desenvolupat pel grup de treball 3GPP com a un servei extra per les xarxes mòbil (Cellular Networks).

La primera versió que es va incorporar al LTE standard (Release 13) va ser el 2016. Utilitza la modulació OFDM pel «downlink» i la SC-FDMA pel «uplink». L'amplada de banda és de 200 kHz.

Al ser un servei que s'ha incorporat durant el desenvolupament de la xarxa mòbil de quarta generació, molts operadors han hagut d'agregar-lo per a oferir el servei. En la següent generació de xarxes mòbil ja estaran estandarditzades les bandes per a oferir serveis d'IoT des del principi, un exemple seria com el rang de freqüències 1 de l'estàndard 5G-NR són les mateixes que en LTE.

4.5. Imatge global

4.5.1. Les LPWAN

Com ja s'ha dit, principalment, les LPWAN són un tipus de xarxes que es caracteritzen pel seu ampli abast del senyal i els seus datarates baixos. Tot Això genera un baix consum en els dispositius que forment part d'aquest tipus de xarxes.

Un gràfic extret d'una de les fonts bibliogràfiques [5] que representa la situació de les LPWAN respecte a altres sistemes de comunicació és el següent:

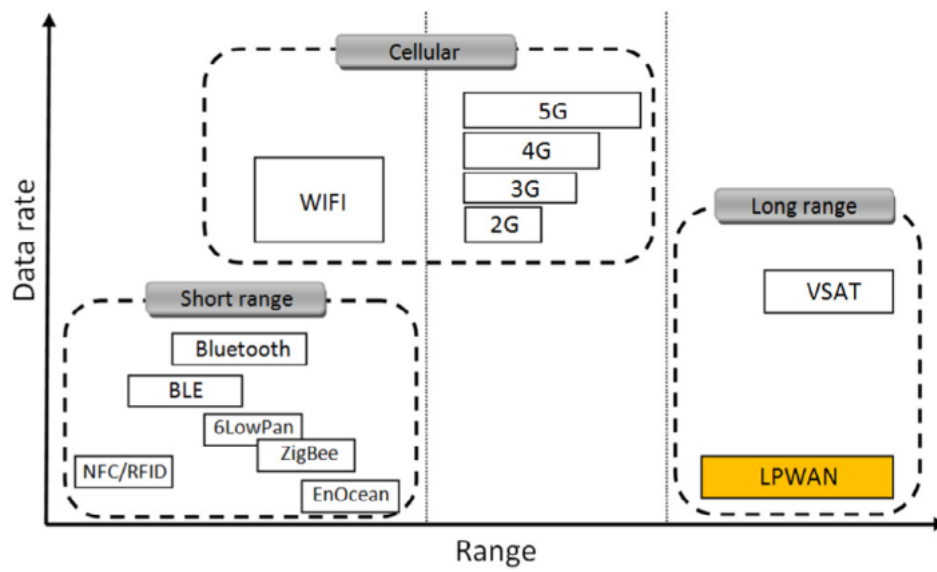


Figura 4.1.: Required data rate vs range capacity

5. Solucions pels dispositius LoRaWAN

Hi ha molts exemples de dispositius que implementen el protocol LoRaWAN. En aquest capítol es mostraran alguns exemples de sistemes de diferents tipus i de diferents solucions per a oferir un producte.

Aquestes solucions no són les que s'han utilitzat per a la part pràctica del projecte. Són les solucions que tenen més elasticitat a l'hora de realitzar un disseny de dispositiu però que augmenten la seva complexitat i demanda de recursos a l'hora de confeccionar-les.

5.1. Tipus de solucions o productes

En aquesta secció es veuran els principals tipus de solucions que es poden trobar per a desenvolupar o obtenir un dispositiu que pugui formar part d'una xarxa LoRaWAN.

5.1.1. Xip integrat

Aquesta solució té com a component únic un xip integrat que realitza les funcions de modulació compatibles amb la modulació LoRa. Es pot optar per un component que únicament actua de transceptor, controlat per un microcontrolador extern, o que formi part d'un sistema integrat en un sol xip juntament amb una unitat de processat o microcontrolador (perifèric).

Com que aquesta solució únicament s'implementa la capa física, les altres capes del protocol LoRaWAN queden per implementar en aquesta opció.

És una opció que pot ser interessant també, en els casos que estan ben documentats i es poden realitzar implementacions pròpies per a conèixer la tecnologia.

També pot ser una opció si es vol realitzar un desenvolupament més compacte per una aplicació més específica. És l'opció més flexible, però la que té més càrrega d'investigació i desenvolupament.

5.1.2. Mòdul

En aquest cas, es dóna un mòdul aïllat del microcontrolador principal i es comunica amb aquest amb una interfície o protocol de comunicació com pot ser l'UART, i2c, SPI...

És una solució més còmoda que en el cas anterior, perquè ve amb una API que proporciona el fabricant per a poder utilitzar-lo de manera externa.

La principal diferència entre aquesta solució i l'anterior és la integració de la interfície en el mateix encapsulat que el microcontrolador. També cal destacar que en el cas anterior s'han d'afegir circuits externs a l'encapsulat.

5.1.3. Dispositiu de funció específica

Aquest és el cas d'un dispositiu que no s'ha de realitzar un desenvolupament intern. Únicament s'ha de realitzar una configuració per afegir-lo al sistema. Un exemple seria un sensor que ja ve amb el sistema de comunicació LoRaWAN integrat.

És la solució més còmoda per afegir-la a un sistema però és la menys flexible.

5.1.4. Placa de desenvolupament

És un producte que incorpora un sistema amb microcontrolador amb un chip de modulació LoRa amb altres aspectes i interfícies.

Pot ser útil per al procés de desenvolupament o disseny d'un sistema que incorpori característiques similars. Se sol caracteritzar pel seu suport quant a llibreries i elasticitat d'eines a l'hora de treballar juntament amb una bona documentació.

5.2. SAM R34/R35

És una família de microcontroladors de molt baix consum. Té una interfície de comunicacions per a un transceptor de UHF. Aquesta interfície té suport per la modulació LoRa i FSK.

Alguns aspectes de les característiques operacionals del chip, més a [13]:

- Arquitectura: ARM Cortex M0+ fins a 48 MHz clk.
- Memòria Flash de (256, 128 o 64) KBytes.
- Memòria SRAM de (32, 16 o 8) KBytes.

5.2.1. LoRaWAN Software API

El fabricant proporciona una llibreria que implementa el protocol LoRaWAN pel microcontrolador [13]. També proporciona una API per a utilitzar-la [14].

A més de la llibreria que implementa la capa LoRaWAN, també es proporciona una sèrie de mòduls per les diferents funcionalitats del microcontrolador. També algunes implementacions de firmware més específiques per al control del sistema per l'UART [13].

Cal destacar que aquest software que implementa les capes pel protocol LoRaWAN no se'n proporciona el codi font. Únicament, està disponible l'API per a utilitzar la llibreria.

També cal destacar que la interfície de modulació del microcontrolador no està documentada. [13] (Datasheet punt 10).

La interfície del microcontrolador està internament composta per un «master» SPI es comunica implementat amb la família de microcontroladors SAM L21 [17] amb un «slave» que gestiona el transceptor. No s'ha trobat més informació del control d'aquest perifèric del microcontrolador, S'ha d'utilitzar les llibreries proporcionades pel fabricant [13].

5.2.2. Mòdul de desenvolupament pel microcontrolador

També existeix una placa que està pensada per aspectes de desenvolupament de software per sistemes amb el microcontrolador [15].

5.3. RN2483 Module

Aquest producte desenvolupat i distribuït per l'empresa Microchip, consta d'un mòdul que implementa les capes pel protocol LoRaWAN. En el cas d'aquest mòdul, la interfície que utilitza per a la comunicació és l'UART.

Aquest mòdul ve programat amb un firmware que implementa una API per a poder enviar comandes al mòdul i disposar de les seves funcionalitats [16].

5.4. Informació extra

Generalment, en la majoria de solucions, el fabricant posa a disposició una llibreria o software que implementa la capa LoRaWAN o la capa física LoRa. Depenent de la companyia que ven el producte dóna una documentació més detallada amb un codi obert o opten per una opció més privativa on donen una llibreria amb una API o una API única per accedir a les funcionalitats de forma externa.

6. The Things Network (V3)

Actualment, s'està posant en marxa la versió 3 del servidor (The Things Stack V3). Això implicarà que la versió anterior serà substituïda finalment per aquesta més nova.

Aquesta versió té com a objectiu augmentar l'experiència dels usuaris de la xarxa, dur a terme més funcionalitats en el futur i augmentar la seguretat. Com ja es comenta en la documentació [34], aquesta nova versió és una reformulació i implementació d'una nova arquitectura interna de la xarxa TTN.

6.1. Arquitectura

L'arquitectura interna del servidor és mostra a la següent imatge [35]:

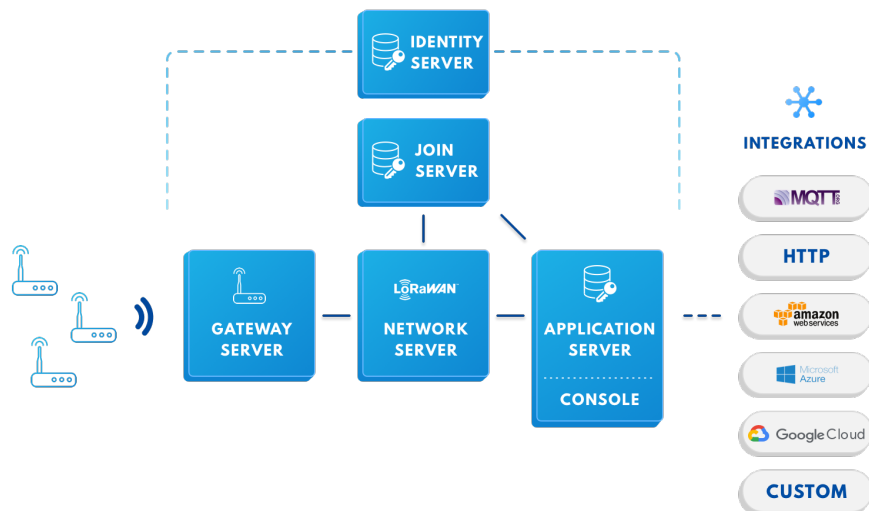


Figura 6.1.: Backend architecture The Things Network V3

6.1.1. Network Server

Aquesta part de la xarxa és l'encarregada de controlar la capa de xarxa del protocol LoRaWAN. Controla aspectes com les comandes MAC, els paràmetres regionals (freqüències i posicions) o els data rates adaptatius (ADR). Ha de tenir registrats els diferents dispositius que formen part de la xarxa.

Es comunica amb el «Application Server» per a controlar i distribuir els missatges entre aplicacions i dispositius. En la documentació hi ha explicats aspectes més específics com el control dels missatges de downlink i uplink cap al «Gateway Server».

6.1.2. Gateway Server

És el servidor a què es poden connectar els gateways que formen part de la xarxa. Es poden connectar utilitzant diferents protocols i ports com MQTT, Semtech Packet Forwarder, HTTP, gRPC o Web Sockets.

El protocol utilitzat pels Gateways per a la transmissió de dades és el MQTT.

La funció principal del «Gateway Server» és la de mantenir la connexió entre les Gateways i el «Network Server».

6.1.3. Application Server

És la part de la xarxa que s'encarrega de gestionar i controlar l'accés als missatges enviats pels dispositius. També, permet la comunicació d'elements exteriors a la xarxa gràcies a les interfícies que proporciona.

La connectivitat entre les aplicacions i el «Application Server» Es pot realitzar mitjançant diferents mecanismes:

MQTT Protocol

Les aplicacions es connecten al «Application Server» Utilitzant el protocol MQTT [49]. «The Things Stack» actua com a «MQTT Broker» (Servidor) per a poder interactuar amb les aplicacions.

El format dels missatges és JSON.

HTTP Webhooks

S'utilitzen els HTTP Webhooks per a la interacció amb les aplicacions.

Integracions

Es poden utilitzar integracions i crear-ne de pròpies per al desenvolupament d'aplicacions.

6.1.4. Identity Server

Aquesta part del servidor de la xarxa s'ocupa de la identificació dels seus elements: Gateways, dispositius, aplicacions, usuaris, organitzacions... També controla l'accés al control d'elements de la xarxa amb l'API Keys.

És una part que està controlant les altres parts dels servidors com el «Gateway Server», «Network Server», «Application Server»...

6.1.5. Join Server

És la part que s'encarrega de la generació de les claus d'autenticació pel protocol LoRaWAN que s'utilitzaran en el «Network Server» i «Application Server». Servirà per a registrar nous dispositius i gestionar-los.

També s'encarrega de gestionar les sessions dels dispositius a la xarxa.

6.2. Registre de Gateways i dispositius

El registre de gateways i dispositius és semblant al de la versió 2 del servidor «The Things Stack». S'ha d'accedir a la consola de «The Things Stack» V3.

6.2.1. Registre de dispositius

Per a registrar un dispositiu, primer s'ha de registrar una aplicació; el dispositiu s'haurà de registrar dins de l'aplicació. El registre d'un dispositiu es pot realitzar de dues maneres: OTAA (Over The Air Activation) o ABP.

L'activació dels dispositius, es realitza en el moment que el dispositiu fa l'operació de JOIN a la xarxa.

Over The Air Activation (OTAA)

Després d'haver creat l'aplicació s'haurà d'anar a l'opció de registre de dispositiu. Els paràmetres necessaris per al registre d'un nou dispositiu són els següents:

- Device ID: és la ID del dispositiu, ha de ser única pels dispositius dins d'una APLICACIÓ.
- Device EUI: és l'identificador del dispositiu, que tindrà gravat en el firmware consta de 8 bytes.
- App Key: es generarà de forma automàtica.
- App EUI: s'ha de posar el EUI de l'aplicació que tindrà accés al dispositiu.

Es necessitarà l'App Key posteriorment per a activar el dispositiu.

ABP

Aquesta opció d'activació és la menys utilitzada perquè s'ha de programar un protocol per a la generació de les claus de sessió amb el dispositiu.

Per a triar aquesta opció, s'ha d'anar a la configuració del dispositiu (TTN Console) o canviar el mètode d'activació.

6.3. Aplicacions

La gestió d'aplicacions serveixen per a poder comunicar les aplicacions amb els dispositius. Es poden crear i a través de la consola de la xarxa i se li assigna una ID i un propietari. Les aplicacions que estaven per a la versió 2 de la xarxa, s'hauran d'adaptar per a la versió 3.

Les aplicacions serveixen per a gestionar l'accés i les comunicacions amb dels dispositius amb elements externs, serveis, interfícies a la xarxa.

6.3.1. Integracions

Les integracions es poden crear a l'aplicació, des de la consola. Serveixen per al processament de les dades de forma externa a la xarxa.

