Trabajo de Fin de Grado

Grado en Ingeniería en Tecnologías de Industriales

Desarrollo de una red LoRaWAN® para IoT

Memoria

Autor: Pedro Andre Camarena Gamarra
Director: Antonio Calomarde
Convocatoria: Cuatrimestre de primavera 2022

ETSEIB

UPC

Resumen

Se investigará la implementación de una red LoRaWAN® para el IoT. En el futuro, se espera contar con varios tipos de redes que brinden una solución a los diferentes retos que se plantean dentro de la industria.

Una vez realizado el estudio e implementación del presente proyecto, se dispone de la información suficiente para afirmar que la red LoRaWAN® se ha desarrollado con éxito dentro del campus de la ETSEIB.

Abstract

The implementation of a LoRaWAN® network for the IoT will be investigated. In the future, it is expected to have several types of networks that provide a solution to the different challenges that arise within the industry.

Once the study and implementation of this project has been carried out, sufficient information is available to affirm that the LoRaWAN® network has been successfully developed within the ETSEIB campus.

Sumario

Resumen **1**

Sumario **3**

Glosario **5**

Prefacio **7**
- Origen del proyecto 7
- Motivación 7
- Requerimientos previos 7

Introducción **8**
- Objetivos del proyecto 8
- Alcance del Proyecto 8

Marco teórico **9**
- Introducción al IoT 9
- Análisis de la red IoT 11
- Estructura de red IoT 12
- Protocolos de comunicación IoT 13
- Porque utilizar LoRa® 14
- Protocolo de comunicación LoRa® 15
- Red de comunicación LoRaWAN® 18

Experimentación **23**
- Análisis de componentes 23
- Dispositivo STM32WL55 28
- Gateway Multitech Conduit IP67 32
- Conexión entre sistemas 34
- Comentarios 37

Conclusiones **39**

Agradecimientos **40**

Bibliografía **41**

1. Glosario

LoRaWAN® : Protocolo de comunicación y arquitectura de red.

LoRa® *: Técnica de modulación.*

IoT : Internet of the things

C : Lenguaje de programación.

Gateway : Puerta de enlace.

ED/end node : Dispositivo Final.

LPWAN : Redes de área grande y poder bajo.

AES: Estándar de encriptación avanzado.

SF: Factor de dispersión.

BW : Ancho de Banda.

CC : Capacidad del canal de comunicación.

TX : Transmitir.

RX : Recibir.

DC : *Ciclo de Trabajo.*

Uplink : Comunicación del dispositivo final hasta el gateway.

Downlink : Comunicación del gateway hasta el dispositivo final.

POE : Power On Ethernet. Alimentación por ethernet.

IP : Internet Protocol.

Red de datos : Redes de comunicación que permiten la comunicación de información entre diferentes dispositivos.

2. Prefacio

2.1. Origen del proyecto

Este proyecto comenzó debido a que existe la tendencia de interconectar cada vez más dispositivos entre sí. Debido a esto se han ido utilizando distintos tipos de redes para diversas aplicaciones y es dentro del IoT donde LoRa® posee relevante importancia.

2.2. Motivación

LoRa® es diferente a los protocolos de comunicación más utilizados. Además, existe una gran posibilidad de implementar la red LoRaWAN® dentro de la industria. Actualmente, existen equipos de trabajo dentro de empresas que se encargan de desarrollar proyectos IoT empleando esta red de comunicación.

2.3. Requerimientos previos

En la parte de investigación de este proyecto se abordaron los siguientes aspectos:

- Protocolos de telecomunicaciones.
- Microprocesadores.
- Protocolos de comunicación en ordenadores
- Programación de software embebido en c.
- Configuración básica de routers (servidores de red) que emplean Linux para ejecutar los sistemas desarrollados por los fabricantes.
- Cómo configurar sesiones de red en servidores LoRaWAN®.

3. Introducción

3.1. Objetivos del proyecto

El objetivo es establecer el protocolo de comunicación LoRa® entre dos dispositivos. Adicionalmente, poner en funcionamiento la red LoRaWAN® dentro de la ETSEIB.

3.2. Alcance del Proyecto

Incluye principalmente los siguientes temas:

- Protocolo de comunicación LoRa®.
- Servidores de red LoRa®.
- Servidores locales y en la nube conectados a Internet.

4. Marco teórico

4.1. Introducción al IoT

El Internet of Things o Internet de las cosas establece la conectividad entre distintos dispositivos (“cosas”) con el objetivo de intercambiar información a través de la red. Estos dispositivos a menudo son sistemas que procesan información y realizan distintas funciones como analizar, controlar, interactuar o medir. Estos dispositivos son vistos como objetos capaces de comunicarse mediante Internet u otra red con otros componentes.

En la actualidad, según *Findstack[3]* se proyecta un crecimiento exponencial de dispositivos conectados a la red. Además, dentro del informe VNI de Cisco (2021) se registró una cantidad de 345 millones de dispositivos conectados a Internet.

Todos los dispositivos electrónicos inteligentes que se implementan en proyectos IoT tienen en común, microcontroladores o microprocesadores. Estos elementos están compuestos por millones de transistores que forman estructuras lógicas digitales. El microprocesador se encarga de ejecutar tareas con una gran capacidad de cómputo, mientras que el microcontrolador coordina las tareas de un dispositivo. Estos elementos reciben y envían información mediante canales físicos de comunicación binaria (buses). Los buses envían la información hacia los diferentes elementos dentro de la estructura del microcontrolador.

Los microcontroladores están ideados para cumplir con todas las funciones de un dispositivo electrónico gracias a la estructura conformada por buses, registros de memoria, periféricos y unidades de control. En la Figura 4.1.1 se observan las conexiones de los sistemas de comunicación electrónica en las dos arquitecturas digitales más conocidas.

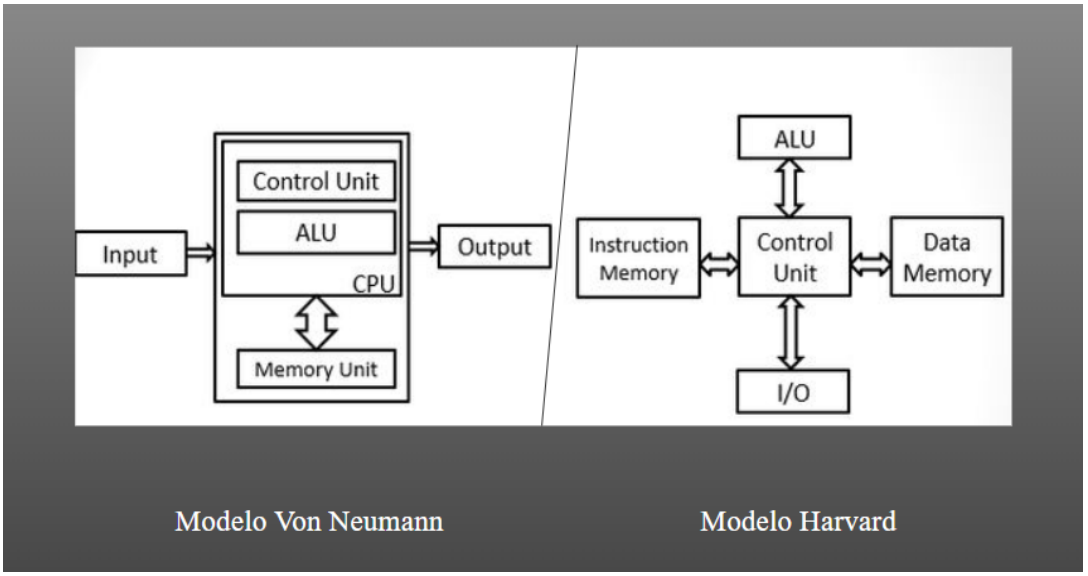


Figura 4.1.1. Arquitectura Von Neumann y Harvard dentro de un microcontrolador **[12]**.