Hi ha diferents alternatives d'integracions o sistemes per a comunicar les aplicacions amb la xarxa TTN.

- MQTT: És un protocol que permet gestionar una aplicació a través d'una api i missatges. Hi ha una llibreria de Python. El servidor d'aplicacions exposa un servidor MQTT amb el que es pot interactuar de forma externa a través d'un MQTT Client.
- Webhooks: Permet la interacció amb la xarxa utilitzant el protocol Web (HTTPS) i una api. Consisteix en la creació d'un endpoint que rebrà les peticions HTTPS del «Application Server».

7. Llistat de components triats pel projecte

En aquest capítol es diran els components que s'han triat per a realitzar les proves del projecte.

7.1. Gateway Microtik Lora 868Mhz

És un Gateway desenvolupat per la companyia Microtik. La referència del producte desenvolupada pel fabricant «wAP LR8 kit» opera en els rangs de freqüències de 868 MHz. Algunes de les característiques són les següents [36]:

- Connector Ethernet.
- Alimentació per Ethernet o per DC de 9 a 30 V.
- consum màxim de 7W.
- Connectivitat WiFi (802.11b/g/n).
- Tipus de memòria no volàtil FLASH.

7.2. Antena pel Gateway de Microtik

Serà necessària una antena externa pel Gateway per poder utilitzar la comunicació LoRa amb els dispositius.

- Antena omnidireccional
- Pot operar en el rang de freqüències d'entre 824 i 960 MHz.
- Sistema de subjecció a la paret.

7.3. TTN Node Sensor

És un sensor que està integrat per a la connexió a una xarxa o comunicació LoRaWAN. Està desenvolupat per la companyia «The Things Industries» i dissenyat per a realitzar les primeres proves amb la xarxa «The Things Network».

- Sensor de temperatura.
- Acceleròmetre digital.
- Sensor de llum.
- Polsador i LED.
- Freqüències de 868 MHz.

7.4. Heltec ESP32 + LORA SX1276 868Mhz

Consta d'una placa de desenvolupament amb una unitat de processament ESP32. Conté el mòdul de modulació LoRa SX1276 que opera a les freqüències de 868 MHz.

Serà utilitzat per a realitzar proves amb el protocol LoRaWAN. S'ha triat aquest component gràcies a la seva bona documentació [37] comparat amb alternatives d'altres fabricants.

8. Preparació i configuració dels dispositius

En aquest capítol, s'aprofundirà en les eines de cada dispositiu mencionat i triat en el capítol anterior per a preparar-los i configurar-los.

Encara que en aquest capítol es descriu com connectar els dispositius a la xarxa TTN, també hi ha un capítol on es realitzarà la connexió i registre de dispositiu de manera pràctica a la xarxa TTN (V3). Aquesta configuració per la xarxa «The Things Network» en les següents seccions estarà enfocada en la configuració interna dels dispositius i com obtenir els diferents tipus de paràmetres per a la connexió.

8.1. The Things Node

En aquesta part, es prepararà el node per a connectar-lo a la xarxa a través d'un gateway. També es descriuran els passos seguits per a realitzar aquesta part.

S'ha de descollar la carcassa amb què ve el dispositiu i extreure la PCB amb per a poder connectar-hi el cable micro-USB per programar-lo.

8.1.1. Preparació de l'entorn de software

Llibreria TTN per l'Arduino IDE

Primer de tot, s'ha d'instal·lar l'Arduino IDE i descarregar les llibreries de «The Things Network» [38]. Aquestes llibreries es poden trobar des del gestor de llibreries a l'IDE d'Arduino buscant «TheThingsNetwork» i «TheThingsNode».

Hi ha disponibles una sèrie d'exemples per a carregar al dispositiu.

Agregació de la placa «SparkFun Pro Micro»

S'ha d'afegir la placa «SparkFun Pro Micro» per a poder pujar els programes al dispositiu.

S'ha d'accedir a la pestanya «*Fitxer* > *Preferències*» i agregar la url a l'apartat «Gestor de URLs addicionals de targetes»: https://raw.githubusercontent.com/sparkfun/Arduino_Boards/master/IDE_Board_Manager/package_sparkfun_index.json

Després, s'ha de descarregar des del gestor de targetes: «*Eines* > *Placa* > *Gestordetargetes*» i buscar «SparkFun AVR Boards».

8.1.2. Trobar el EUI del dispositiu

Per a poder registrar el dispositiu a la xarxa és necessari el EUI del dispositiu. Per a trobar-lo, s'haurà de carregar l'exemple «DeviceInfo» que està disponible a la llibreria per l'Arduino IDE.

S'haurà de modificar el «define» que defineix el «freqPlan». S'ha de substituir la part del codi on posa «REPLACE_ME» per la banda de freqüència corresponent a la nostra regió «TTN_FP_EU868».

Els passos per a registrar el dispositiu a una aplicació de la xarxa TTN es troben al punt 8.2 d'aquest document.

S'ha de seleccionar La placa SparkFun Pro Micro i el microcontrolador ATmega32U4 (3.3V, 8MHz) abans de pujar el codi. Posteriorment, seleccionar el port, pujar el codi i mirar el monitor del port sèrie.

S'ha de buscar i guardar el DevEUI per a registrar el dispositiu a la xarxa.

8.2. Heltec ESP32 + LORA SX1276 868Mhz

En aquesta secció, es veurà com s'han fet algunes proves amb les plaques de desenvolupament amb microcontrolador ESP32 i LoRa SX1276.

Per a realitzar les proves, s'ha utilitzat la documentació de la placa de desenvolupament [37].

8.2.1. Preparació de l'entorn de software d'Arduino IDE

Per a poder realitzar la instal·lació de les eines necessàries per a treballar amb el dispositiu, són necessaris el programa Arduino IDE i un cable USB.

Instal·lació del development framework

Per a instal·lar tot el necessari, s'ha d'iniciar l'Arduino IDE i anar a la finestra «Fitxer > Preferències». Després a l'apartat «Gestor d'URLs addicionals de targetes» posar la següent URL (Pot ser que canviï i retorni un error, en aquest cas s'ha de consultar la documentació [37]):

```
https://github.com/Heltec-Aaron-Lee/WiFi_Kit_series/releases/download/0.0.5/package_heltec_esp32_index.json
```

Posteriorment, s'ha d'obrir la pestanya «Eines > Placa > Gestor de Places». Després, buscar el paquet «Heltec ESP32 Series Dev-boards» i instal·lar-lo.

També hi ha disponible el codi font del framework de desenvolupament i informació de la placa [39].

Cal instal·lar la llibreria serial de python (pyserial).

Instal·lació de les llibreries

Per a instal·lar les llibreries que venen amb el paquet de desenvolupament, s'ha d'obrir la finestra «*Programa > IncloureLlibreria > AdministrarLlibreries*».

S'ha de buscar «Heltec ESP32» i instal·lar la llibreria per la placa de desenvolupament.

Configuració dels paràmetres de la placa

Per a poder pujar un programa a la placa de desenvolupament, s'ha de configurar el IDE d'Arduino de manera correcta.

S'ha d'obrir la finestra «Eines» i configurar alguns dels paràmetres que fan referència a la placa amb què estem treballant.

- Placa: WiFi LoRa 32(V2).
- Upload Speed: 921600.
- CPU Frequency: 240MHz (WiFi/BT).
- Core Debug Level: None.
- PSRAM: Disabled.
- LoRaWAN Region: REGION_EU868 (Triar la regió on s'està situat).
- LoRaWAN Debug Level: None.
- Port: El port a què està connectat el dispositiu.

8.2.2. Càrrega d'un programa d'exemple

També el que es pot fer és carregar un programa d'exemple per a comprovar que es pot programar la placa de desenvolupament o per treballar des d'un codi d'exemple i modificar-lo.

Els programes d'exemples es poden obrir des de la pestanya «*Fitxer > Exemples > HeltecESP32Dev – Boards*». El programa que ve carregat de fàbrica és el «*Wi-Fi_LoRa_32FactoryTest*».

8.2.3. dispositiu LoRaWAN

En aquesta secció es veurà com poder utilitzar la placa desenvolupament com a dispositiu LoRaWAN i provar de connectar-lo a la xarxa TTN [40].

Requisits inicials

Són necessàries els següents components:

- Arduino IDE.
- Framework de desenvolupament d'Arduino per les plaques Heltec WiFi_Kit_series.
- Un Gateway connectat a un servidor LoRaWAN.
- El dispositiu, en aquest cas la placa de desenvolupament.
- La Llibreria LoRaWAN per la placa de desenvolupament.

Instal·lació de la Llibreria LoRaWAN

Per a carregar la llibreria a l'Arduino IDE s'ha de descarregar, en format comprimit (zip), des del repositori [41].

Un cop s'ha descarregat, s'ha d'anar a l'Arduino IDE i obrir la pestanya «Programa > IncloureLlibreria > AfegirLlibreria.ZIP» i seleccionar el fitxer descarregat. Cal comprovar que la Llibreria «ESP32_LoRaWAN» està en la llista.

També s'inclouen alguns codis d'exemple per a connectar un dispositiu.

8.2.4. Connexió a la xarxa TTN

Per a connectar el dispositiu a la xarxa TTN cal carregar l'exemple «OTAA» que serà el mètode d'activació del dispositiu a la xarxa.

El registre del dispositiu a la xarxa es realitzarà des de la consola i la seva explicació es pot trobar en el punt 8.3 d'aquest document.

8.2.5. Comunicació LoRa punt a punt entre dos dispositius

En aquesta secció es realitzarà una comunicació entre dues plaques de desenvolupament Hektec per modulació LoRa. En aquest cas, es partirà dels programes d'exemple de la llibreria de la placa «LoRaReceiver» i «LoRaSender», s'anirà modificant per a realitzar una sèrie de proves.

En el repositori de la llibreria hi ha l'API [43] per a utilitzar les funcionalitats del mòdul LoRa de la placa i també la seva inicialització [42].

8.2.6. Programació en C

Programació del microcontrolador: ESP-IDF

Per a programar el microcontrolador amb C s'haurà de fer ús del framework de desenvolupament «ESP-IDF» que permet pujar aplicacions al microcontrolador [44].

A la documentació hi ha disponibles documents tècnics i datasheets del ESP32.

Llibreries LoRa i LoRaWAN

Hi ha disponible un repositori [45] que implementa en C les funcionalitats necessàries per a utilitzar el modulador LoRa i també implementa la capa LoRaWAN. Aquest repositori ha estat desenvolupat de manera conjunta amb LoRa Alliance i semtech. Es podria utilitzar per adaptar-lo al microcontrolador amb què es vulgui utilitzar.

El codi és compatible amb els integrats «LoRa RF modulators»:

- SX1272.
- SX1276.
- SX126X.
- LR1110.

A la documentació del repositori [46] es mostra com fer el Portat a la plataforma de Hardware que es vulgui utilitzar. Es pot utilitzar el repositori de Heltec[41] per veure com han realitzat aquesta adaptació i utilitzar-la o realitzar-la des de zero a partir del repositori oficial [45].

8.2.7. Pujant un programa o projecte al dispositiu amb ESP-IDF (C Programming Language)

En aquesta secció s'explicarà com programar el microcontrolador esp32 amb un projecte per ESP-IDF.

Projecte

El primer que s'ha de tenir és un projecte per a poder pujar a la placa de desenvolupament amb un microcontrolador esp32. Quan s'instal·la el programari ESP-IDF també hi ha uns projectes d'exemple que es poden utilitzar del plantilla o per pujar-los al microcontrolador.

Per començar es pot copiar el projecte que se situa a «esp-idf/examples/get-started/hello_world/» a un directori on poder treballar.

Configuració

Recordar que a la documentació [44] hi ha una guia per a instal·lar l'ESP-IDF. És important també instal·lar tots els paquets necessaris per al seu funcionament [44] (Step 1).

Prèviament s'ha d'executar el shell script amb la comanda «./export.sh» i després anar al directori de treball. Això s'haurà de realitzar cada cop que es vulgui treballar amb l'ESP-IDF en un nou terminal.

S'han d'executar les següents comandes al terminal on s'està treballant:

```
$ cd <<PATH_TO_WORKING_DIR>>
$ idf.py set-target esp32
$ idf.py menuconfig
```

Des del «menuconfig» es poden configurar algunes opcions pel compilador i pel procés de pujada a la placa.

Per a compilar el projecte s'ha d'executar la comanda:

```
$ idf.py build
```

Finalment, per pujar el projecte al microcontrolador, s'ha d'executar la següent comanda.

```
$ idf.py -p PORT [-b BAUD] flash
```

S'ha de posar el «port» on està connectada la placa de desenvolupament i el «baudrate» al que es pujarà el programa.

També es pot crear un nou projecte amb la comanda:

```
$ idf.py create-project --path <<PATH>> <<PROJECT_NAME>>
```

Per a més informació veure la secció «Build System» de la documentació [44].

8.2.8. Resum de les implementacions

- Programació amb l'entorn de desenvolupament per Arduino IDE de Heltec (fabricant de la placa)
- Programació de l'aplicació amb «C Programming Language» adaptant les llibreries de Sempitech per l'integrat SX1276 i amb ESP-IDF.

8.3. Gateway

8.3.1. Instal·lació i configuració inicial

El primer que s'ha de fer és alimentar el Gateway amb la font que ve a la caixa del mateix i connectar-lo a internet a través d'un cable Ethernet.

El gateway generarà un punt d'Accés WiFi i s'haurà de connectar un ordinador per a poder configurar el Gateway.

S'ha d'accedir a la xarxa WiFi que proporciona el Gateway (ve sense contrasenya de fàbrica). Després, des del navegador s'ha de connectar a l'adreça ip 192.168.88.1. El primer cop, no demanarà contrasenya a la configuració s'haurà d'aplicar una per la xarxa WiFi i per l'accés a la configuració

Hi ha un paper amb les credencials a la capsa del Gateway que s'ha utilitzat per a les proves.

8.3.2. Configuració LoRaWAN del Gateway

Per accedir a la configuració LoRaWAN del Gateway [48] s'ha d'anar al menú «WebFig» pel botó de dalt a la dreta. Posteriorment, anar al menú LoRa. També es pot consultar la documentació oficial del fabricant [54] per poder veure totes les opcions de configuració de la comunicació LoRa.

Configuració del servidor TTN

Per a configurar el servidor d'accés a la xarxa TTN V3:

- 1) Anar a la secció «Servers».
- 2) Registrar un Nou Servidor.
- 3) Name: TTN-EU-V3
- 4) Addr: eu1.cloud.thethings.network
- 5) Els dos Ports: 1700
- 6) Configurar el guany de l'antena.

Per a la configuració del Gateway LoRa:

- 1) Anar a la secció «Devices».
- 2) Pulsar el dispositiu que hi ha a la llista (no es poden fer canvis a la configuració quan està habilitat).
- 3) Apuntar el «Gateway Id».
- 4) Triar el Freq. Plan Adequat.
- 5) Marcar com a «Public Network».
- 6) Triar el servidor TTN-EU-V3 que s'ha creat en el pas anterior.

Només es pot assignar un servidor per «dispositiu» (7.3.2). Cal destacar, que la configuració no permet afegir cap altre dispositiu LoRa i per tant, no permet enviar dades a dos servidors alhora.

Configuració dels canals LoRaWAN del Gateway

Serà important, tenir en compte que es pot configurar aspectes dels canals LoRaWAN del gateway per a modificar diferents valors i paràmetres LoRa [54].

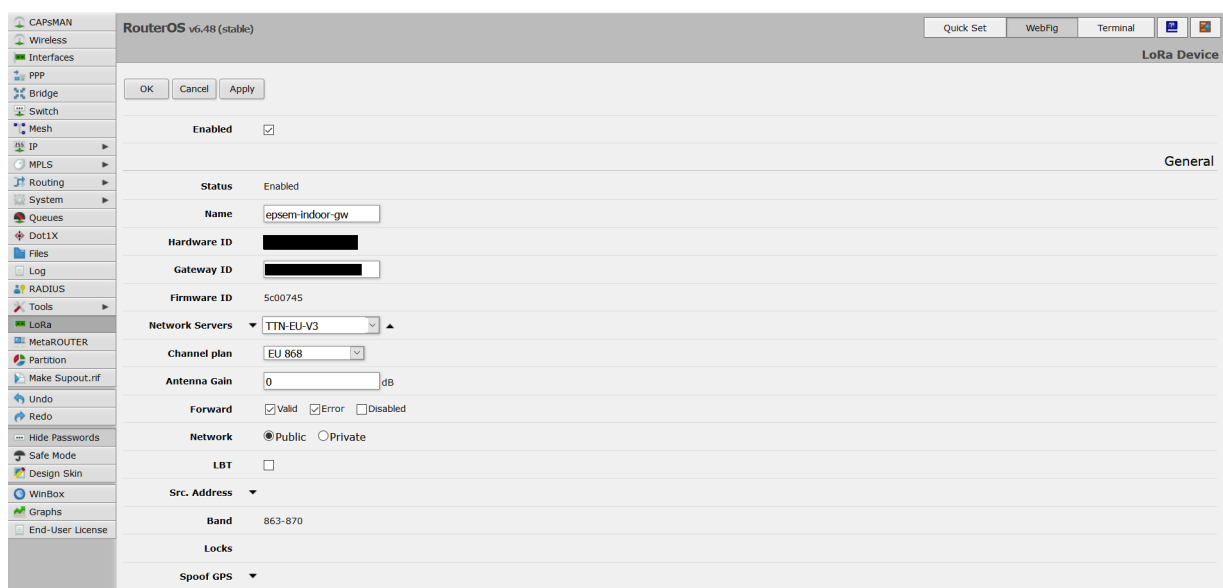


Figura 8.1.: Configuració LoRaWAN del Gateway Microtik

9. Connexió a la xarxa TTN V3

En aquesta secció es veurà com configurar els Gateways i dispositius per connectar-los a la xarxa TTN V3 des de la consola [47].

9.1. Registre del Gateway a la xarxa

En aquesta secció es descriurà el procés a seguir per a registrar el Gateway des de la consola de la xarxa TTN V3 [47]. També hi ha una documentació per a configurar el Gateway de Mikrotik [48].

Primer de tot, s'ha d'accedir a la consola de la xarxa TTN V3 [47]:

- 1) Afegir un nou Gateway (botó «Add gateway»).
- 2) Posar-li una ID.
- 3) Posar-li la EUI que s'ha extret de la configuració del gateway.
- 4) Completar altres camps com la visibilitat de Gateway o el «Freq. Plan».

Un cop creat, si tot ha anat bé, ha d'aparèixer l'estat del Gateway com a «connected».

9.1.1. Registre del dispositiu The Things Node a la xarxa

Per a registrar el dispositiu a la xarxa s'ha de crear una aplicació i registrar-lo.

S'haurà de generar prèviament una «appEUI» des de la consola de control de l'aplicació on es registrarà el dispositiu.

Posteriorment, es procedirà al registre d'un nou dispositiu omplint amb els valors necessaris en els apartats de «deviceEUI» i «appEUI».

Un cop registrat el dispositiu a l'aplicació s'haurà generat una «App Key» que s'haurà de pujar al dispositiu juntament amb la «app EUI» per a completar el procés de registre de dispositiu.

9.1.2. Connexió del dispositiu a la xarxa

En aquesta part es carregarà un codi d'exemple de la llibreria TTN per L'IDE d'Arduino que permetrà que es connecti el dispositiu a la xarxa i envii dades a l'aplicació on està registrat.

Applications > provaenricgarciadiaz > 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.
the-things-node

Device EUI
The device EUI is the unique identifier for this device on the network. You can change the EUI later.
00 04 A3 0B 00 EB 43 10 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

App EUI
70 B3 D5 7E D0 04 05 82

Cancel Register

Figura 9.1.: Registre d'un dispositiu a la xarxa TTN

S'haurà de carregar el codi d'exemple «Basic» i modificar alguna part del codi. S'ha d'anar a la finestra «Fitxer > Exemples > TheThingsNode > Basic».

Abans de pujar el codi s'ha de modificar les definicions «appEui» i «appKey». També comprova que el «Frequency Plan» és el correcte.

9.2. Registre d'un dispositiu Heltec ESP32 LoRa

En aquesta secció es veurà com registrar un dispositiu a la xarxa.

El primer que s'ha de fer és crear una aplicació i després afegir un nou dispositiu. Cal triar la configuració manual del dispositiu per a la placa de desenvolupament. En el cas que es tingui una marca de la llista també es pot fer des del llistat de dispositius LoRaWAN.

S'obrirà un menú per a registrar un dispositiu. Els passos més importants són els següents:

- 1) Triar l'opció d'activació OTAA.
- 2) LoRaWAN version: s'ha de consultar el manual de la llibreria del dispositiu. Per la placa és MAC v1.0.2.

- 3) Donar-li al botó «Start».
- 4) S'ha de posar una «APP ID» i «DEV ID» que han de ser la mateixa que hi ha configurada a la placa.
- 5) Triar Freq. plan.
- 6) Generar o apuntar una AppKey.

The screenshot shows the TTN V3 device registration interface. At the top, there are two tabs: "From The LoRaWAN Device Repository" and "Manually". Below the tabs, there are three numbered steps: 1. Basic settings (End device ID's, Name and Description), 2. Network layer settings (Frequency plan, regional parameters, end device class and session keys), and 3. Join settings (Root keys, NetID and kek labels). The "Network layer settings" step is currently active. It contains three dropdown menus: "Frequency plan" (set to "Europe 863-870 MHz (SF9 for RX2 - recommended)"), "LoRaWAN version" (set to "MAC V1.0.2"), and "Regional Parameters version" (set to "PHY V1.0.2 REV A"). Below these are checkboxes for "LoRaWAN class capabilities": "Supports class B" (unchecked) and "Supports class C" (checked).

Figura 9.2.: Registre de dispositiu a una aplicació a la xarxa TTN V3

En el cas de la placa de desenvolupament Heltec, el DevId, AppId i AppKey s'ha de posar des del codi quan es puja el programa. Es pot generar les Id des de la consola de la versió anterior.

- DevId: Proporcionat pel fabricant o s'ha de trobar la manera de generar-lo en cas contrari.
- AppId: Pot ser tot 0, generat de forma aleatòria o que el fabricant l'especifiqui (poc comú).
- AppKey: Es pot generar des del registre del dispositiu a la xarxa TTN.

Si tot ha sortit bé, s'ha de veure l'estat «connected» del Gateway i les dades que envia el dispositiu a la xarxa des de la consola (submenú de l'aplicació on està registrat el dispositiu «Live Data»).

9.3. Aplicacions TTN

9.3.1. Introducció

Des de la consola [47] de la xarxa TTN es pot accedir a les aplicacions que disponibles per l'usuari de la xarxa. Permeten gestionar els dispositius que tenen registrats i les seves dades juntament amb la seva comunicació. Un dispositiu només pot ser gestionat per una sola aplicació.

Les aplicacions de la consola poden fer de pont entre la xarxa TTN i els seus dispositius amb elements o serveis externs a la xarxa a través de les interfícies.

9.3.2. Integració MQTT

Paquets «mosquitto-clients» i «mosquitto-dev» disponibles al repositori de paquets de la versió «Stable» de Debian 10.

MQTT [49] és un protocol de comunicació que es basa en la subscripció i publicació de missatges cap a un Servidor (MQTT Broker). Se sol utilitzar en les comunicacions de sensors «low power».

En el cas de l'aplicació TTN, genera una interfície que actua com un «MQTT Broker» on un Client MQTT s'hi pot subscriure o publicar missatges per a interactuar amb l'aplicació.

Subscripció i publicació des de mosquitto

El paquet «mosquitto-clients» posa a disposició dos programes per a poder connectar amb un «MQTT Broker». En aquest cas es poden executar a través des d'un terminal de «BASH» amb les comandes «mosquitto_sub» i «mosquitto_pub» per subscriure's i publicar.

- Cal anar al menú de l'aplicació «*Integrations > MQTT*» i generar una «API Key».
- Es farà servir el nom d'usuari, contrasenya (API Key) i adreça IP que es mostren.
- Cal saber a quin «Topic» es vol subscriure o publicar.

Els Tòpics als que es pot subscriure són els següents [50]:

- v3/{application id}@{tenant id}/devices/{device id}/join
- v3/{application id}@{tenant id}/devices/{device id}/up
- v3/{application id}@{tenant id}/devices/{device id}/down/queued
- v3/{application id}@{tenant id}/devices/{device id}/down/sent
- v3/{application id}@{tenant id}/devices/{device id}/down/ack
- v3/{application id}@{tenant id}/devices/{device id}/down/nack
- v3/{application id}@{tenant id}/devices/{device id}/down/failed

- v3/{application id}@{tenant id}/devices/{device id}/service/data
- v3/{application id}@{tenant id}/devices/{device id}/location/solved

El «tenant id» en el cas de The Things Network és ttn.

Els tòpics per publicar són els següents:

- v3/{application id}@{tenant id}/devices/{device id}/down/push

Un exemple de subscripció utilitzant «mosquitto_sub» seria de la següent manera:

```
$ mosquitto_sub -h eu1.cloud.thethings.network
-t <<v3/{application id}@{tenant id}/devices/{device id}/up>>
-u <<{application id}@{tenant id}>>
-P <<PASSWORD>> -d
```

Llibreries

Existeixen una sèrie de llibreries que implementen el protocol MQTT per diferents llenguatges de programació. Permeten la implementació de clients MQTT que poden realitzar accions de publicació o subscripció, entre altres funcionalitats, contra un «MQTT Broker».

Un exemple bastant complet seria la llibreria de Python «paho-mqtt» [51].

També existeix una llibreria de C que implementa el protocol [52].

10. Proves amb els dispositius

En aquest capítol es realitzaran una sèrie de proves amb els diferents dispositius per a testejar alguns aspectes relacionats amb la comunicació amb LoRa.

10.1. Heltec ESP32 + SX1276 LoRa

En aquesta secció es realitzaran proves amb la placa de desenvolupament Heltec descrita en capítols anteriors. També s'utilitzaran les llibreries i el framework per Arduino IDE per programar les plaques de desenvolupament.

L'objectiu d'aquestes proves és veure alguns aspectes sobre la comunicació punt a punt amb modulació LoRa.

10.1.1. Temps de transmissió

En aquesta part es veuran els temps d'execució del codi que s'encarrega de transmetre. Es veuran els temps des del transmissor. Es farà servir la llibreria proporcionada pel desenvolupador de la placa de Heltec.

També s'aniran variant alguns paràmetres de la comunicació com el Spreading Factor (SF), TxPower, o l'Amplada de Banda del senyal enviat (BW). Tot això per a veure quin efecte té en les diferents mesures de temps.

En aquesta secció es testejarà el temps que el codi de programa tardarà a transmetre variant paràmetres de la comunicació.

S'ha desenvolupat un programa que es pujarà a la placa de desenvolupament Heltec per a obtenir mostres dels temps de transmissió en funció de diferents paràmetres.

Alguns aspectes o característiques a tenir en compte de l'experiment:

- Bytes transmesos: Nombre de bytes transmesos en cada transmissió.
- Es realitzen 500 mostres per a cada experiment de transmissió.
- No es comprova si el receptor rep la informació transmesa.

Repositori: El repositori amb el codi utilitzat per als tests: <https://github.com/XicoPi/tfg-LoRa/blob/main/test/>

Full de càlcul: El full de càlcul amb les mostres es troba al següent enllaç: https://docs.google.com/spreadsheets/d/18aSeaN_7F0Z5PWYVhTgQ5_Zq8bE31yst8YMZ0zoIk3Y/edit?usp=sharing

S'han realitzat 10 mostres per cada nombre de bytes i s'ha realitzat l'experiment fins a 150 bytes:

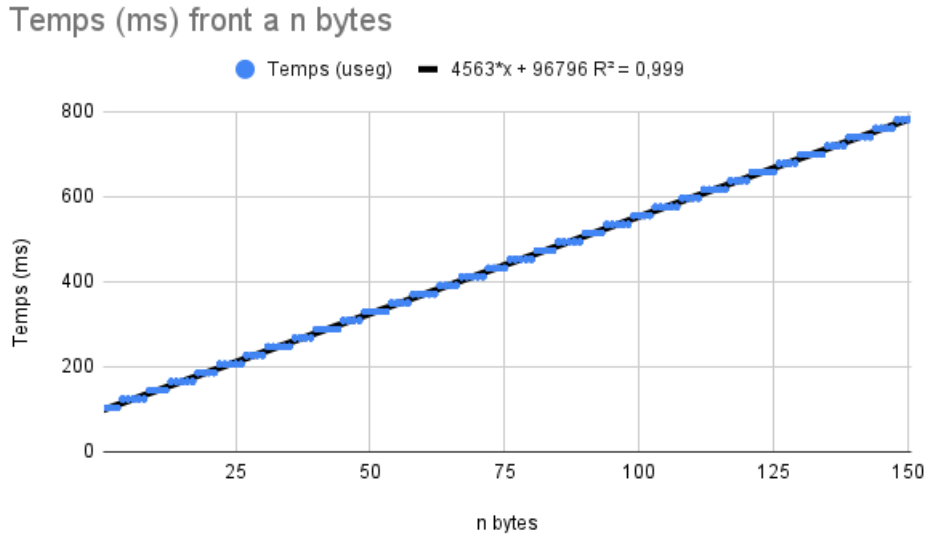


Figura 10.1.: Gràfica de dispersió del temps en funció dels bytes transmesos amb SF=9 i BW=125 kHz

Observacions: Com es pot veure, la recta de regressió indica un grau significatiu de correlació. També s'ha vist que per a transmetre el mateix número el de transmissió canvia lleugerament per a diferents transmissions encara que el valor del byte sigui el mateix.