En primer lugar, un protocolo de comunicación es un sistema de reglas que utiliza dos o más objetos para transmitir información mediante la variación de una magnitud física. Por ejemplo, el primer protocolo de comunicación que se inventó fue el código Morse. La aplicación más útil de estos protocolos de comunicación ocurre cuando se integra estos procesos a nivel electrónico (digital o analógico), ya que nos permiten comunicarnos de manera rápida con otros dispositivos. Esto es útil si se desea controlar sistemas mecánicos o eléctricos para fines industriales. Finalmente, podemos conectarnos dentro de una red mundial interconectada conocida como Internet para almacenar millones de datos de empresas o personas de manera descentralizada mediante el uso de servidores remotos (definición de nube).

Por otro lado, el Internet está coordinado por la ICANN *"Internet Corporation for Assigned Names and Numbers"* quien se encarga de administrar todos los nombres de los sitios web y asociarlos a una sola dirección IP. El Internet está implantado a nivel mundial gracias a una extensa red de cables de fibra óptica tendidos sobre el lecho marino o enterrados en tierra que interconectan todos los continentes del planeta.

A la hora de implementar una aplicación IoT, se tendrá que definir la red y esta debe especificar el conjunto de protocolos de comunicación a utilizar, en muchos casos ya existen protocolos de comunicación estandarizados para la red. Debido a que el conjunto de los protocolos de comunicación es grande, se comentará acerca de un modelo estandarizado que segmenta los distintos protocolos de comunicación que dan origen a la red. El modelo OSI *"Open Systems Interconnection"* es un modelo que se encarga de

segmentar la comunicación digital por capas (ISO/IEC 7498-1). Como se puede observar en la Figura 4.1.2. Las capas dentro de una red quedan agrupadas dependiendo de la funcionalidad que tengan.

En los siguientes apartados analizaremos este modelo aplicado al IoT.

Figura 4.1.2. Modelo OSI **[5]**.

4.2. Análisis de la red IoT

Según lo explicado anteriormente, los sistemas IoT se pueden dividir en diferentes capas y cada una explica las funcionalidades de los diferentes protocolos de comunicación. Dentro de estos protocolos, están los que establecen la transmisión de datos de manera inalámbrica mediante ondas electromagnéticas, se encuentran en la capa física (1).

Entre la capa física y la capa de red (3) se establece la comunicación relacionada con el medio físico. Mientras que desde la capa de transporte (4) en adelante, la comunicación se relaciona más con el ordenador (host) conectado. Cabe destacar que la capa de aplicación (7) puede funcionar como central de control de la información, ejecutando de manera automática comandos dirigidos hacia el resto de las demás capas.

Los protocolos de comunicación inalámbricos, en comparación a los cables de red como

el RJ45 (cable de ethernet), cuentan con una mayor latencia y con sistemas de seguridad y encriptación de datos, debido a que esta información se emite en el aire y puede ser captada por varios receptores desconocidos. Estos protocolos inalámbricos están diseñados para soportar distintos flujos de información (bit/s), y los más utilizados actualmente son el WiFi, Bluetooth, 3G, 4G, 5G, Zigbee, LoRaWAN®, Nb-iot, Z-bee, etc.

4.3. Estructura de red IoT

A la hora de definir una estructura de red dentro del IoT, se tiene que definir específicamente la aplicación que se le desea dar al sistema. Es por eso que existen diferentes tipos de arquitectura, las cuales tienen en común la función de integrar componentes. Dentro de estas arquitecturas se encuentran las que incorporan a la red sensores de equipos urbanos, conexión con la nube a través del Gateway IoT e integración de estos datos con Internet. La arquitectura de la Figura 4.3.1 es un ejemplo de arquitectura disruptiva dentro del IoT, ya que plantea la integración del IoT a la red pública urbana.

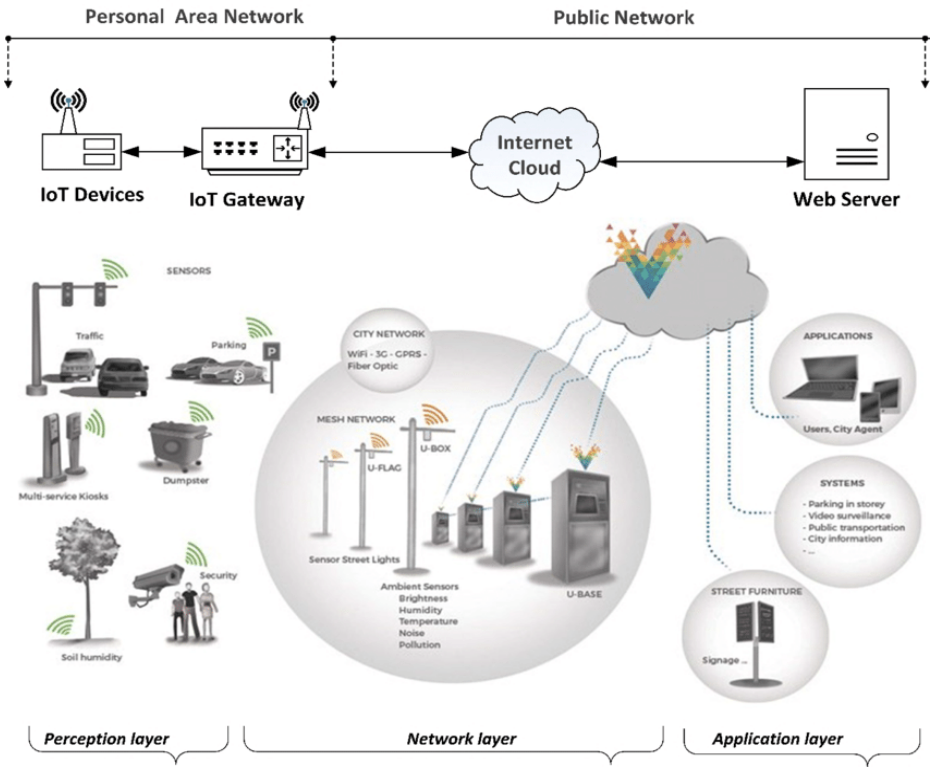


Figura 4.3.1. Ejemplo de arquitectura disruptiva del IoT **[13]**.

4.4. Protocolos de comunicación IoT

Los protocolos de comunicación disponen de diferentes tipologías. Algunas de las características más importantes a la hora de definir un protocolo de comunicación son las siguientes:

- Velocidad en la transmisión de datos.
- Rango de cobertura.
- Coste de implementación de la red.
- Latencia.
- Canales de comunicación

Podemos ver en la Figura 4.4.1. que se pueden identificar dependiendo de la distancia de comunicación, velocidad de transmisión de datos y coste de implementación.

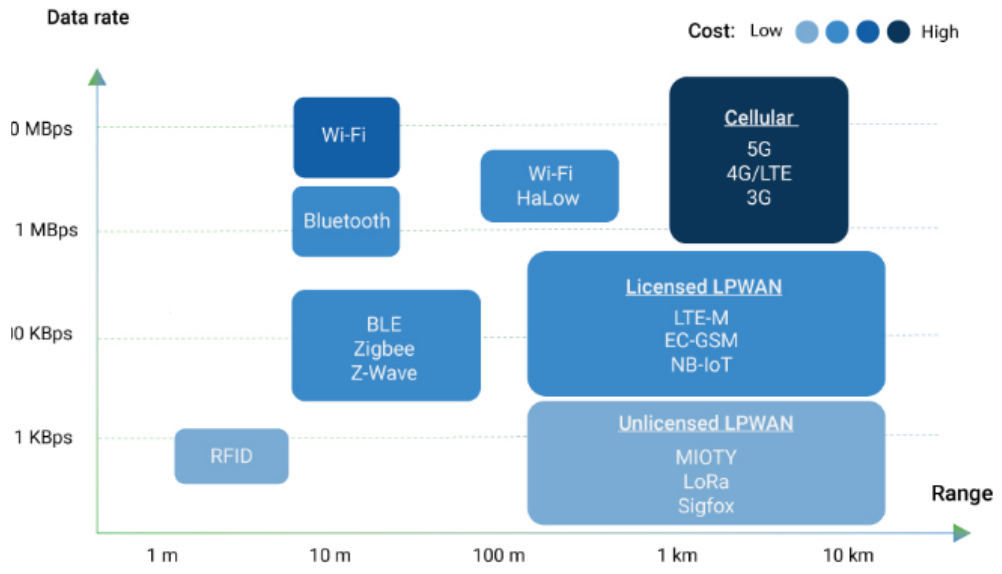


Fig. 4.4.1. Esquema de protocolos de comunicación **[14]**.

Dentro de esta clasificación, el nivel de eficiencia de cada protocolo depende de la madurez tecnológica en la que se encuentre el mismo. En cuanto a software, han ocurrido muchas actualizaciones desde la invención de cada uno de estos protocolos. Donde destacan las siguientes:

- El BLE (Bluetooth Low Energy) permite llegar a la eficiencia energética

- El WiFi que se encuentra regulado con los estándares del IEEE 802.11, está perfeccionando sus protocolos de envío de paquetes de información para ofrecer un mejor servicio.
- Las redes de telefonía móvil con el desarrollo del 5G tienen una velocidad de transmisión de datos mayor que sus predecesoras, el 4G y el 3G.

Sin embargo, todo este desarrollo está fijado a las características físicas de transmisión. LoRa® se caracteriza principalmente por ofrecer un amplio rango de comunicación, poco consumo de energía, estructura de red y coste de implementación económico en comparación a las antenas de telefonía móvil.

La red funciona de la siguiente manera, los dispositivos finales o *"end devices"* o *"end nodes"* utilizan LoRa® para establecer una conexión con el *"gateway"*. El *"gateway" o puerta de enlace* es un componente que establece los medios de comunicación de la red LoRaWAN® con los dispositivos finales. Además, a través de protocolos de comunicación de servidores MQTT se puede direccionar esta información a diferentes servidores (remotos o locales). Una representación esquemática de esta estructura de red se puede observar en la Figura 4.4.2. donde se muestra el esquema de red en una arquitectura LoRaWAN®.

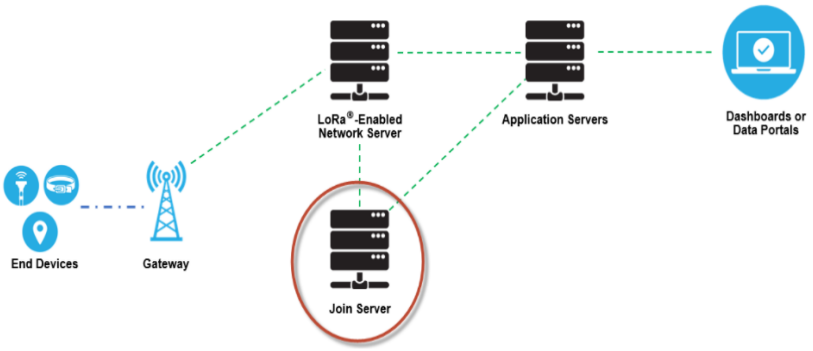


Fig. 4.4.2. Servidor de unión LoRaWAN® en una implementación de una red típica de LoRaWAN® **[2].**

4.5. Porque utilizar LoRa®

Dentro del IoT existe la necesidad técnica de administrar los cientos de dispositivos de la red de manera organizada y significativa. LoRa® es útil donde se planea implementar dispositivos a largo alcance y de bajo consumo.

Semtech creó este protocolo para satisfacer las necesidades de una red más grande que

cubra un espacio amplio para los usuarios finales. Esta misión también fue adoptada por parte de la Alianza LoRa® la cual tiene como propósito estandarizar los protocolos LPWAN (Low Power Wide Area Network) para el desarrollo de aplicaciones IoT. LoRaWAN® tiene como objetivo el convertirse en el tipo de red Sub-Ghz más importante de Europa y de otros países del mundo gracias a que ofrece a sus usuarios la posibilidad de comunicar información a través de largas distancias y con bajo coste.

Como muestra de su implicación dentro de la industria IoT, LoRaWAN® se implementa en los siguientes proyectos:

- Monitorización de datos eléctricos en parques fotovoltaicos desarrollados por la empresa Weidmüller IoT Hub.
- Monitorización de datos para sistemas HVAC con módulos LoRa® IoT desarrollados por la empresa Schneider Electric.

4.6. Protocolo de comunicación LoRa®

LoRa® es un sistema de comunicación que se deriva de las Redes de área grande y consumo bajo (LP WAN). Esta técnica de comunicación es parecida al sistema que usan los animales para comunicarse y se caracterizan por ser cíclicos, fijados en un determinado intervalo de valores, y de poseer cambios de frecuencia que determinan cómo se comunican los datos.

Las características principales de LoRa® vienen gracias a la técnica de modulación implementada. Esta señal se caracteriza por tener la posibilidad de ser demodulada correctamente por debajo del umbral de ruido. La señal que se emite contiene una variación lineal de la frecuencia de onda, la cual aumenta o disminuye su valor. El dispositivo recoge esta información y utilizando la técnica *"Chirp spread spectrum" (CSS)* demodula la señal.

LoRa® implementa la variación de frecuencia, desde ahora conocida como *chirp,* como la base para demodular cada símbolo de bits. Esta señal se puede dividir en s imágenes desde 0 hasta $2^{SF} - 1$ posibles imágenes donde SF es el factor de dispersión. A cada una de estas s imagenes le corresponde una señal binaria c que se puede observar en la Figura 4.6.1.. Donde Ts es el período de muestreo de la variación de la frecuencia, ***s*** es el número binario de la imagen que le corresponde a la señal para cada caso n y k es el incremento natural en cada instante. El implemento de la función mod lleva la señal al valor ínfimo cuando se alcanza el límite del ancho de banda BW. Esta discontinuidad ocasionada por la función mod permite las variaciones de la señal chirp y es la que

genera las diferentes señales dentro del conjunto s.

$$c(nT_s + kT) = \frac{1}{\sqrt{2^{SF}}} e^{j2\pi\left((s(nT_s))+k) \mod 2^{SF}\right)\frac{k}{2^{SF}}}$$