Temps (s) front a Spreading Factor

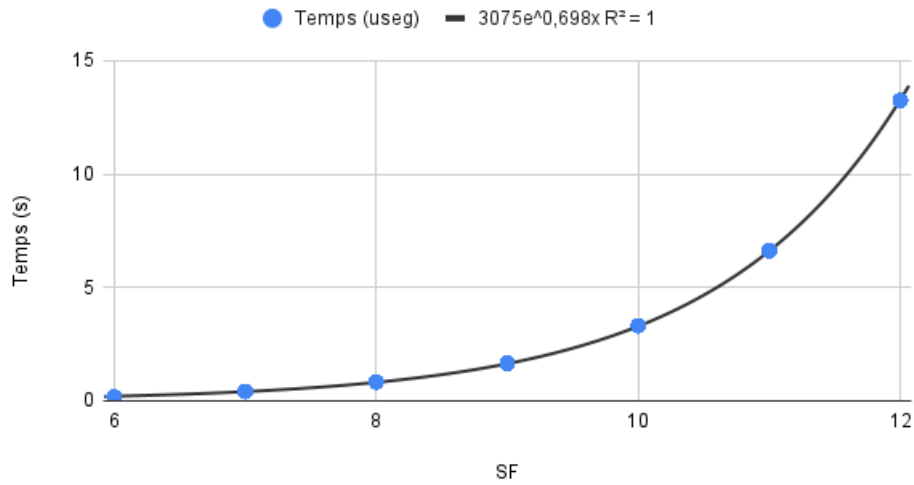


Figura 10.2.: Gràfica de dispersió del temps en funció del Spreading Factor amb BW=7,8 kHz i $n_{bytes} = 1$ Bytes

Observacions: Es pot veure una funció de dispersió de tipus exponencial. Indica que com més s'eixampla l'espectre del senyal més afecta al «data rate» del senyal disponible.

Temps (ms) front a BW(kHz)

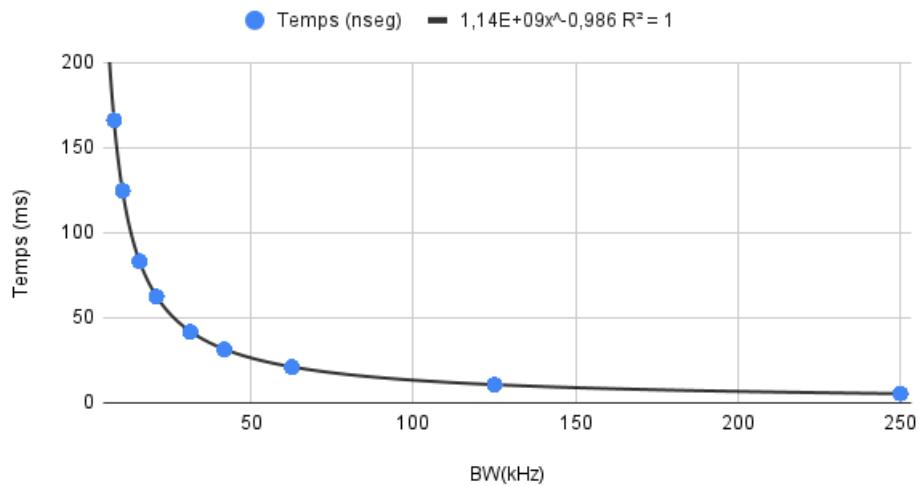


Figura 10.3.: Gràfica de dispersió del temps en funció de l'amplada de banda amb SF=6 i $n_{bytes} = 1$ Bytes

Observacions: La gràfica de dispersió indica que quan més ample és el BW del senyal, menor serà el temps de transmissió de la informació.

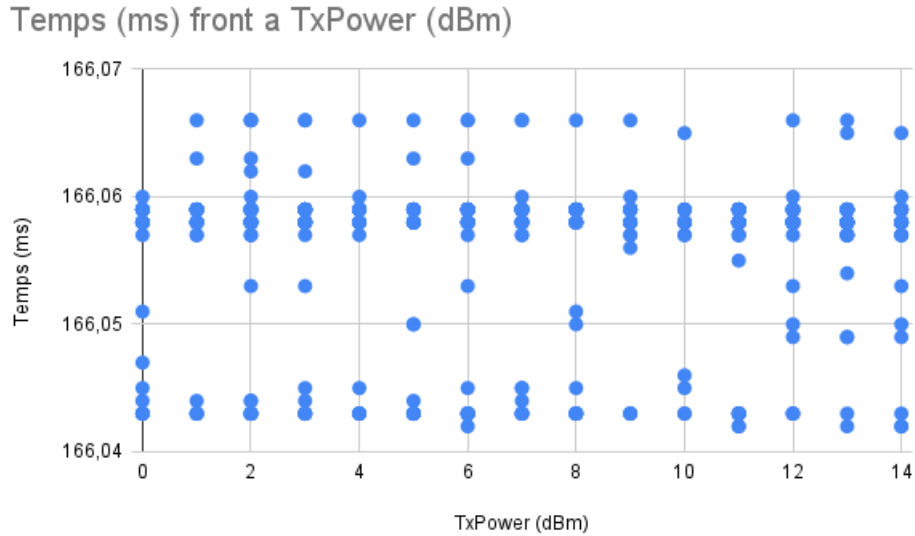


Figura 10.4.: Gràfica de dispersió del temps en funció de la potència de transmissió amb SF=6, BW=7,8 kHz i $n_{bytes} = 1$ Bytes

Observacions: Com es pot veure, no existeix correlació entre la potència i el temps de transmissió.

Comparació amb els valors teòrics

Després s'ha de comparar els temps de transmissió obtinguts, amb els teòrics obtinguts a partir de [2], són majors si només es té en compte el temps amb els bytes d'informació de «payload».

$$Rb = SF * \frac{BW * RC}{2^{SF}} \quad (10.1)$$

On Rb és el «bit rate» del senyal en (bit/s). Per «Figure 9.1» (1 Byte), SF=9, BW=125 kHz, RC=4/5 s'obté el valor teòric $T_{bit} = 4,55$ ms/bit. En canvi el valor observat és de $T_{byte_observat} \simeq 103$ ms.

L'augment del temps de transmissió és degut al preàmbul del senyal «chirp», necessari perquè el receptor pugui realitzar l'operació de desmodulat. Per tant sembla el «Time On Air» és major del que s'extreu amb el càlcul del «bit rate» teòric o especificat per l'estàndard.

Mirant el datasheet del tranceptor SX1276 [55] s'ha trobat com calcular alguns aspectes i característiques del senyal durant la seva transmissió. També s'explica com es pot calcular la mida de la «payload». On s'ha de tenir en compte també el temps que es tarda en enviar el «header», CRC, «Coding Rate». S'ha utilitzat les següents fórmules:

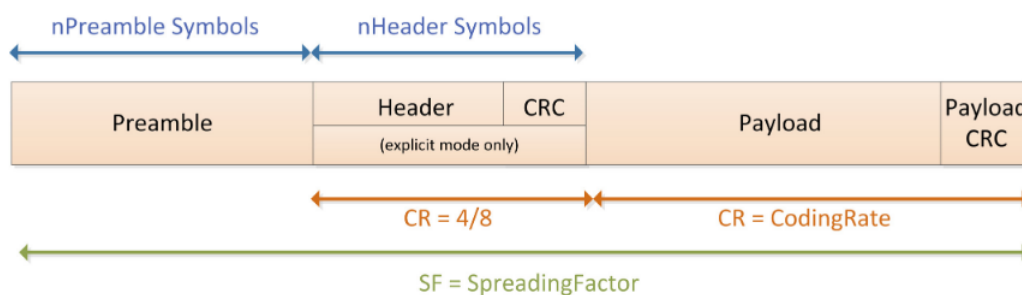


Figura 10.5.: Estructura del «LoRa Packet»

El contingut del «LoRa Packet» és el següent:

- Un preàmbul.
- Un «header» (opcional).
- «Data payload»

$$T_{sym} = \frac{1}{R_{sym}} = \frac{2^{SF}}{BW} (\text{sec/symbol}) \quad (10.2)$$

T_{sym} és el període de símbol, el temps que dura un senyal «chirp». Es calcula $T_{sym} = 4,096$ ms/symbol

$$T_{payload} = T_{sym} * \left[8 + \text{ceil} \left(\frac{8PL - 4SF + 28 + 16(CRC) - 20(IH)}{4SF} \right) * (CR + 4) \right] \quad (10.3)$$

Significat dels elements:

- PL: nombre de Bytes del «payload», pot pendre valors del 1 al 255 que és el màxim nombre de bytes d'informació per «payload».
- CR: «Coding rate».
- SF: «Spreading Factor».
- CRC: Pot pendre valors $\{0, 1\}$ que indiquen si hi ha CRC.
- IH: Indica si hi ha «Implicit Header» pot pendre per valors $\{0, 1\}$.

Fixant que $CR = 1$, $PL = 1$ Byte, $SF = 9$, $CRC = 1$, $IH = 0$ llavors $T_{payload} = 53,248$ ms.

$$T_{preamble} = (n_{preamble} + 4,25) * T_{sym} \quad (10.4)$$

Com indica a les especificacions del datasheet, el valor de $n_{preamble}$ depén del mode de «header» que s'utilitzi que per defecte és 8 (symbols/preamble).

$$T_{packet} = T_{preamble} + T_{payload} \quad (10.5)$$

Si substituïm els valors que $T_{preamble} = 50,176$ ms i juntament amb $T_{payload}$, s'obté un valor de $T_{packet} = 103,424$ ms. Valor que coincideix amb la mitjana de les mostres mesurades a la «Figure 9.1» $T_{packet_avg} = 103,695$ ms. També es pot verificar amb un calculador online del T_{packet} [56].

Finalment, també es pot veure a la gràfica de temps de transmissió en funció dels bytes enviats «Figure 9.1» com el temps de transmissió està escalonat mantenint-se constant per un cert nombre de bytes per després realitzar un salt significatiu en el temps de transmissió «Figure 9.5». Tot això, es pot veure a la fórmula «9.3» a l'operació $\text{ceil}()$ que depèn del «Spreading Factor» (SF) i del nombre de bytes per «payload» (PL). A la taula de dades es pot veure que coincideixen els salts de temps de transmissió d'igual manera que com està reflectit a la fórmula «9.3».

$$PL = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20] \quad (10.6)$$

$$(9.3) \rightarrow \text{ceil}(PL) = [1, 1, 1, 2, 2, 2, 2, 2, 3, 3, 3, 3, 4, 4, 4, 4, 4, 5, 5, 5] \quad (10.7)$$

Si ho comparem amb la «Figure 9.6», es veu que el temps de transmissió es manté constant per agrupacions de mides de «payload» quan incrementa el seu nombre bytes (PL). També coincidirà amb les agrupacions de $n_{payload}$ i $T_{payload}$ que depenen de la funció $\text{ceil}(PL)$.

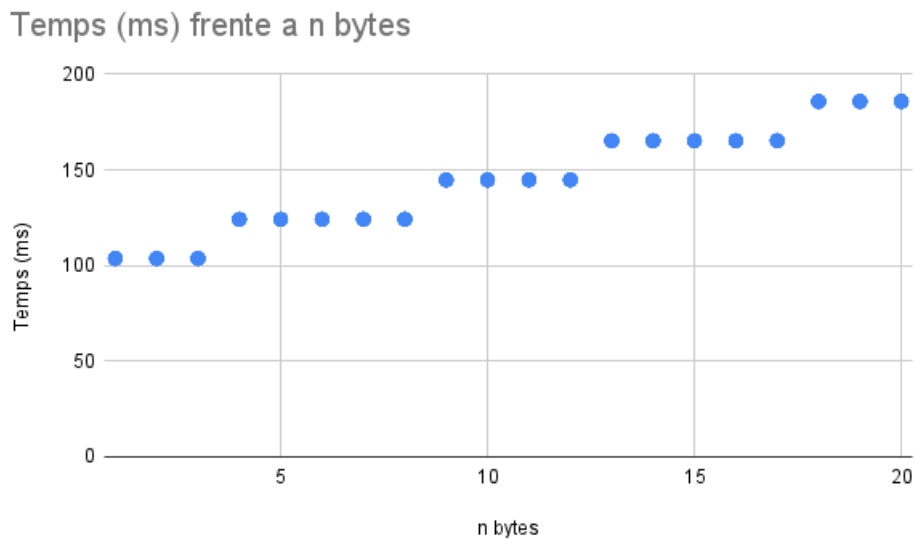


Figura 10.6.: Temps de transmissió en funció del nombre de bytes

10.1.2. Jugant amb els paràmetres LoRa

En aquesta part es faran algunes proves amb les configuracions dels diferents paràmetres per la modulació LoRa (Spreading Factor i Amplada de banda).

Per al testatge es necessitarà un dispositiu transmissor i un de receptor. L'objectiu serà veure quines configuracions permeten la comunicació efectiva entre els dispositius a una distància considerablement propera.

Les condicions dels dos elements (plaques de desenvolupament ESP32 + LoRa):

- Distància molt propera.
- Alimentació per USB.
- Les antenes que venien amb les plaques.
- Potència de transmissió: 14 dBm.

Cal destacar que els dos dispositius han de tenir la mateixa configuració perquè es puguin comunicar.

Els resultats de les proves de comunicacions són els següents:

- Valor d'amplada de banda mínim per la comunicació: 62.5 kHz.
- Valor de Spreading Factor mínim per la comunicació: 7.

Després de realitzar les proves s'ha pogut comprovar que per un amplada de banda inferior a 62.5 kHz no es pot fer efectiva la comunicació. També s'ha trobat que per un Spreading Factor de 6 tampoc es poden comunicar els dispositius entre ells.

Finalment, s'ha determinat que es pot fer qualsevol combinació de valors pels paràmetres entre 62.5kHz - 250 kHz i 7 - 12 de Spreading Factor

Els resultats es poden trobar en el següent full de càlcul: https://docs.google.com/spreadsheets/d/12q6MjqG0UDQLKqumlyijg5ayzek5IMnuk3lik_9nzCw/edit?usp=sharing

10.2. Desat de dades dels dispositius a una base de dades amb protocol MQTT

En aquesta secció es realitzaran proves per al desat de dades a una base de dades utilitzant la integració MQTT de les aplicacions de la xarxa TTN.

El repositori amb tots els fitxers amb les implementacions estan al repositori del projecte [53].