Figura 4.6.1. Ecuación de la señal transmitida en LoRa® **[23].**

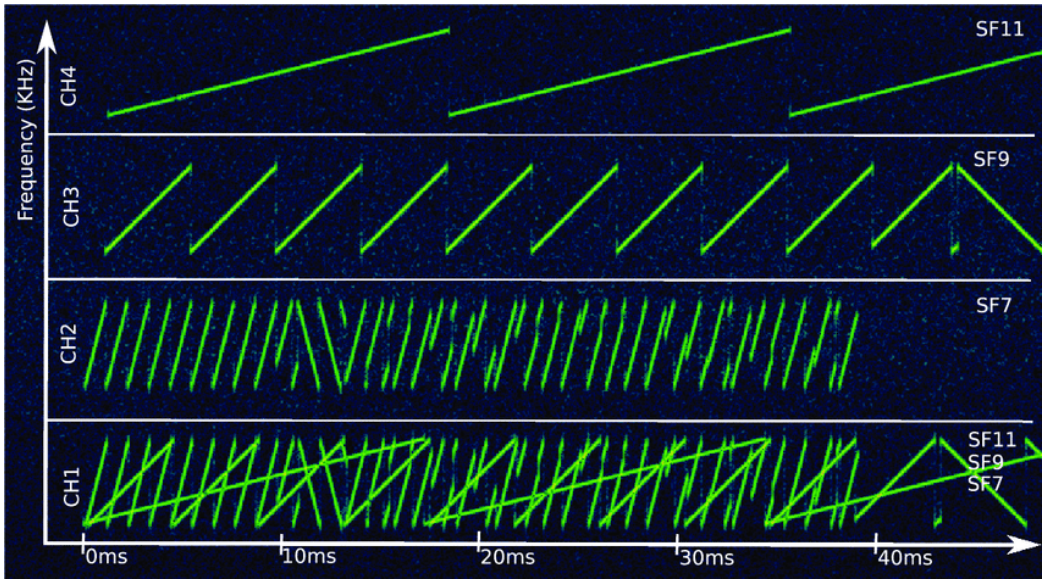


Figura 4.6.2. Transmisión real de una señal LoRa® **[25].**

El *chirp* es emitido y debido a factores externos e internos como lo son la atenuación, distorsión, ruido o alineación, el receptor ha de aplicar la técnica de CSS después de identificar el paquete de información que se transmite. En primer lugar, LoRa® utiliza una técnica de identificación que consiste en enviar el mismo símbolo repetidas veces [Figura 4.6.2]. Esta señal es el chirp base y sirve para corregir la alineación de los datos, ya que al contar con un mismo tipo de frame en cada comunicación puede autoajustarse para reconocer los frames posteriores.

Una vez la señal llega al receptor ocurre una técnica de demodulación conocida como la "transformada de Fourier rápida". Esta consiste en multiplicar la señal por la señal base "*dechirping*" y después por cada una de las ***s*** para encontrar la correlación máxima, es decir, encuentra el valor dentro del grupo de ***s*** que más se parezca la señal. Finalmente,

se obtiene el valor binario de la señal.

Como mencionamos al inicio LoRa® puede demodular las señales por debajo del ruido de fondo. Este fenómeno se debe a que el factor de correlación puede demodular la señal aunque contenga un ruido (dB) elevado.

A nivel funcional, LoRa® cuenta con una frecuencia de banda de 868 MHz para Europa (entre 863-870 MHz), esta frecuencia de banda varía dependiendo del continente donde se encuentre implementado el proyecto. Además, posee un ancho de banda BW (rango de frecuencias dentro de una banda) que varía en función de la del factor de dispersión SF *("spreading factor"*) y se encuentra entre 125 KHz o 500 KHz. El factor de dispersión es un valor adaptativo que se fija dependiendo de la distancia en la que se encuentre el gateway y el dispositivo final.

A la hora de establecerse la comunicación hay una relación entre el flujo de información máximo en bits/s o bps llamado capacidad del canal CC. LoRa® modifica la CC debido al factor de dispersión SF y el ancho de banda BW. Estos cambios también tienen efecto sobre la tasa de bits y el rango de la comunicación [Figura 4.6.3.].

Modulation	Spreading factor	Bandwidth [kHz]	Maximum application throughput per channel [bps]	Maximum application layer throughput per end device per channel [bps]		
				10% duty cycle[1]	1% duty cycle[2]	0.1% duty cycle[3]
LoRa	12	125	146.1	14.61	1.46	0.15
LoRa	11	125	261.4	26.14	2.61	0.26
LoRa	10	125	584.2	58.42	5.84	0.58
LoRa	9	125	1359.2	135.92	13.59	1.36
LoRa	8	125	2738.1	273.81	27.38	2.74
LoRa	7	125	4844.7	484.47	48.45	4.84
LoRa	7	250	9689.3	968.93	96.89	9.69
GFSK	-	150	45660.4	1851.6[4]	456.6	45.66

Figura 4.6.3. Modulación LoRa® **[15].**

En términos de prestaciones, al aumentar el SF se aumenta el tiempo que tarda en enviar un mensaje, esto hace que la potencia de transmisión aumente TX como consecuencia el sistema se comunica a menor velocidad y en un mayor rango. Por otro lado, disminuir el SF tiene como consecuencia: la disminución del rango, el aumento de la tasa de datos y la disminución de la potencia de transmisión TX [Figura 4.6.4.].

Spreading Factor (For UL at 125 KHz)	Bit Rate	Range (Depends on Terrain)	Time on Air for an 11-byte payload
SF10	980 bps	8 km	371 ms
SF9	1760 bps	6 km	185 ms
SF8	3125 bps	4 km	103 ms
SF7	5470 bps	2 km	61 ms

Figura 4.6.4. Tasa de datos adaptable LoRa® **[15].**

Las decisiones sobre la velocidad de datos dentro de un sistema LoRa® pueden ocurrir de manera automática si se programa. Esto tiene como consecuencia una mejora en la capacidad de la red.

En resumen, las prestaciones clave que permiten que LoRa® sea una red eficaz son las siguientes:

- Tasa de datos (bps) adaptable, ya que permite variar el rango de alcance dentro de una misma banda.
- Demodulación mediante CSS que permite mayor inmunidad al ruido.
- Establecimiento de canal en movimiento (resistente al “Efecto Doppler”), ya que se puede establecer el canal en movimiento sin perder información en la comunicación.
- Alto rango de comunicación, ya que al ser una red que opera en la banda de frecuencia de las Sub-Ghz posee un amplio radio de alcance.

4.7. Red de comunicación LoRaWAN®

LoRaWAN® es la red de dispositivos o nodos que utiliza el método de modulación LoRa®. Las características de LoRaWAN® son las siguientes:

- Bidireccionalidad, es decir, que tanto gateways como dispositivos finales se pueden enviar y recibir información . Se le conoce como Uplink a la comunicación desde el dispositivo final hacia el gateway y Downlink al revés.
- 8 Canales de comunicación en paralelo, pudiendo aumentar hasta 16 o 64 dependiendo de la tecnología del gateway.
- Limitaciones de uso de los canales durante el ciclo de trabajo (Duty Cicle) en Europa.

- Dispositivos de diferentes clases A, B y C.
- No puede soportar una carga útil de mensajes tan elevada.
- Limitaciones en el Downlink queda limitada porque actualmente el 90% o más de las comunicaciones utilizan Uplinks.
- Encriptación: Encriptación AES *"Estándar de encriptación avanzada"* que permite la encriptación de datos de punto a punto.
- Capacidad de red mejorada, ya que posee un sistema bidireccional de comunicación más eficiente por ser un sistema que "escucha".

LoRaWAN® define tres tipos de clases: A, B y C. La clase A [Figura 4.7.1] se caracteriza porque el mensaje siempre lo comienza el dispositivo final mediante el Uplink. Después de que la transmisión se culmine, el dispositivo abre dos ventanas de Downlink, si el dispositivo no recibe ninguna señal vuelve a repetir este proceso.

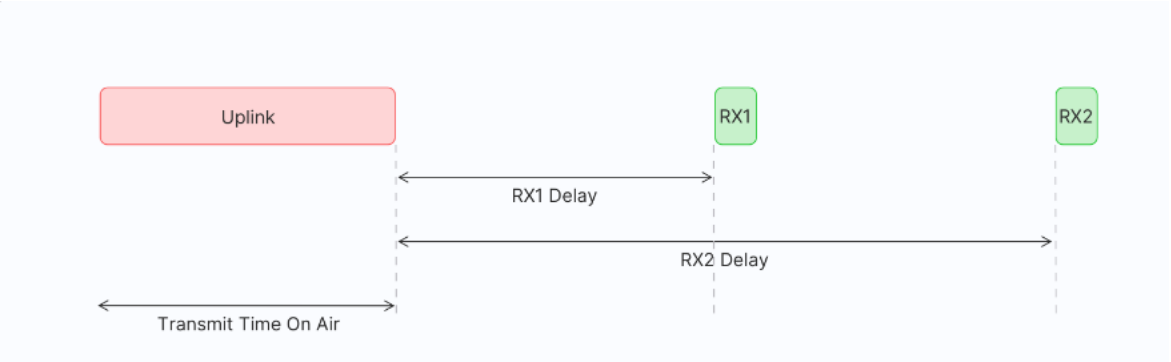


Figura 4.7.1 Establecimiento de comunicación de la clase A **[20].**

Una vez que se abren estás ventanas, el servidor LoRaWAN® puede responder dentro de estos dos intervalos RX1 y RX2. En caso favorable, se establece la comunicación y la recepción del mensaje. Esta clase se utiliza muy a menudo debido a que consumen muy poca energía porque los tiempos entre Uplinks sucesivos se pueden mantener bastante largos y puede permanecer en modo de ahorro de energía.

La clase B [Figura 4.7.2] usa ventanas de recepción adicionales a comparación de la clase A, la finalidad es sincronizar el envío de información. El período de sincronización se establece a través del *"beacon"* y después de sincronizar, el dispositivo envía un Downlink *"PIN"* con el objetivo de obtener el tiempo de sincronización del servidor LoRaWAN®. Finalmente, se establece la comunicación de manera sincronizada repitiendo el proceso de la clase A.

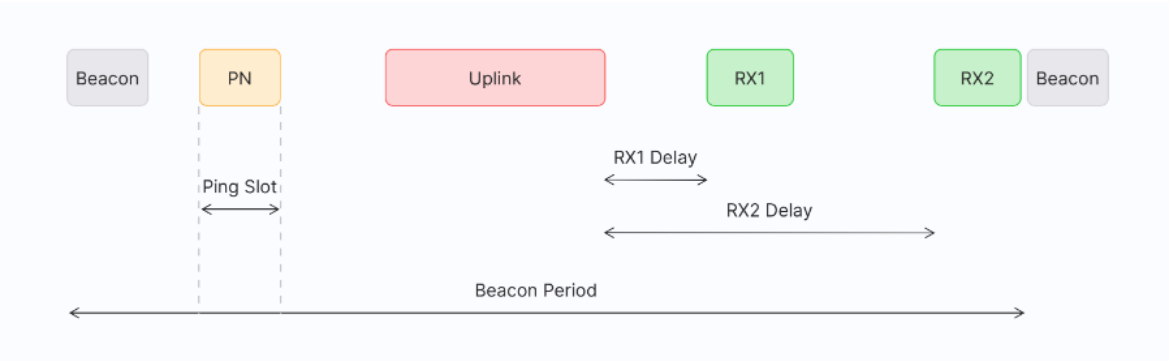


Figura 4.7.2 Establecimiento de comunicación de la clase B **[20].**

Por último, la clase C [Figura 4.7.3] se caracteriza principalmente por mantener una ventana de recepción continua, a menos que se encuentre en transmisión. Este factor la hace obtener una latencia menor a cambio de consumir mucha más energía.

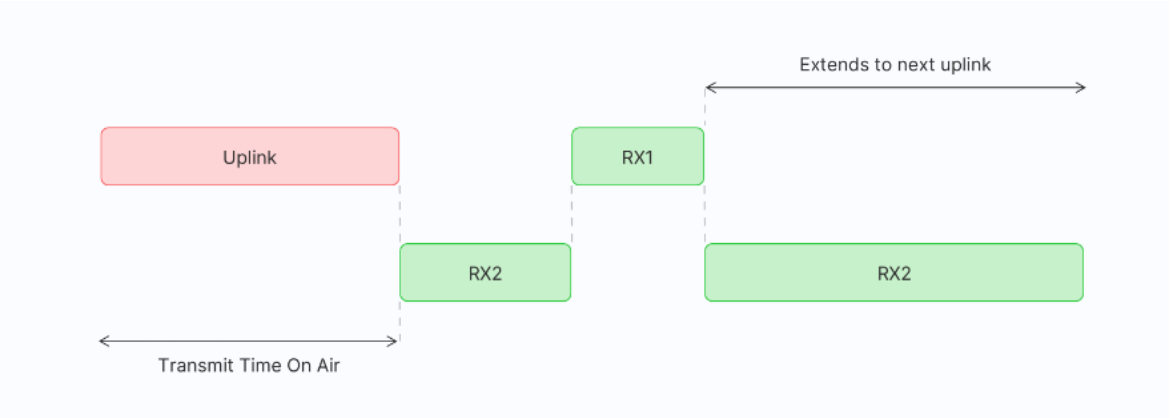


Figura 4.7.3 Establecimiento de comunicación de la clase C **[20].**

LoRaWAN® es viable a partir del uso de gateways que enlazan a los nodos finales. Los gateways están diseñados con tecnologías de red LoRaWAN® para que puedan establecer una comunicación segura con el dispositivo final [Figura 4.7.4.]. Cabe mencionar que la MAC (Medium Access Control) proporciona el control del flujo de información en el aire y es la encargada de establecer la capa de enlace de datos (2) en función de la clase LoRaWAN®.

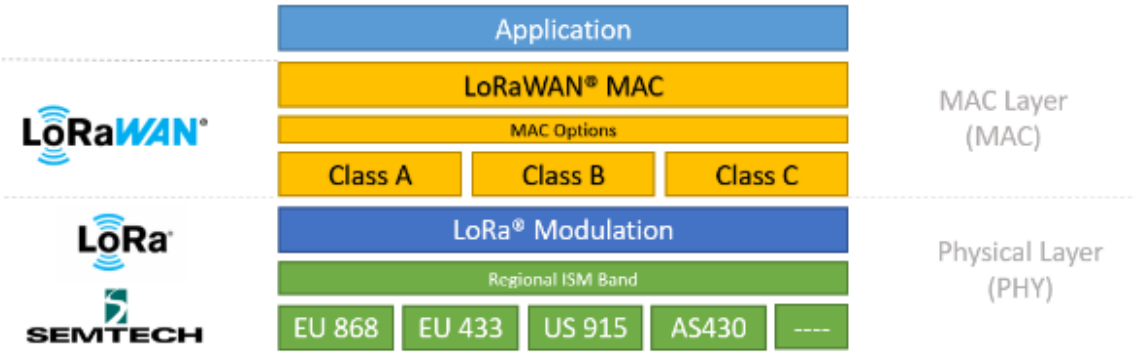


Figura 4.7.4 LoRaWAN® Capas de comunicación **[21].**

El gateway reenvía la información que emiten los dispositivos finales hacia el servidor LoRaWAN®. Este sistema tiene la posibilidad de ser implantado a gran escala porque el servidor LoRaWAN® puede conectarse a múltiples gateways que a su vez se conectan con múltiples dispositivos. Esta estructura ramificada [Figura 4.7.5] brinda la posibilidad de mejorar la conexión, ya que el dispositivo final puede ser atendido por más gateways lo cual reduce el error de comunicación y aumenta la capacidad de geolocalización de los dispositivos finales en caso de que ambos dispongan de estos módulos.

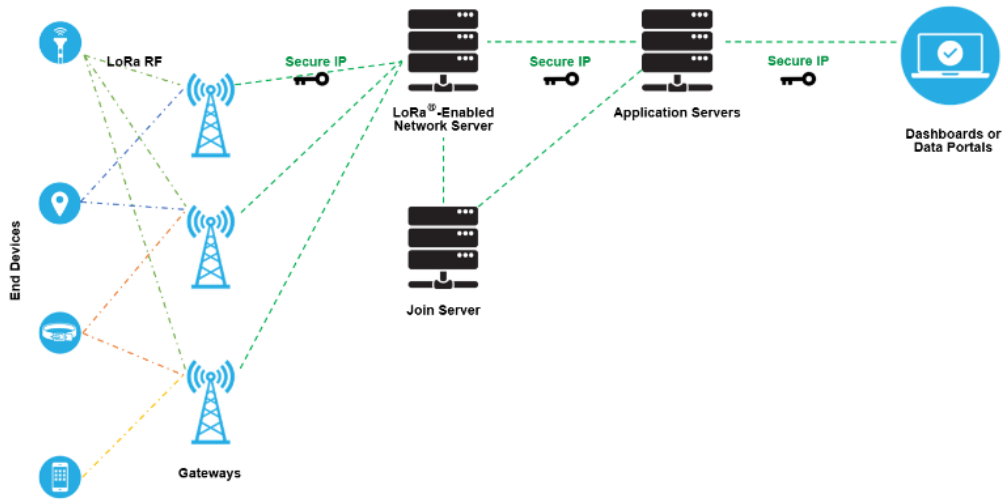


Figura 4.7.5 Implementación de red LoRaWAN® **[21].**

Una vez la comunicación sea redirigida hacia el servidor LoRaWAN®, para que pueda establecerse el intercambio de información, se necesitará del servidor de unión (Join

Server) y del servidor de la aplicación (Application Server) [Figura 4.7.6].