Els elements que conformen l'experiment són els següents:

- DDBB MySQL: Base de dades MySQL hostejada a la universitat.
- Mòduls python pel recull de dades: consta de dos mòduls, on es reben les dades per protocol MQTT i es desaran a la base de dades.
- TTN MQTT Integration: és la integració MQTT que permet la transmissió de dades amb el protocol MQTT.
- Node o sensor LoRaWAN: la font de la informació que s'enviarà a la xarxa.
- Gateway: per a connectar el sensor a Internet i a la xarxa TTN.

10.2.1. Mòdul python mqtt_processor

És el mòdul de python 3 que utilitza la llibreria «paho_mqtt» per subscriure's al «MQTT Broker» generat per la integració MQTT de la xarxa TTN.

Parseja el missatge rebut que està en format JSON pera poder tractar-lo com a un diccionari de python. També crida al mòdul db_utils per a pujar-lo a la DDBB.

10.2.2. Mòdul pyhton db_utils

S'encarrega d'insertar les dades dels missatges de la interfície de la xarxa TTN a la base de dades.

Defineix una classe «TTN_database» amb els mètodes per a pujar la informació dels dispositius, aplicacions i continguts dels missatges a la Base de dades.

Atributs de la classe TTN_database

- host: host on està la DDBB.
- user: nom d'usuari per a accedir a la DDBB.
- password (privat): contrasenya per a accedir a la DDBB.
- database: nom de la DDBB on es desaran les dades.

Mètodes de la classe TTN_database

- insert_app(): permet inserir l'aplicació a la base de dades.
- insert_device(): permet inserir el dispositiu a la DDBB.
- insert_uplink_msg(): permet inserir el missatge, transmès pel dispositiu a l'aplicació TTN, a la DDBB.

10.2.3. Format del missatge

El format del missatge, rebut des de la xarxa TTN a través de la seva integració MQTT, és el següent (Anotacions de python, són diccionaris parsejats a partir d'un missatge JSON):

```

Json_Msg_t = str
TTN_app_id_t = str
TTN_dev_id_t = str
ttn_app_t = {'application_id': TTN_app_id_t}

Parsed_Msg_t = {
    "end_device_ids": device_t,
    "correlation_ids": List[str],
    "received_at": str,
    "uplink_msg": uplink_msg_t
}

device_t = {
    "device_id": TTN_dev_id_t,
    "application_ids": TTN_app_id_t,
    "dev_eui": str,
    "join_eui": str,
    "dev_addr": str
}

uplink_msg_t = {
    "received_at": str,
    "session_key_id": str,
    "f_port": int,
    "f_cnt": int,
    "frm_payload": str,
    "rx_metadata": list,
    "settings": dict,
    "decoded_payload": dict,
    "consumed_airtime": str
}

msg_payload_t = {
    "received_at": str,
    "battery": int,
    "event": str,
    "light": int,
    "temperature": float
}

```

10.2.4. Esquema de la base de dades

Per a l'esquema de la base de dades, s'ha seguit el format del missatge rebut des de la xarxa TTN per a registrar la informació en taules en una DDBB de tipus SQL.

```
create table if not exists applications (  
    application_id varchar(255) primary key  
);  
  
create table if not exists devices (  
    device_id varchar(255) primary key,  
    application_id varchar(255) references applications,  
    dev_eui varchar(17),  
    join_eui varchar(17),  
    dev_addr varchar(25)  
);  
  
create table if not exists uplink_messages (  
    msg_id_time datetime,  
    device_id varchar(255) references devices,  
    session_key_id varchar(255),  
    f_port int,  
    f_cnt int,  
    frm_payload varchar(255),  
    rx_metadata TEXT(65535),  
    settings TEXT(65535),  
    consumed_airtime varchar(32),  
    primary key (msg_id_time, device_id)  
);  
  
create table if not exists node_decoded_payloads (  
    msg_id_time datetime primary key references uplink_messages,  
    battery int,  
    event varchar(16),  
    light int,  
    temperature float  
);
```

Tots els resultats estan al repositori [53].

10.3. Enviament i visualització de les dades (DDBB influxDB)

En aquest apartat, es realitzarà el desat de les dades a una DDBB de tipus influxDB per a després poder visualitzar-les des de l'aplicació grafana.

S'ha implementat un programa de python que es subscriu a la interfície MQTT de l'aplicació on estan registrats els dispositius i poder inserir les dades a la DDBB per després visualitzar-les a des del Grafana.

El mecanisme d'obtenció de les dades és el mateix que amb el de l'apartat anterior, per la interfície MQTT del servidor d'aplicació TTN.

La implementació està disponible en el repositori, el directori (tfgEnricGarcia) [53].

10.3.1. Visualització de les dades enviades

Actualment, es reben dades del dispositiu The Things Node:

- Bateria: bateria del dispositiu.
- Event: Potser interval, motion o button, depenent de si s'ha enviat les dades degut a l'interval d'un minut, un moviment detectat per l'acceleròmetre o el pulsador del dispositiu.
- Temperatura: sensor de temperatura del dispositiu.
- Light: sensor de llum del dispositiu.

10.3.2. Servei del sistema operatiu

L'element encarregat de l'optenció de les dades de la xarxa TTN i enviament a la DDBB influxDB és un servei del sistema operatiu. Aquest està escrit en python3 i permet configurar l'accés al servei MQTT de la xarxa i l'ús i registre de dispositius per a accedir a les seves dades.

Es troba al directori «tfgEnricGarcia» del repositori del projecte [53].

Fitxers de configuració

Els fitxers de configuració es troben al directori «tfgEnricGarcia/config»

- config.yml: configuració del servei del sistema.
- influx.yml: configuració de l'accés a la DDBB.
- mqtt_local.yml: configuració de la connexió MQTT i dispositius que es volen utilitzar.

El fitxer `mqtt_local.yml` té els següents elements de configuració:

- **HOST:** el servidor mqtt on es subscriurà el programa.
- **PORT:** el port del servei mqtt.
- **USER:** nom de l'aplicació ttn (nom d'usuari mqtt).
- **PWD:** MQTT password.
- **DEVICE_IDS:** diccionari de python on hi ha les «id» dels dispositius com a «key» i el fitxer de python on està definit el callback de tractament del missatge del dispositiu.

Un exemple de valor pel camp de configuració «DEVICE_ID» seria el següent:

```
DEVICE_IDS: {  
  "ttn-node-dev-1": "callbacks/ttnNodeDevCallbackDef.py",  
  "heltec-esp32-lora": "callbacks/heltecEsp32CallbackDef.py"  
}
```

10.3.3. Sistema de callbacks

Com s'ha dit anteriorment, cada dispositiu ha de tenir associat un callback on rebi per paràmetre un missatge («decoded_payload»). Si aquest element està en JSON, com tot el missatge procedent de TTN, serà passat com a paràmetre cap al callback com un diccionari de python. Retorna un diccionari de python amb el nom de la mesura com a «key» i el seu valor com a «value».

No poden haver dos callbacks en un mateix mòdul a la configuració. El callback ha d'estar definit amb el nom «devCallback(msg)».

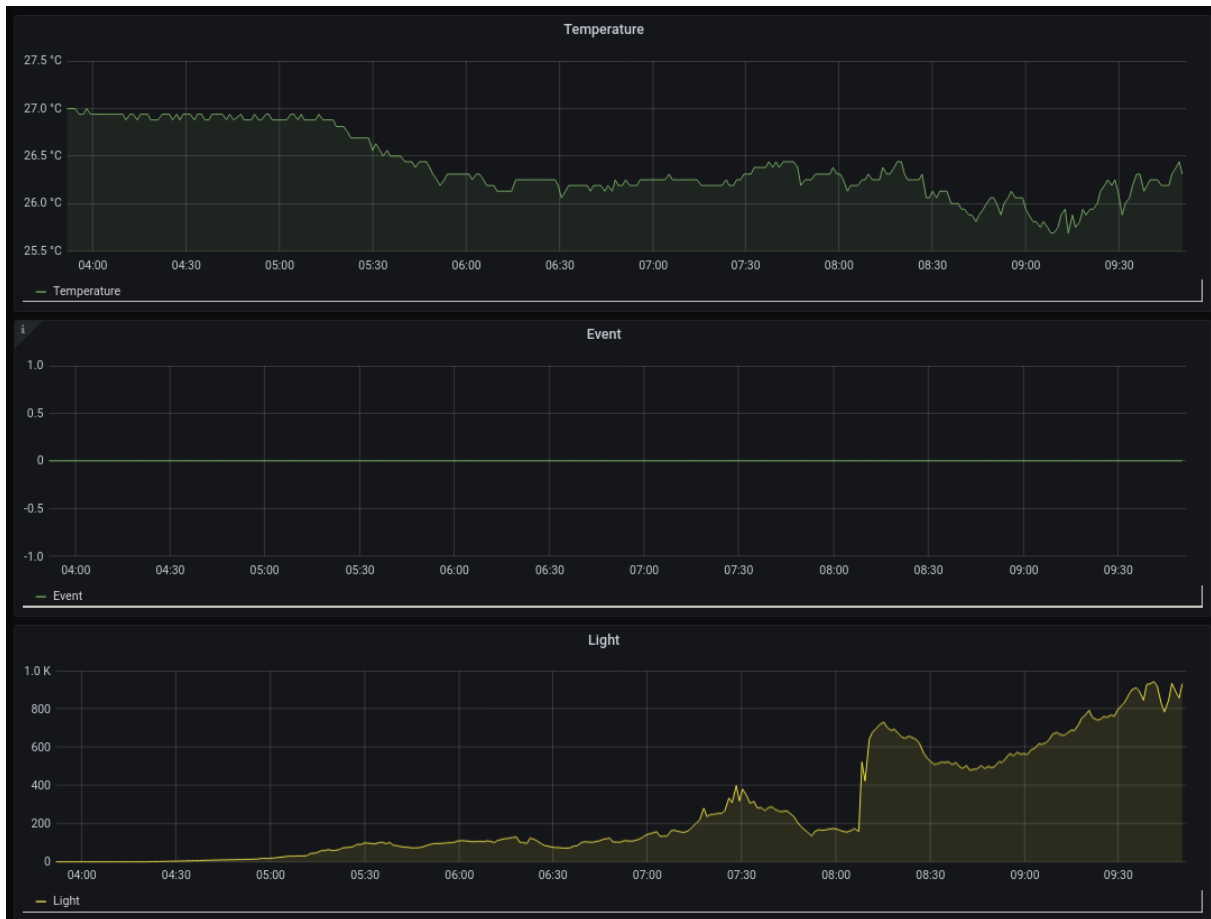


Figura 10.7.: Gràfica de les dades del sensor de temperatura, event i llum

10.4. Heltec ESP32 + Sensors externs

10.4.1. Sensor de Temperatura DS18B20

Llibreries de l'Arduino IDE utilitzades:

- OneWire.
- DallasTemperature.

10.4.2. Acceleròmetre ADXL345

És un acceleròmetre que utilitza un bus de comunicació i2c i spi per a la seva configuració i mesura de les dades.

Llibreries d'Arduino:

- 1) Adafruit ADXL345.
- 2) Adafruit Unified sensor.

De la llibreria d'Arduino Adafruit ADXL345 s'ha hagut de modificar el codi font perquè pogués accedir al sensor pel bus de comunicació i2c o SPI. El codi està adjuntat al repositori del projecte [53]. El que s'ha realitzat és l'especificació dels pins sda i scl de la placa.

S'ha hagut d'anar al directori de libreria d'Arduino i modificar el fitxer: «*Arduino/libraries/Adafruit_ADXL345/Adafruit_ADXL345_U.cpp*». S'ha hagut de modificar línia 3 del mètode «begin» de la classe «*Adafruit_ADXL345_Unified*»:

```
bool Adafruit_ADXL345_Unified::begin(uint8_t i2caddr)
{
    _i2caddr = i2caddr;

    if (_i2c)
        Wire.begin(4, 15);
    /* Modificacio Especificacio :
    dels pins SDA i SCL del bus i2c.*/
    else {
        pinMode(_cs, OUTPUT);
        digitalWrite(_cs, HIGH);
        pinMode(_clk, OUTPUT);
        digitalWrite(_clk, HIGH);
        pinMode(_do, OUTPUT);
        pinMode(_di, INPUT);
    }

    /* Check connection */
    uint8_t deviceid = getDeviceID();
    Serial.println(deviceid);
}
```

```

if (deviceid != 0xE5) {
    /* No ADXL345 detected ... return false */
    return false;
}

// Enable measurements
writeRegister(ADXL345_REG_POWER_CTL, 0x08);

return true;
}

```

10.4.3. Modificació de la llibreria LoRaWAN

S'han realitzat proves de modificació alguns aspectes de la llibreria de la placa de desenvolupament «Heltec ESP32 LoRa» per a realitzar les proves amb un Spreading factor de 12. Malgrat a aquest canvi, s'ha tornat a l'estat del codi inicial perquè el dispositiu reguli els paràmetres que tenen efecte al «data rate» segons el nombre d'intents de connexió amb el Gateway.

També s'ha modificat el nombre d'intents màxims que el dispositiu pot fer a l'hora de fer un «join» i «Tx» a la xarxa TTN. Això, té efecte en les variacions que es realitzaran pels diferents paràmetres LoRa per aconseguir una connexió amb el Gateway i poder transmetre les dades.

La modificació de la llibreria es troba al repositori del projecte [53] a «*ESP32_LoRaWAN-master/src/ESP_LoRaWAN.cpp*»:

- Funció: «static void OnTxNextPacketTimerEvent(void)».
- Mètode de la classe «LoRaWANClass»: «join()».

```

static void OnTxNextPacketTimerEvent( void )
{
    MibRequestConfirm_t mibReq;
    LoRaMacStatus_t status;

    TimerStop( &TxNextPacketTimer );

    mibReq.Type = MIB_NETWORK_JOINED;
    status = LoRaMacMibGetRequestConfirm( &mibReq );

    if( status == LORAMAC_STATUS_OK )
    {
        if( mibReq.Param.IsNetworkJoined == true )
        {
            deviceState = DEVICE_STATE_SEND;
            NextTx = true;
        }
    }
}

```

```
    }
    else
    {
        // Network not joined yet. Try to join again
        MlmeReq_t mlmeReq;
        mlmeReq.Type = MLME_JOIN;
        mlmeReq.Req.Join.DevEui = DevEui;
        mlmeReq.Req.Join.AppEui = AppEui;
        mlmeReq.Req.Join.AppKey = AppKey;
        mlmeReq.Req.Join.NbTrials = 208; <-----

        if( LoRaMacMlmeRequest( &mlmeReq ) == LORAMAC_STATUS_OK )
        {
            deviceState = DEVICE_STATE_SLEEP;
        }
        else
        {
            deviceState = DEVICE_STATE_CYCLE;
        }
    }
}

void LoRaWanClass::join()
{
    if( overTheAirActivation == true )
    {
        Serial.println("joining...");
        MlmeReq_t mlmeReq;

        mlmeReq.Type = MLME_JOIN;

        mlmeReq.Req.Join.DevEui = DevEui;
        mlmeReq.Req.Join.AppEui = AppEui;
        mlmeReq.Req.Join.AppKey = AppKey;
        mlmeReq.Req.Join.NbTrials = 208; <-----

        if( LoRaMacMlmeRequest( &mlmeReq ) == LORAMAC_STATUS_OK )
        {
            deviceState = DEVICE_STATE_SLEEP;
        }
        else
        {
            deviceState = DEVICE_STATE_CYCLE;
        }
    }
}
```

```

else
{
    ...
}
}

```

10.4.4. Dades enviades

Les dades enviades pel dispositiu consten d'una llista de dades, separades per comes, dels sensors:

- Eix X: el valor eficaç de l'eix X de l'acceleròmetre.
- Eix Y: el valor eficaç de l'eix Y de l'acceleròmetre.
- Eix Z: el valor eficaç de l'eix Z de l'acceleròmetre.
- temp: el valor actual del sensor de temperatura.

Per al càlcul del valor eficaç s'utilitza una finestra de 32 mostres mostrejades a 100 Hz.

Les dades són rebudes al procés que intercanvia dades amb la xarxa TTN per a desar-les i visualitzar-les al grafana.

El valor eficaç dels eixos de l'acceleròmetre es realitzen utilitzant el mode FIFO del sensor ADXL345 que habilita un «buffer» de 32 mesures. Gràcies a això, es permet l'obtenció de les dades de cop sense haver d'estar massa temps amb el microcontrolador fent operacions de mostreig a 100 Hz, cosa que faria que augmentés el consum del microcontrolador.

Per a poder utilitzar el mode FIFO del sensor s'ha hagut de modificar la llibreria de l'acceleròmetre per a implementar un mètode que permetés la lectura pel bus i2c de múltiples bytes amb una sola instrucció. La lectura del «buffer» intern del sensor s'ha de fer de cop per als tres eixos (x, y, z) perquè no es sobrescrigui quan es desplacen les següents mostres del «buffer» si es realitza per comunicacions separades.

La llibreria modificada es situa al directori «*Arduino/libraries/Adafruit_ADXL345/*»:

- Definició i implementació del mètode: «*FIFO_multiByteRead(void)*» de la classe «*Adafruit_ADXL345_Unified*».
- Definició del tipus «*FIFO_regs_values_t*»

La implementació del mètode per a demanar les dades del «buffer» intern del sensor:

```

FIFO_regs_values_t Adafruit_ADXL345_Unified::FIFO_multiByteRead(void)
{
    FIFO_regs_values_t result;

```

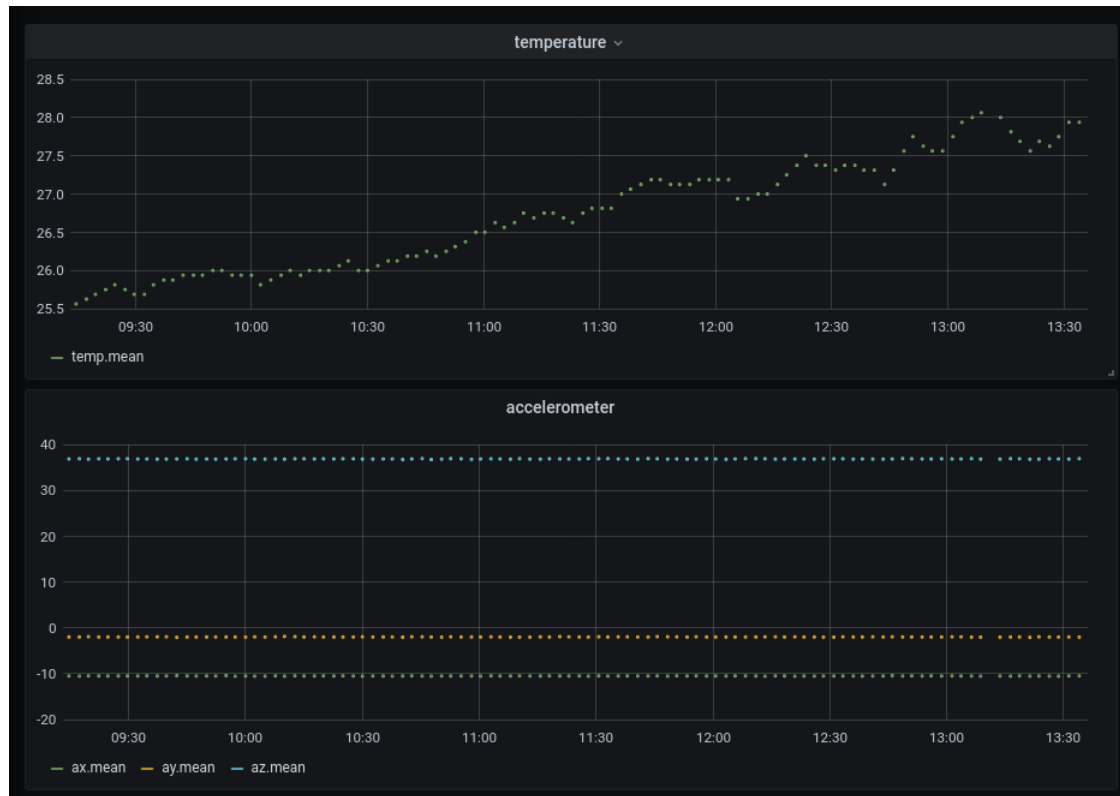


Figura 10.8.: Gràfica de les dades del sensor de temperatura i acceleròmetre (m/s^2) de la placa heltec ESP32 + LoRa

```

Wire.beginTransmission((uint8_t)_i2caddr);
i2cwrite(ADXL345_REG_DATAX0);
Wire.endTransmission(false);

Wire.requestFrom((uint8_t)_i2caddr, (uint8_t)6, (uint8_t)true);

result.x = (int16_t)(i2cread() | (i2cread() << 8));
result.y = (int16_t)(i2cread() | (i2cread() << 8));
result.z = (int16_t)(i2cread() | (i2cread() << 8));
return result;
}

```

10.5. Proves de consum amb la placa de desenvolupament Heltec ESP32 + LoRa (LoRaWAN)

També s'han realitzat algunes proves de consum de la placa de desenvolupament Heltec amb els sensors per a veure el seu comportament.

10.5.1. Gràfic de consum d'una transmissió amb SF=7

La captura s'ha realitzat des de la font d'alimentació que alimentava la placa a través del connector de la bateria.

Al gràfic es poden veure varis aspectes interessants sobre la placa.

- S'ha observat que el microcontrolador es manté en mode sleep fins que arriba l'ordre de transmetre.
- Un cop finalitza el mode sleep, el consum s'estabilitza al voltant dels 60 mA.
- Posteriorment, arriba al pic de consum de 150 mA durant el temps que transmet les dades.
- També s'observa que el microcontrolador segueix despert durant la finestra de recepció amb el valor estable, altrament, al voltant dels 60 mA.
- Finalment, s'observa l'activació del mòdul de recepció (petits pics de consum) que corresponen a la finestra de recepció del dispositiu (LoRaWAN CLASS A).
- El consum del microcontrolador en el mode sleep és aproximadament 1 mA.

Per a tenir el consum mínim del dispositiu, és necessari reduir el temps que no està en mode sleep. També, s'ha d'augmentar el temps entre transmissions al Gateway.

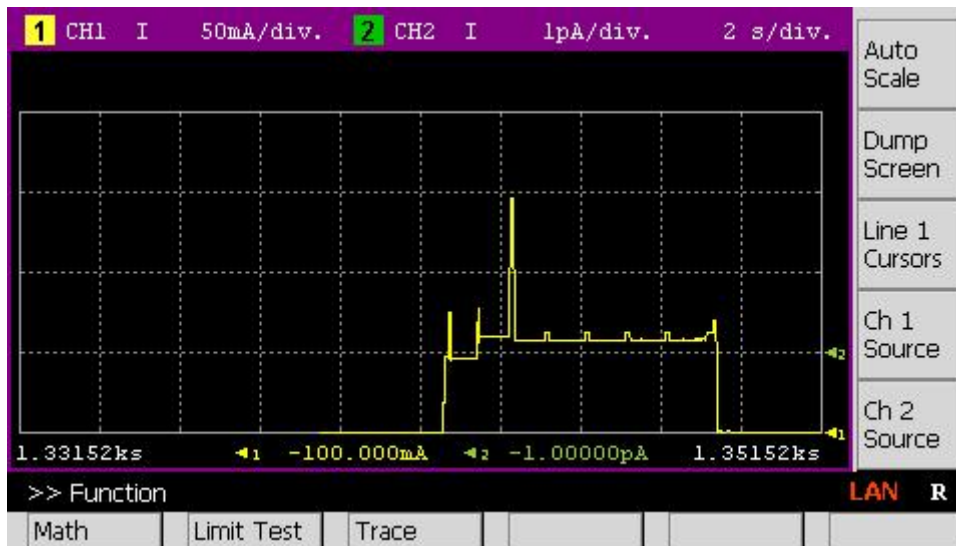


Figura 10.9.: Consum de l'ESP32 durant una transmissió de dades SF=7

10.5.2. Gràfica de consum amb un SF diferent de 7

També són interessants les gràfiques de consum amb un SF diferent de 7 per veure com es pot veure reflectida una transmissió més llarga en el consum.

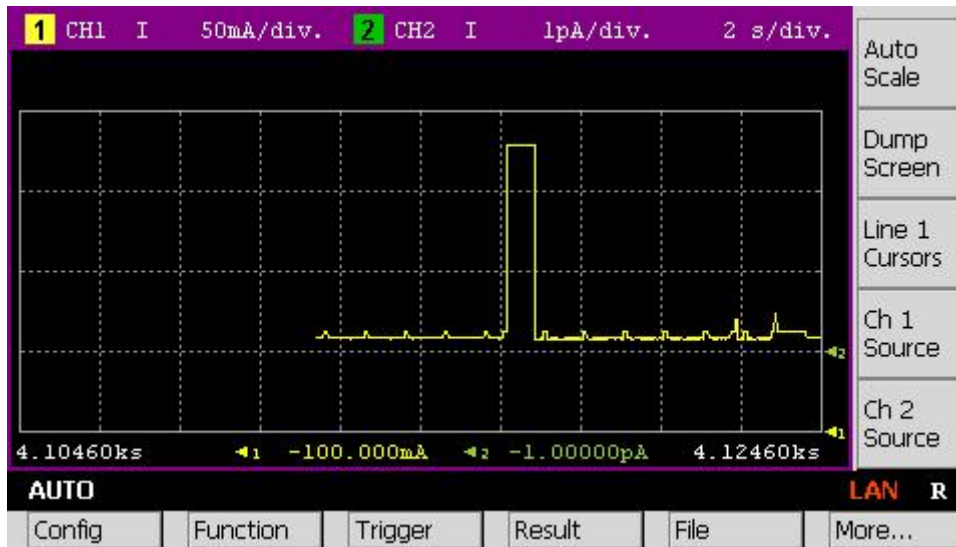


Figura 10.10.: Consum de l'ESP32 durant una transmissió de dades SF=10

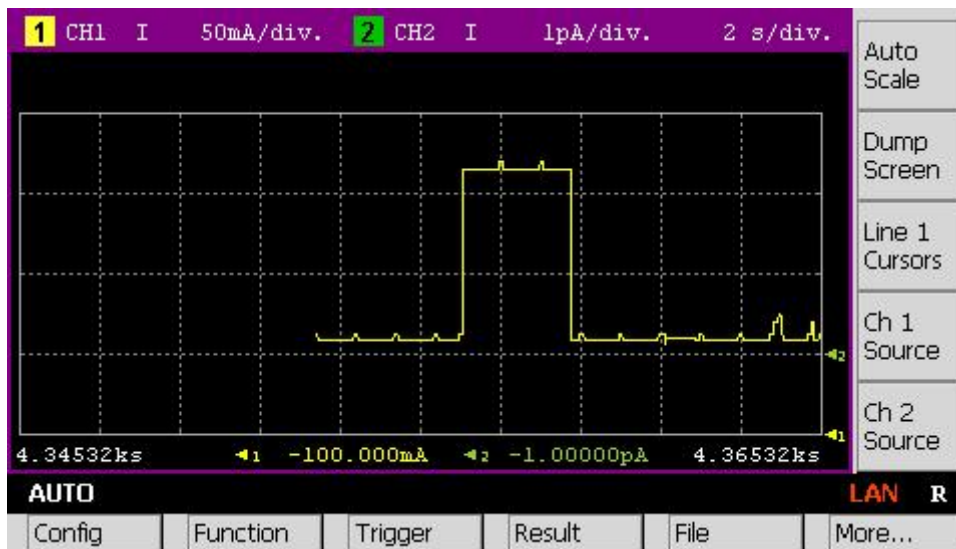


Figura 10.11.: Consum de l'ESP32 durant una transmissió de dades SF=12

Es pot veure com la falta de cobertura s'expressa en un consum més allargat en el temps mentre dura la transmissió de les dades. D'igual manera que el temps on el microcontrolador no està en mode sleep.

La raó d'aquests pics és la disminució del bitrate amb l'augment del SF. I això es tradueix en un temps de transmissió (i consum) per missatges de la mateixa llargada de bytes més llarg.

Des de la pantalla de monitoratge del Gateway o dispositiu de la consola TTN s'ha pogut contrastar la recepció del missatge amb els paràmetres LoRa de Spreading Factor corresponents als valors expressats en les mesures i les gràfiques.

Per altra banda, s'ha pogut veure com els temps de transmissió de les gràfiques de consum varien de forma semblant a les gràfiques dels temps de transmissió.

10.6. Instal·lació del Gateway a l'EPSEM

Una de les feines realitzades, també ha sigut la instal·lació del gateway a la xarxa de Gui-fi.net, que proporciona connexió a Internet, per a començar a rebre missatges de dispositius LoRaWAN pel seu enviament a la xarxa TTN.

11. Conclusions

En aquesta part del document es descriuran les conclusions del projecte. Pel que fa als objectius, aquests s'han complert:

11.1. Valoració general

S'ha realitzat una recerca i recollit informació sobre la comunicació LoRa i quins elements i solucions hi ha per a poder muntar un sistema de comunicació basat en aquesta modulació del senyal. També, s'han vist diferents solucions per a xarxes de baix consum que comparteixen característiques amb la modulació LoRa i que utilitzen aquests tipus de comunicacions punt a punt per a connectar dispositius a Internet (Les anomenades LPWAN).