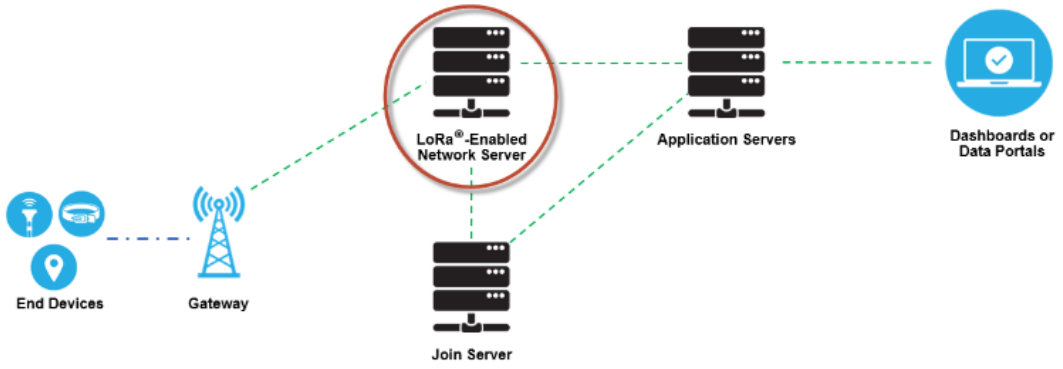


Figura 4.7.6 Servidor de red LoRaWAN® **[21].**

Por un lado, el servidor LoRaWAN® maneja y controla la red en cada proceso. Se encarga de establecer la encriptación simétrica de 128 bits de punto a punto, manejar la comunicación durante el proceso de unión, garantizar la comunicación de los paquetes de datos recibidos (identificarlos y cuantificarlos), modificar las variables de la modulación LoRa®, establecer la capa de enlace de datos (2) y responder a las consultas de uplink o downlink.

El servidor de unión (Join Server) es el encargado de establecer la comunicación entre gateway y dispositivo final. Contiene todas las direcciones necesarias (dirección EUI del gateway y del dispositivo) y las llaves (contraseñas) del servidor LoRaWAN® y del servidor de aplicación. Por último, el servidor de aplicaciones (App Server) se encarga de transformar la información de los datos enviados por los dispositivos para que sea analizada. Además, puede generar las peticiones de downlink para establecer peticiones hacia los dispositivos que se encuentren enlazados.

Los dispositivos nodo y gateway han de contar a nivel de Hardware con el módulo LoRa®. Estos módulos actualmente se pueden encontrar en una amplia gama de dispositivos porque las licencias LoRa® las han comprado diferentes fabricantes. No obstante, las prestaciones que pueden llegar a tener estos sistemas dependen de la calidad del dispositivo que se utilice y del código implementado a nivel de software.

En los proyectos LoRaWAN® es importante contar con un gateway de altas prestaciones debido a que está conectado con diferentes dispositivos finales y es el receptor central de la comunicación.

5. Experimentación

5.1. Análisis de componentes

En el mundo de los microcontroladores existen diversas estrategias de empresa que establecen los modelos de negocio dentro de este mercado. Para seleccionar el dispositivo adecuado nos centraremos en los diseñadores de los microprocesadores porque son los que permiten definir cuál es la capacidad del procesador a nivel interno. Dentro de este contexto se encuentra ARM, una empresa líder que vende licencias de microprocesadores. Esta empresa posee diferentes líneas de diseño como el ARM Cortex R, ARM Cortex-M y ARM Cortex-A. Estos diseños se utilizan en la mayoría de los dispositivos electrónicos que existen en la actualidad.

El ARM Cortex-R se utiliza en aplicaciones de tiempo real, por ejemplo en redes de datos (routers) o en coches. El ARM Cortex-M se utiliza en sistemas IoT o en dispositivos embebidos, ya que son sistemas de bajo costo y bajo consumo de energía. Por último, el ARM Cortex-A se utiliza en aplicaciones de alta capacidad de respuesta como en ordenadores de escritorio.

Para que el dispositivo final de nuestra red LoRa® funcione de manera efectiva dentro de una aplicación IoT será necesario utilizar un microprocesador dentro de la línea ARM Cortex M. Dentro de esta línea tenemos al ARM M4 que tiene buenas prestaciones a la hora de la comunicación inalámbrica y poder bajo de consumo. También se encuentra el ARM M0 + , estos dispositivos están destinados a necesidades muy limitadas con niveles de reloj mucho más bajos que el M4. Por tanto, el ARM Cortex M4 es el diseño ideal para el dispositivo final.

Se puede implementar el dispositivo final mediante Arduino, Raspberry Pi o STMicroelectronics. Cabe mencionar que la empresa creadora de LoRa® Semtech se encuentra actualmente asociada con STMicroelectronics razón por la cual los demás fabricantes se verían limitados a integrar LoRa®, mediante módulos de comunicación externos a la placa principal.

En la Tabla 5.1.1 se mencionan las ventajas y desventajas que conlleva usar los componentes de los principales fabricantes de hardware. Estos fabricantes poseen entornos de desarrollo integrado *"IDE"* donde se programan las tareas que ejecutará el

dispositivo.

	ARDUINO	RASPBERRY PI	STMicroelectronics
Ventajas	Tiene una amplia comunidad elaborando proyectos.	Posees las funciones de un ordenador portátil.	Posee software muy funcional.
Desventajas	Software que esconde procesos de configuración del dispositivo.	Configuración a través de sistema operativo.	Poca comunidad elaborando proyectos.

Tabla 5.1.1 Ventajas y desventajas de los principales fabricantes de hardware.

El software que posee STMicroelectronics contiene un entorno de desarrollo integrado *"IDE"* donde se pueden encontrar plantillas de trabajo con código genérico para el tipo de aplicaciones que se deseen implementar. Por ejemplo, si deseas implementar una red LoRa®, STM implementa la base de código que configura el funcionamiento de los relojes y registros internos del componente. En cambio, en Arduino y en Raspberry Pi el código es más explicativo y existe una comunidad colaborativa en diversas páginas web, como github.com o arduino.cc o projects.raspberrypi.org.

Finalmente, se decidió utilizar dispositivos de la marca STMicroelectronics porque comporta un beneficio de código genérico y se puede implementar las prestaciones de eficiencia para el microprocesador.

Dentro del conjunto de dispositivos IoT de la marca STM, podemos observar en la Figura 5.1.1. que tenemos las líneas de productos STM32WL y STM32WB para las comunicaciones inalámbricas.

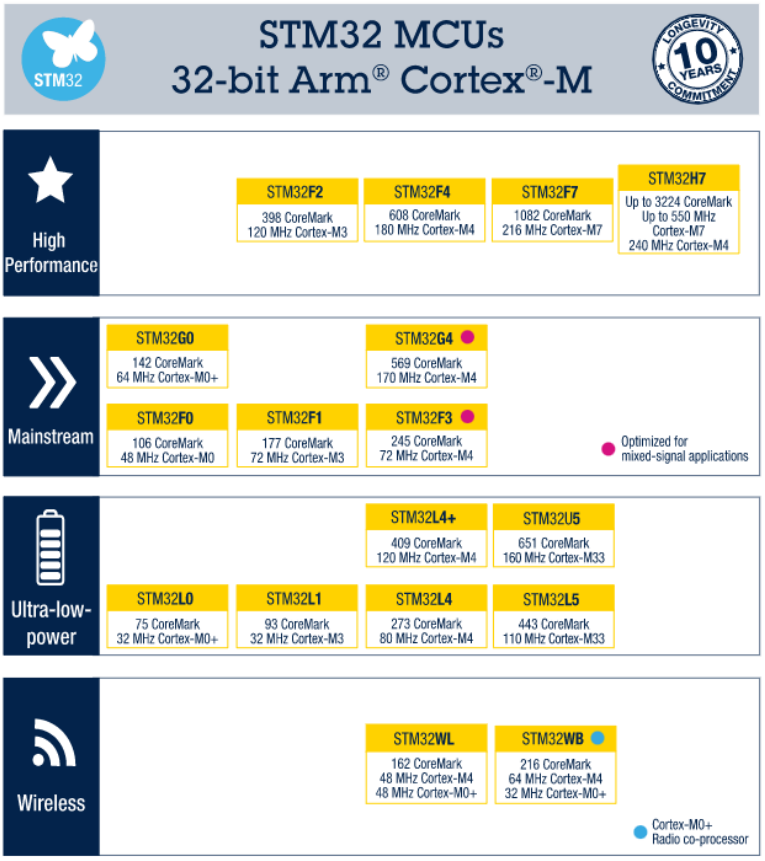


Figura 5.1.1 Componentes STM para aplicaciones IoT **[17]**.

La línea de productos STM32WL posee módulos de comunicación LoRa®, mientras que la línea STM32WB posee módulos de comunicaciones Bluetooth, Zigbee o Thread.

Dentro de la línea de producción del STM32WL se puede encontrar el STM32WL55 y el STM32WLE5 (Figura 5.2.2.). La diferencia está en el doble núcleo que incorpora el STM32WL55 (M4 y M0 +).

El departamento de electrónica de la ETSEIB dispone del producto STM32WL55. Por tanto, como cualquiera de estos dos componentes podría ser implementado como dispositivo final, se utilizó el que disponían en la facultad. Cada dispositivo se vende a 41,5 €, precio económico si se desea implementar estos dispositivos a larga distancia.

STM32WL MCU Series
32-bit Arm® Cortex®-M4/-M0+

RADIO
- Multi-modulation Sub-GHz radio
- 2 programmable power outputs
- Sensitivity down to -148 dBm

SECURITY
- AES 128/256-bit
- True Random Number Generator
- Private Key Accelerator
- PCROP / WRP
- 48-/96-bit unique IDs

CONNECTIVITY
- 2x SPI, 3x I2C, 1x ULPUART, 2x USART
- 16- and 32- bit timer

ANALOG
- LDO and built-in DC/DC
- 1x ADC 12-bit
- 1x DAC 12-bit
- Temperature sensor

OTHER
- 2x DMA (7 channels)
- 7x Timers (16 and 32 bits)
- 2x ULP Comparators
- ART Accelerator™
- Low voltage 1.8 to 3.6V

Product line	f_{CPU}	Flash Memory (Kbytes)	RAM (Kbytes)	Modulations: LoRa®	Modulations: (G)FSK	Modulations: (G)MSK	Modulations: BPSK	Radio Frequency Range (MHz)	Dual Power Output	Advanced Security Features	Operating Temperature Range
Cortex®-M4 single-core line											
STM32WLE5	Up to 48	Up to 256	Up to 64	•	•	•	•	150 to 960	1 output up to 22dBm 1 output up to 15dBm (consumption-optimized)		-40 to 85°C (with radio) -40 to 105°C (without radio)
STM32WLE4	Up to 48	Up to 256	Up to 64		•	•	•	150 to 960	1 output up to 22dBm 1 output up to 15dBm (consumption-optimized)		-40 to 85°C (with radio) -40 to 105°C (without radio)
Cortex®-M4 and -M0+ dual-core line											
STM32WL55	Up to 48	256	64	•	•	•	•	150 to 960	1 output up to 22dBm 1 output up to 15dBm (consumption-optimized)	Key Management Services Secure hardware isolation, secure boot, secure firmware update and secure firmware install	-40 to 85°C (with radio) -40 to 105°C (without radio)
STM32WL54	Up to 48	256	64		•	•	•	150 to 960	1 output up to 22dBm 1 output up to 15dBm (consumption-optimized)	Key Management Services Secure hardware isolation, secure boot, secure firmware update and secure firmware install	-40 to 85°C (with radio) -40 to 105°C (without radio)

STM32 Ecosystem

STM32Cube | Evaluation tools | Software tools | Embedded Software | Hardware tools | Security (STM32 Trust) | MadeFor STM32 | ST Partners (Partner Program)

STM32 Solutions

Connectivity

STM32 Learning / Communities

STM32 Community | STM32 Education | STM32 MCU Wiki | STM32 GitHub

Figura 5.1.2 Línea de productos STM32WL para aplicaciones IoT **[18]**.

Por otra parte, con el objetivo de implementar un gateway se necesitará un dispositivo inteligente capaz de configurar los servicios de la red. Diseñar un gateway es más

complejo que un nodo final porque al tener una estructura de servidor, ha de garantizar que la información sea correctamente enlazada, encriptada y distribuida. Por tanto, se ha de contar con Gateways fabricados por parte de empresas con aplicaciones desarrolladas para el funcionamiento adecuado de los servicios que pueden ofrecer estos dispositivos. Los principales gateways dentro del mercado son los siguientes:

- The things indoor gateway fabricado por The Things Network.
- RG1xx gateway fabricado por la empresa Laird.
- Multitech Conduit fabricado por la empresa Multitech.
- Wirnet iBTS gateway de la empresa Kerlink.
- Cisco LoRa® Gateway fabricado por la empresa Cisco.

Dentro de estos productos podemos encontrar sistemas que permiten crear una red LoRaWAN® de dos maneras, mediante un servidor en la nube o un servidor local. En este caso, si buscamos implementar el proyecto en la universidad, solo es seleccionable la opción de hardware que te permita generar una red local propia. Esto es debido a que al estar dentro de la UPC la red de la universidad podría verse vulnerada.

La puerta de enlace o gateway se ha de instalar en un lugar específico y que cumpla con una serie de prácticas para que las ondas electromagnéticas lleguen a su destino. Estas prácticas incluyen tanto la instalación del gateway como el grado de protección del mismo. A nivel de protección, existe el estándar IP que consiste en una puntuación alfanumérica de 6 y 9 dígitos que indica el grado de protección frente al polvo y al agua respectivamente. A nivel de instalación, se ha de tener en cuenta los siguientes factores:

- Proximidad frente a objetos metálicos.
- Protecciones de metal en el dispositivo.
- Recintos que envuelvan al gateway por metal o plástico recubierto de metal.
- Toma a tierra adecuada.
- Montaje físico adecuado.

Considerando los factores de prestaciones y económicos, el departamento de electrónica adquirió el Gateway Multitech IP67 (febrero 2022) con el módulo de LoRaWAN® y GPS incorporados.