D'igual manera, s'ha vist un tipus de xarxa (LoRaWAN), que està inclòs dins del conjunt de xarxes LPWAN, i aspectes com la seva arquitectura, estàndard, entitats, certificacions... Atrament, s'han vist solucions específiques per a confeccionar elements d'una xarxa LoRaWAN.

Posteriorment, s'ha estudiat la xarxa «The Things Network». És una xarxa que permet el registre de nous gateways per formar part de la xarxa i que permeten la connectivitat de dispositius a aquesta. S'han descrit les dues versions d'arquitectures de la xarxa (versió 3). També, s'ha explicat com configurar i connectar gateways i dispositius a la xarxa TTN.

A més a més, s'ha fet una tria de dispositius per a la comunicació de dispositius a través de la xarxa TTN i per fer experiments amb la capa física LoRa.

Finalment, s'han configurat i descrit les eines per a treballar amb els components triats per a poder fer les proves també realitzades com a exemple. Com també disposar de les dades proporcionades pels dispositius a una DDBB i permetre el registre de nous dispositius.

La finalitat d'aquest projecte ha sigut posar a disposició les eines i els coneixements, fruit de la recerca, per a futurs sistemes, experiments, projectes que necessitin o s'ajustin a les característiques per a utilitzar les tecnologies treballades en aquest projecte: LoRa, LoRaWAN, TTN...

11.2. Valoració personal

Personalment, he quedat satisfet amb el treball realitzat al projecte. He valorat de manera positiva tot el coneixement que he pogut obtenir i sintetitzat en aquest document per a utilitzar-lo com a base per a futurs treballs o aplicacions amb els elements estudiats.

Pel que fa la part pràctica del projecte, per una part s'ha configurat els elements necessaris per a la connexió a la xarxa «The Things Network» i poder utilitzar-la de forma pràctica per a la connexió de dispositius a serveis d'Internet. He trobat aquesta tasca motivadora degut a l'elasticitat que disposen els sistemes que utilitzen la xarxa TTN. M'ha sorprès l'alt grau d'escalabilitat que permet la xarxa i pot ser molt útil per a projectes futurs.

Respecte als experiments amb els dispositius que es comuniquen utilitzant la modulació LoRa, S'han pogut posar a prova alguns aspectes de la tecnologia de modulació LoRa.

Bibliografia

- [1] Quadre nacional d'atribució de freqüències
<https://www.boe.es/eli/es/o/2017/10/25/etu1033/dof/spa/pdf>
- [2] SEMTECH - AN1200.22 - LoRa Modulation Basis
<http://wiki.lahoud.fr/lib/exe/fetch.php?media=an1200.22.pdf>
- [3] T. T. Nguyen, H. H. Nguyen, R. Barton and P. Grossetete, "Efficient Design of Chirp Spread Spectrum Modulation for Low-Power Wide-Area Networks,in IEEE Internet of Things Journal, vol. 6, no. 6, pp. 9503-9515, Dec. 2019, doi: 10.1109/JIOT.2019.2929496.
- [4] A. Berni and W. Gregg, "On the Utility of Chirp Modulation for Digital Signaling,in IEEE Transactions on Communications, vol. 21, no. 6, pp. 748-751, June 1973, doi: 10.1109/T-COM.1973.1091721.
- [5] Kais Mekki, Eddy Bajic, Frederic Chaxel, Fernand Meyer, A comparative study of LPWAN technologies for large-scale IoT deployment, ICT Express, Volume 5, Issue 1, 2019, Pages 1-7, ISSN 2405-9595.
- [6] A. Lavric, A. I. Petrariu and V. Popa, L-Long Range SigFox Communication Protocol Scalability Analysis Under Large-Scale, High-Density Conditions,in IEEE Access, vol. 7, pp. 35816-35825, 2019, doi: 10.1109/ACCESS.2019.2903157.
- [7] G. L. Stuber, "Soft decision direct-sequence DPSK receivers,in IEEE Transactions on Vehicular Technology, vol. 37, no. 3, pp. 151-157, Aug. 1988, doi: 10.1109/25.16541.
- [8] Sigfox - device radio specifications
<https://build.sigfox.com/sigfox-device-radio-specifications>
- [9] Normativa - Unlicensed ISM bands European standard
https://www.etsi.org/deliver/etsi_en/300200_300299/30022002/03.02.00_20/en_30022002v030200a.pdf
- [10] LoRa Alliance Technical Committee, L-LoRaWAN Specification", LoRa-Alliance, October 2020.
https://lora-alliance.org/resource_hub/lorawan-104-specification-package/
- [11] LoRa Alliance Org., "What is LoRaWAN Specification".
<https://lora-alliance.org/about-lorawan/>
- [12] RP002-1.0.2, LoRa Alliance, L-LoRaWAN Regional Parameters", O.Seller, A.Yegin
https://lora-alliance.org/resource_hub/rp2-102-lorawan-regional-parameters/
- [13] Microchip Website, ATSAM R34 J18 Microcontroler, Software and documents section
<https://www.microchip.com/wwwproducts/en/ATSAMR34J18>

- [14] Microchip, SAM R34/R35 Microchip LoRaWAN Stack Software API Reference Manual
<https://ww1.microchip.com/downloads/en/DeviceDoc/SAM-R34-R35-Microchip-LoRaWAN-Stack-Software-API-Reference-Manual.pdf>
- [15] Microchip Website, SAM-R34 Xplained Pro Evaluation Kit
<https://www.microchip.com/DevelopmentTools/ProductDetails/dm320111>
- [16] Microchip Website, RN2483
<https://www.microchip.com/wwwproducts/en/RN2483>
- [17] Microchip, SAM L21 Family datasheet
https://ww1.microchip.com/downloads/en/DeviceDoc/SAM_L21_Family_DataSheet_DS60001477C.pdf
- [18] The Things Network Webpage
<https://www.thethingsnetwork.org/>
- [19] The Things Network Webpage, Network Architecture
<https://www.thethingsnetwork.org/docs/network/architecture.html>
- [20] The Things Network Webpage, Things Stack V3 Documentation
<https://www.thethingsindustries.com/docs/getting-started/>
- [21] Git Hub The Things Network, Semtech Packet Forwarder
https://github.com/TheThingsNetwork/packet_forwarder/tree/legacy
- [22] The Things Network Webpage, Console
<https://console.thethingsnetwork.org/>
- [23] The Things Network Webpage, RAK2245 Pi Hat Edition TTN documentation
<https://www.thethingsnetwork.org/docs/gateways/rak2245/index.html>
- [24] RAK Wireless Webpage, RAK833 Module
<https://www.rakwireless.com/en-us/products/lpwan-gateways-and-concentrators/rak833>
- [25] Semtech Corp Copyright, LoRa Basic Station of a LoRa packet forwarder.
<https://doc.sm.tc/station/>
- [26] Tektelic, KONA Macro IoT Gateway spec sheet
<https://tektelic.com/catalog/kona-macro-lorawan-gateway>
- [27] The Things Network Webpage, TTN Application documentation: Application Manager API
<https://www.thethingsnetwork.org/docs/applications/manager/index.html>
- [28] The Things Network Webpage, TTN Application documentation: Data API
<https://www.thethingsnetwork.org/docs/applications/mqtt/index.html>
- [29] The Things Network Webpage, TTN Application documentation: SDKs & Libraries
<https://www.thethingsnetwork.org/docs/applications/sdks.html>

-
- [30] Python Pip Webpage, Python TTN Library Project
<https://pypi.org/project/ttn/>
- [31] The Things Network Webpage, TTN Application documentation: Integrations
<https://www.thethingsnetwork.org/docs/applications/integrations.html>
- [32] RAK Wireless Webpage, RAK811 Breakout board Datasheet
<https://docs.rakwireless.com/Product-Categories/WisDuo/RAK811-Breakout-Board/Datasheet/>
- [33] The Things Network Webpage, TTN Devices documentation
<https://www.thethingsnetwork.org/docs/devices/registration.html>
- [34] The Things Industries Webpage, The Things Stack V3 Documentatation
<https://www.thethingsindustries.com/docs/getting-started/>
- [35] The Things Industries webpage, The Things Stack V3 Architecture
<https://www.thethingsindustries.com/docs/reference/components/>
- [36] Mikrotik Webpage, Mikrotik Gateway LoRa 868 MHz wAP LR8 kit
https://mikrotik.com/product/wap_lr8_kit
- [37] Heltec Automation Documentation Webpage, Heltec ESP32 + LoRa Node Documentation
<https://heltec-automation-docs.readthedocs.io/en/latest/esp32/index.html>
- [38] The Things Network Webpage, The Things Node documentation
<https://www.thethingsnetwork.org/docs/devices/node/quick-start.html>
- [39] GitHub Repository, Heltec WiFi_Kit_series Arduino Develop Enviroment
https://github.com/Heltec-Aaron-Lee/WiFi_Kit_series
- [40] Heltec Automation Documentation Webpage, Heltec ESP32 + LoRa Node Documentati-on, LoRaWAN
<https://heltec-automation-docs.readthedocs.io/en/latest/esp32/lorawan/index.html>
- [41] GitHub Repository, Heltec ESP32 LoRaWAN
https://github.com/HelTecAutomation/ESP32_LoRaWAN
- [42] GitHub Repository, Heltec_ESP32 Library Repository
https://github.com/HelTecAutomation/Heltec_ESP32
- [43] GitHub Repository, Heltec_ESP32 Library Repository, LoRa Module's API
https://github.com/HelTecAutomation/Heltec_ESP32/blob/master/src/lora/API.md
- [44] Espressif Documentations Webside, ESP-IDF Project Documentation
<https://docs.espressif.com/projects/esp-idf/en/latest/esp32/>
- [45] GitHub Repository, LoRa-net, LoRa-Mac node Repository
<https://github.com/Lora-net/LoRaMac-node>

- [46] GitHub.io (Webpages), LoRa-net (LoRa Alliance & Semtech) documentation webpage, LoRa-Mac node Repository Documentation
<https://stackforce.github.io/LoRaMac-doc/LoRaMac-doc-v4.5.1/index.html>
- [47] The Things Network, The Things Network V3 Console (Europe 1)
<https://eu1.cloud.thethings.network/console/>
- [48] The Things Network, The Things Stack V3 Documentation, Microtik Gateway instalation
<https://www.thethingsindustries.com/docs/gateways/mikrotikrouterboard/>
- [49] MQTT Website, MQTT Website Home Page
<https://mqtt.org/>
- [50] The Things Stack Documentation, MQTT Server (MQTT Broker) documentation
<https://www.thethingsindustries.com/docs/integrations/mqtt/>
- [51] Python Package Index Webpage, Pyhton paho-mqtt Package
<https://pypi.org/project/paho-mqtt/>
- [52] Mosquitto Webpage, libmosquitto C documentation
<https://mosquitto.org/man/libmosquitto-3.html>
- [53] GitHub Repository, tfg LoRa Enric Garcia
<https://github.com/XicoPi/tfg-LoRa>
- [54] Microtic Documentations, Microtik RouterOS Documentation
<https://help.mikrotik.com/docs/display/ROS/General+Properties>
- [55] Adafruit, SX1276/77/78/79 - Datasheet
https://cdn-shop.adafruit.com/product-files/3179/sx1276_77_78_79.pdf
- [56] LoRatools Website, Air time calculator
<https://loratools.nl/#/airtime>
- [57] Navarrete, D.; Quispe-Coica, F.A. Low-Power Wide-Area, LoRa/LoRaWAN, para la gestión del agua en el entorno rural. 2020.
- [58] E. Raghuvvera, P. Kanakaraja, K. H. Kishore, C. Tanvi Sriya, D. P. B and B. Sai Krishna Teja Lalith, "An IoT Enabled Air Quality Monitoring System Using LoRa and LPWAN,"2021 5th International Conference on Computing Methodologies and Communication (ICCMC), 2021, pp. 453-459, doi: 10.1109/ICCMC51019.2021.9418440.

Part II.
Apèndix

A. The Things Network V2

Actualment, estan coexistent dues versions del servidor que controla la xarxa, que rep el nom de «The Things Stack» (V2 i V3). La versió 2 del servidor acabarà desapareixent (desembre de 2021), és per això que es recomana utilitzar la versió 3.

Prèviament, s'ha treballat amb la versió 2 del servidor fins que s'ha posat en funcionament la nova versió (V3).

A.1. The Things Industry

És la companyia propietària de la xarxa i que s'encarrega de gestionar-la amb el control del seu servidor principal.

A.2. Arquitectura de la xarxa

En aquesta secció es descriurà l'arquitectura de la xarxa de «The Things Network». Es parla del sistema intern per a poder manejar els dispositius i aplicacions que es poden agregar per formar part de la xarxa. [19]

A.2.1. Visió Externa

Està clar que aquesta part fa referència a com es veu el conjunt de la xarxa de manera externa sense tenir en compte el seu funcionament intern (backend). Els elements principals són els següents:

- Dispositius: són els elements que s'encarreguen de fer una feina específica i que utilitzen la xarxa per a aconseguir la seva connexió amb elements externs.
- Gateways: són dispositius que crea el pont entre els dispositius i el servidor. Aquest dispositiu tindrà la capacitat de comunicar-se amb els dispositius utilitzant una comunicació amb LoRa i amb el servidor pel Protocol d'Internet.
- Servidor: En termes generals és la part encarregada de controlar la xarxa sent el seu element central. Entre altres coses, controla i organitza les aplicacions, accessos a dispositius i gestiona el flux de les dades.
- Aplicacions: És la part que es desenvolupa per a complir un o més objectius involucrant els dispositius a què té accés gràcies a la xarxa.

El servidor, per exemple també té com a funció proporcionar una interfície per a les aplicacions perquè aquestes no hagin d'implementar el protocol LoRaWAN per a establir una comunicació amb els dispositius.

A.2.2. Part interna o Backend

En aquesta part es parlarà de les diferents parts més internes de la xarxa. Concretament, es parlarà de l'arquitectura de la part central de la xarxa.

L'esquema presentat a la documentació de l'arquitectura de la xarxa és el següent:

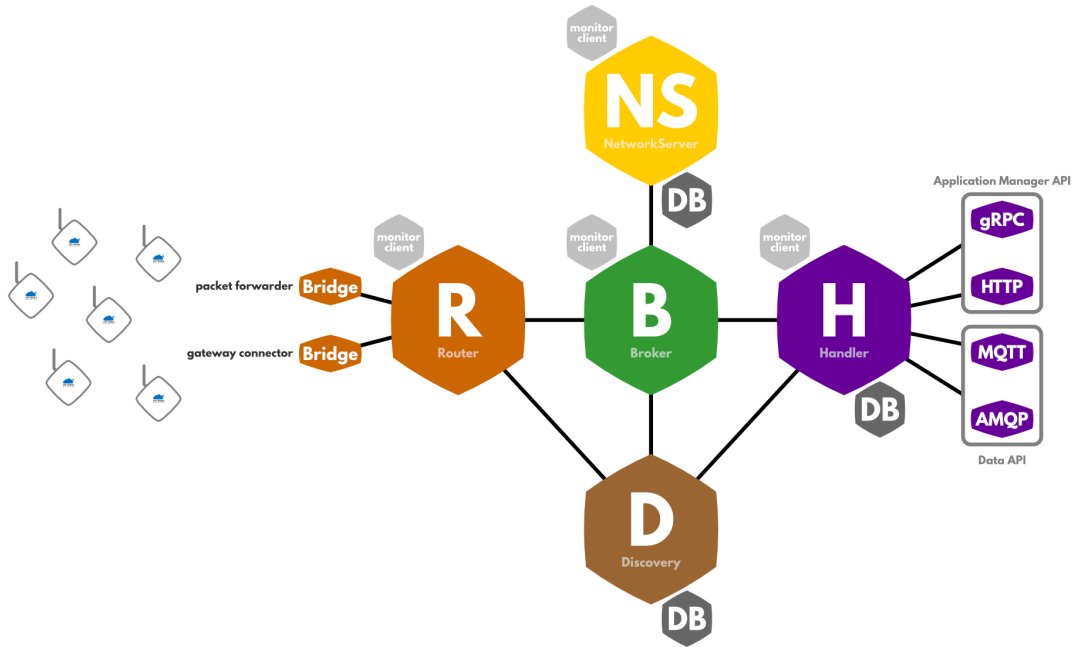


Figura A.1.: Backend architecture The Things Network V2

- Nodes: són els dispositius que transmetran els missatges utilitzant LoRa.
- Gateways: són els encarregats de rebre la retransmissió de dades dels diferents dispositius i enviar-les cap al backend o sistema.
- Router: és l'element on estarà connectat el gateway per a transmetre les dades. Cada router estarà connectat a un o més brokers. S'encarrega de transmetre les dades de cada gateway cap al broker que correspongui segons l'organització del sistema.
- Broker: Són la part central de la xarxa de «The Things Network». S'encarrega de mapejar cada dispositiu amb les seves aplicacions. D'aquesta manera poder gestionar els missatges d'interacció entre aplicacions i dispositius.
- Network Server: és la part encarregada de gestionar les parts específiques del protocol LoRaWAN.
- Handler: s'encarrega de gestionar les dades d'una o més aplicacions també és on es pot controlar o registrar aplicacions o dispositius.

La comunicació entre el handler i les aplicacions es realitza utilitzant un conjunt d'APIs.

A.2.3. Funcionament del nucli

Com ja s'ha dit la xarxa «The Things Network» és una xarxa que té un punt central (servidor) que està mantingut per una entitat concreta. Aquest servidor central rep el nom de «Things Network Stack» que actualment s'utilitza la versió 3 [20]. Aquesta versió està disponible per a utilitzar de forma gratuïta i es descriu a la seva pròpia documentació com a «build on an open source core».

Les funcions principals del nucli del sistema central de la xarxa són les següents:

- Funcions d'organització i distribució dels Gateways per al seu ús i compliment de normativa d'emissions electromagnètiques de cada regió.
- Funcions de control i organització dels dispositius o nodes. S'ha de tenir un control per l'accés als dispositius de les aplicacions i també amb la seguretat de les comunicacions.
- Són necessàries funcions que permetin la comunicació de les aplicacions amb la xarxa i puguin accedir a les dades.
- És necessària també adaptar-se al tipus de servei que pot oferir cada part distribuïda de la xarxa per a obtenir la màxima optimització de flux de dades, per exemple. És important poder estar descobrint noves aplicacions per a anticipar-se a l'aparició de nous serveis a la xarxa.

A.3. Gateways

Com ja s'ha dit, els Gateways són dispositius que formen part de la xarxa que es comunicaran amb els «devices» a través de la modulació LoRa. Agafaran aquests missatges i els enviaran a un «router» de la xarxa de «The Things Network». Perquè es pugui proporcionar la informació del sensor a les aplicacions corresponents.

«The Things Network» proporciona una interfície web per a poder controlar els diferents dispositius amb el teu compte d'usuari. Aquest sistema de control per als usuaris rep el nom de «The Things Network Console». [22]

A.3.1. Registre d'un Gateway

Configuració amb la xarxa

Per al registre i gestió de Dispositius, Aplicacions i Gateways s'ha de tenir un compte en «The Things Network». Accedint al «Console» es té accés a les aplicacions o Gateways vinculats amb el compte d'usuari que està autenticat en el moment.

- legacy packet forwarder: s'ha de marcar l'opció si el gateway l'utilitza. [21]
- Gateway ID: s'ha de triar una id per al gateway en el cas que aquest no utilitzi l'opció anterior (Legacy Packet forwarder) ha de ser única per a cada gateway nou de la xarxa.

- Gateway EUI: s'ha de posar en el cas que s'utilitzi l'opció «Legacy packet forwarder». Consta de 8 Bytes que identifiquen el node LoRa, solen apareixen inscrits en alguna part del mòdul o en el propi firmware.
- Descripció: s'ha d'afegir una descripció pel gateway,
- Frequency Plan: fa referència a la banda de freqüències que utilitzarà el gateway, que depèn de la normativa de la regió. Per a Europa només hi ha disponible la banda de 868 MHz la de 433 MHz no apareix com a opció.
- Router: s'ha de seleccionar un router on el gateway es connectarà.
- Regió: seleccionar la regió on estarà situat el Gateway.
- Situació de l'antena: finalment, triar si l'opció que correspongui a l'antena del gateways i se situa en l'exterior o en l'interior.

Configuració interna del Gateway

Depèn de cada gateway, s'ha de revisar les especificacions tècniques de cada dispositiu per a veure com es configura.

Essencialment, s'ha de facilitar la connexió a Internet del dispositiu i proporcionar-lo dels elements externs corresponents com l'antena en el cas que no la tingui incorporada.

A.3.2. Exemples de Gateways

En la documentació de la xarxa hi ha una sèrie d'exemples o opcions de gateways que són compatibles amb la xarxa de «The Things Network». El preu dels mateixos gateways pot dependre de certes característiques. La característica que més influència té en el preu del producte serà si el gateway està preparat per estar en exteriors o per només interiors.

Els Gateways que poden anar en exteriors són més cars que els d'interior perquè els fabricants han d'assegurar i certificar que no tindran cap problema perquè els dispositius puguin suportar inclemències climàtiques.

També hi ha certes solucions on l'electrònica no ve empaquetats en una carcassa i solen estar dissenyats perquè siguin un mòdul o complement extern a un sistema de computació estàndard o conegut com una Raspberry Pi. Aquests et proporcionen l'elasticitat per a poder experimentar i trobar solucions més específiques.

RAK2245 Pi Hat Edition

Aquest és el cas d'un producte que es ven com a component complementari a una Raspberry Pi perquè compleixi la funció de gateway LoRaWAN [23]. És un producte que consta d'una shield que es connecta a la Raspberry Pi que incorpora el mòdul LoRaWAN RAK833 [24].

Aquest producte té l'avantatge de poder ser configurat fàcilment des de la Raspberry Pi. En la mateixa web de TTN es pot trobar una guia per a configurar la Raspberry Pi perquè compleixi la funció de Gateway LoRaWAN.

Aspectes a tenir en compte:

- Tx Power: fins a 27 dBm
- Rx Sensitivity: des de -139 dBm @ (SF12 125KHz BW)
- Elasticitat i versatilitat: an ser un component complementari extern d'una Raspberry Pi, aquesta, pot complir algunes altres funcions diferents de la del Gateway.
- Soporta 8 canals de «uplink» i un de «downlink».
- Alimentació a 5V.
- Facilitat: la configuració del dispositiu és senzilla.
- Preu: el preu és relativament baix comparat amb altres alternatives.
- No està preparat per a estar a un lloc visible o exterior.
- No ve amb carcassa.

També hi ha packs que inclouen diversos complements al mòdul RAK2245 com una carcassa metàl·lica, antenes, i una sèrie de perifèrics com receptors GPS o per connexió a Internet a través de Xarxes de Mòbil. Un exemple seria «RAK7243C Pilot Gateway» que també apareix a la documentació de «The Things Network».

The Things Outdoor Gateway

Aquest és un exemple de Gateway preparat per anar instal·lat en exteriors. És un producte que està desenvolupat per l'empresa «The Things Industries», que és que controla i supervisa la xarxa de «The Things Network». Té un preu baix comparat amb altres gateways per exteriors amb un preu aproximat de 390€ sense impostos.

Algunes de les característiques són les següents:

- Es pot instal·lar en exteriors.
- Té un mòdem per a 3G i 4G (xarxa Mòbil).
- És resistent a l'aigua.
- S'alimenta pel connector Ethernet (PoE).
- Antenes per LTE, GPS i LoRaWAN.
- Es configura mitjançant una interfície web.

The Things Indoor Gateway

És el gateway per interiors desenvolupat per la companyia «The Things Industries» que té un preu aproximat de 73€ sense impostos.

Algunes de les característiques són les següents:

- 8 canals per LoRaWAN.
- Utilitza l'arquitectura «Basic Station» [25].
- Connectivitat per WiFi.
- Alimentació per cable USB type C.
- Antena omnidireccional incorporada.

També hi ha alternatives més cares que ofereixen també sistemes i eines de control del Gateway. També permeten més dispositius connectats alhora. Un exemple podria ser el KONA Macro IoT Gateway [26].

A.4. Aplicacions

Les aplicacions són elements externs al servidor que accedeixen a la xarxa de «The Things Network» per a poder obtenir les dades de diferents elements d'aquesta i interactuar amb ells a través d'una sèrie d'interfícies o APIs.

La comunicació entre la xarxa (servidor) i les aplicacions es realitza a través d'Internet. S'ha de registrar les aplicacions a la xarxa per a poder utilitzar-la d'igual manera que amb els Gateways o els dispositius. A la documentació de la xarxa ofereix una sèrie de APIs i llibreries/SDK per a diferents entorns de programació.

També, ofereix solucions integrades on s'ofereixen aplicacions ja integrades i amb possibilitat de configuració per a estalviar càrrega d'implementació.

A.4.1. Registre d'aplicació a la xarxa TTN

Perquè una aplicació tingui accés a la xarxa s'ha de registrar a través d'un compte d'usuari i la consola de la «Webpage» de TTN (The Things Network).

A l'apartat d'aplicacions a la consola TTN s'ha de triar l'opció d'afegir una nova aplicació i omplir els camps:

- Application ID: serà el nom de l'aplicació, ha de ser únic per a tota la xarxa.
- Description: una descripció per a l'aplicació.
- Application EUI: serà el número que identificarà l'aplicació dins la xarxa.
- Handler: s'ha de triar un handler, en el nostre cas es triarà el d'Europa «ttn-handler-eu».

Un cop registrada l'aplicació, dins de la configuració a la consola, ja es podran registrar dispositius proporcionant l'EUI del dispositiu i de l'aplicació. Cada aplicació pot tenir més d'un EUI d'aplicació. Un cop registrada l'aplicació es generaran les claus d'accés i l'App EUI.

A.4.2. Build an Application

A la documentació de la xarxa TTN, hi ha una explicació de com es pot implementar una aplicació perquè interaccioni amb els elements de la xarxa. Divideix les opcions en tres grups principals:

- APIs: s'accedeix a la xarxa a través d'una sèrie de mètodes per al control d'aplicacions i dispositius registrats a la xarxa TTN.
- SDK o llibreries: hi ha implementades diferents llibreries per al desenvolupament d'aplicacions per la xarxa i per diferents entorns de programació com Java, Go, Node-RED, Node.js i Python.
- Integracions: són solucions per a estalviar feina de programació en el desenvolupament de l'aplicació.

A.4.3. APIs

S'afirma a la documentació de TTN que és la manera més bàsica d'integrar una aplicació a la xarxa perquè s'accedeix de manera directa a funcionalitats del controlador d'aplicacions i dades de la xarxa TTN. Es divideixen en «Data API» i «Application Manager API».

Application Manager API

Aquesta API ofereix funcionalitats per a controlar aplicacions i dispositius que formin part de la xarxa TTN. L'API facilita la comunicació entre el Handler i l'aplicació. Està implementada en gRPC i HTTP [27].

Data API

Utilitza MQTT que és un protocol que proporciona connectivitat de tipus «machine-to-machine» per aplicacions d'Internet de les coses. En ser un protocol sencer a la documentació es mostren llibreries implementades per a diferents entorns de programació que implementen el protocol [28].

A.4.4. SDK o Llibreries

Aquesta eina permet el desenvolupament de l'aplicació en diferents plataformes de programació [29]. Aquestes eines estan implementades de manera interna utilitzant les APIs vistes en el punt anterior. Estan disponibles per als següents llenguatges de programació:

- Go.
- Java.
- Node.js.
- Python [30].
- Node-RED.

A.4.5. Integracions

Són les maneres més fàcils de connectar els dispositius amb les aplicacions. Consisteix a utilitzar una plataforma que manté l'aplicació i es comunica amb la xarxa per a controlar els dispositius [31].

La integració s'ha de controlar i configurar des de la plataforma que s'utilitzarà i des de la «The Things Network Console».

A.5. Dispositius

Cada dispositiu s'haurà de connectar a un gateway i ser registrat des de la consola TTN. Ha de suportar el protocol LoRaWAN per a poder formar part de la xarxa. Es pot utilitzar qualsevol dispositiu que estigui certificat pel protocol LoRaWAN encara que n'hi hagi alguna solució d'exemple en la web de TTN [33].

A.5.1. Exemples de dispositius

A la documentació de TTN es poden veure diverses opcions pel desenvolupament del dispositiu més elàstiques (mòduls o dispositius programables) que els sensors ja muntats o solucions més específiques que es poden trobar en les seccions més comercials de la Web [33].

The Things Uno

És un dispositiu que ve amb un mòdul LoRaWAN i que està pensat per a ser programat amb l'IDE d'Arduino juntament amb les llibreries que se'ns proporciona en la seva pròpia documentació. A la web de TTN explica com preparar l'entorn per a poder utilitzar-lo.

Nota: Malgrat el nom, el dispositiu està basat en un Arduino Leonardo i no en un Arduino Uno.

RAK811 LoRa Breakout Module

Consisteix en un mòdul, amb interfície de modulació LoRa amb el protocol LoRaWAN implementat en el firmware, que es comunica pel port sèrie a través de comandes AT. A més a més, el firmware del mòdul és «open source» [32].

El mòdul està ben documentat i permet una elasticitat significativa.

RAK811 LoRa Evaluation Board

És un dispositiu que funciona amb comandes AT, però que es comunica utilitzant una interfície USB. D'aquesta manera, es pot comunicar amb un ordinador per a interactuar amb el dispositiu.

LoPy

Aquest és un dispositiu que ve amb una interfície LoRa i que es programa amb microPython.