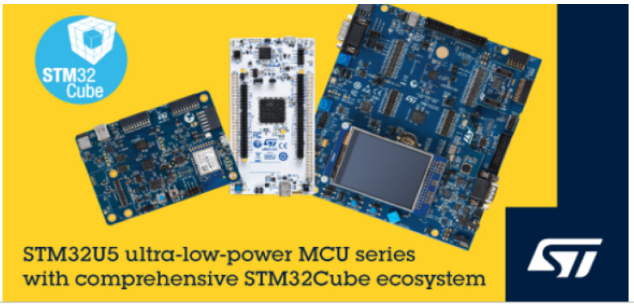
5.2. Dispositivo STM32WL55

La estructura del dispositivo STM32WL55JC1 se puede observar en la Figura 5.2.1. La extensión JC corresponde a la frecuencia de banda LoRa® europea que es igual a 868 MHz. A nivel de desarrollo de software, en la plataforma de desarrollo STM32CUBE MX se encuentran proyectos de diferentes características. La estructura necesaria para la implementación del proyecto se encuentra en la parte de proyectos de red LoRaWAN® con el nombre de *LoRaWAN_End_Node* (Figura 5.2.2.).

Figura 5.2.1. Dispositivo STM32WL5JC1 .

Para programar y depurar los dispositivos de la gama STM32 o STM8 es necesario utilizar ST-LINK. De esta manera, podemos compilar el código y enviar la información a través del ordenador hacia el dispositivo. El enlazador usa el puerto USB del ordenador y el puerto micro-USB del dispositivo como se puede observar en la Figura 5.2.2.

Figura 5.2.2. Enlace y alimentación a través de puerto microUSB y USB .



Examples List: 12 items

Export

*	Name	Board	Board Type	STM32CubeMX Compatible	STM32CubeMX Version	SW Package Installed
☆	LoRaWAN_AT_Slave	NUCLEO-WL55JC1	Nucleo-64	✓	6.5.0	✓
☆	LoRaWAN_AT_Slave	NUCLEO-WL55JC2	Nucleo-64	✓	6.5.0	✓
☆	LoRaWAN_AT_Slave_DualCore	NUCLEO-WL55JC1	Nucleo-64	✓	6.5.0	✓
☆	LoRaWAN_AT_Slave_DualCore	NUCLEO-WL55JC2	Nucleo-64	✓	6.5.0	✓
☆	LoRaWAN_End_Node	NUCLEO-WL55JC1	Nucleo-64	✓	6.5.0	✓
☆	LoRaWAN_End_Node	NUCLEO-WL55JC2	Nucleo-64	✓	6.5.0	✓
☆	LoRaWAN_End_Node_DualCore	NUCLEO-WL55JC1	Nucleo-64	✓	6.5.0	✓
☆	LoRaWAN_End_Node_DualCore	NUCLEO-WL55JC2	Nucleo-64	✓	6.5.0	✓
☆	LoRaWAN_End_Node_DualCoreFreeRTOS	NUCLEO-WL55JC1	Nucleo-64	✓	6.5.0	✓
☆	LoRaWAN_End_Node_DualCoreFreeRTOS	NUCLEO-WL55JC2	Nucleo-64	✓	6.5.0	✓
☆	LoRaWAN_End_Node_FreeRTOS	NUCLEO-WL55JC1	Nucleo-64	✓	6.5.0	✓
☆	LoRaWAN_End_Node_FreeRTOS	NUCLEO-WL55JC2	Nucleo-64	✓	6.5.0	✓

Figura 5.2.3. Panel principal de las plantillas de proyectos LoRaWan® **[IDE: STM32Cube MX]**.

Dentro de este código se puede utilizar diferentes entornos de desarrollo para que el microcontrolador se programe, en este caso se ha optado por utilizar el entorno de desarrollo Keil µVision de la marca ARM. En la Figura 5.2.4 se puede observar como queda el conjunto de carpetas del proyecto. El conjunto está escrito en código C.

Este código se define por ficheros con dos tipos de extensiones. La extensión punto c indica los diferentes archivos ejecutables los cuales serán compilados. En cambio, la extensión punto h se encarga de definir los macros y las variables dentro de un ejecutable c. El compilador empezará a ejecutar nuestros archivos desde el archivo main.c y avanzará una por una las miles de líneas de código dentro del proyecto. Estas líneas han sido definidas a partir de la estructura de un microprocesador y definen los parámetros de reloj para que todo el hardware funcione.

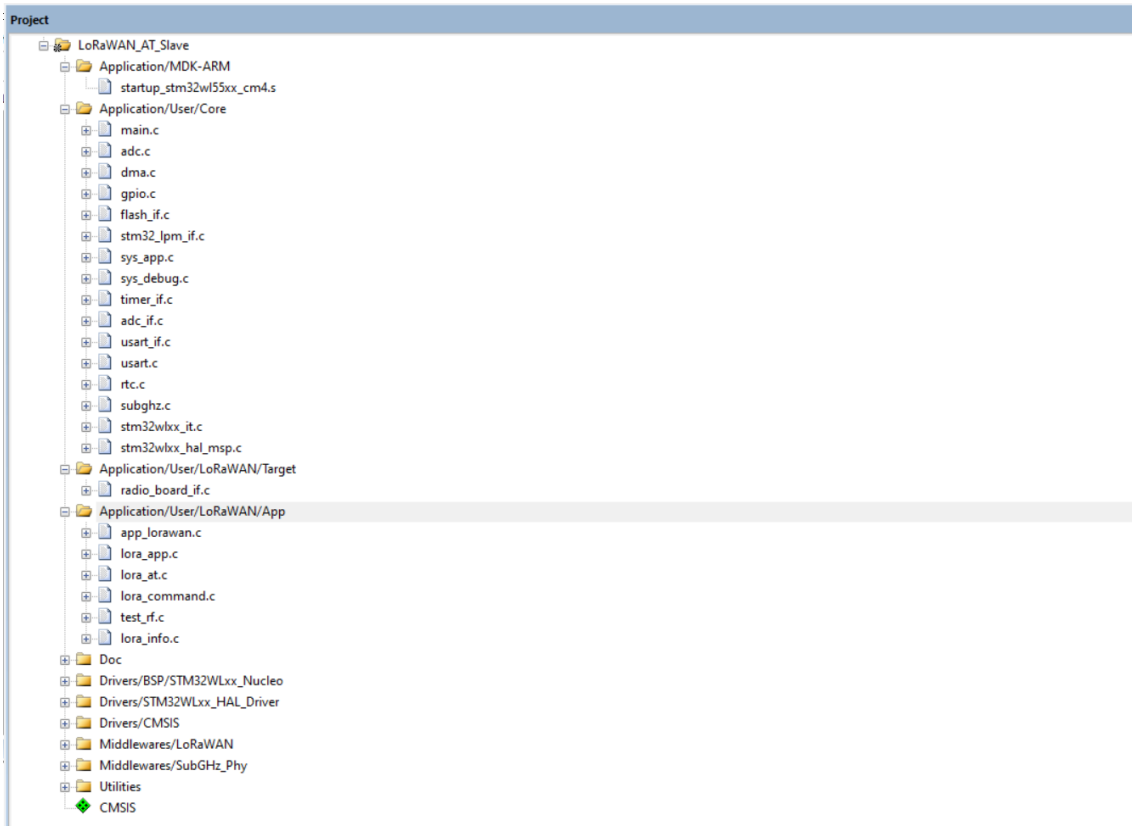


Figura 5.2.4. Carpetas y archivos dentro del proyecto **[IDE: Keil µVision]**.

STM provee una guía en donde se puede ver las partes principales que servirán para construir una aplicación LoRa®. Dentro de este fichero se puede identificar el modelo

operacional (Figura 5.2.5.) del proyecto *LoRaWAN_End_Node*. Este modelo se caracteriza por enviar iterativamente si se activa la petición de unión a la red LoRa® y esperar que exista un sistema de red LoRaWAN® que cuente con un gateway para establecer la comunicación. En caso favorable se establecen una serie de eventos de transmisión hasta que el dispositivo pueda establecer eventos de interrupción de la transmisión (Interrupt Request IRQ) o que ocurra un evento que finalice el protocolo LoRa®. Los datos tardarán más en ser enviados conforme transcurra el tiempo porque el evento temporizador de proceso (Process Timer Event) retrasa el tiempo de envío si se encuentra conectado a la red.

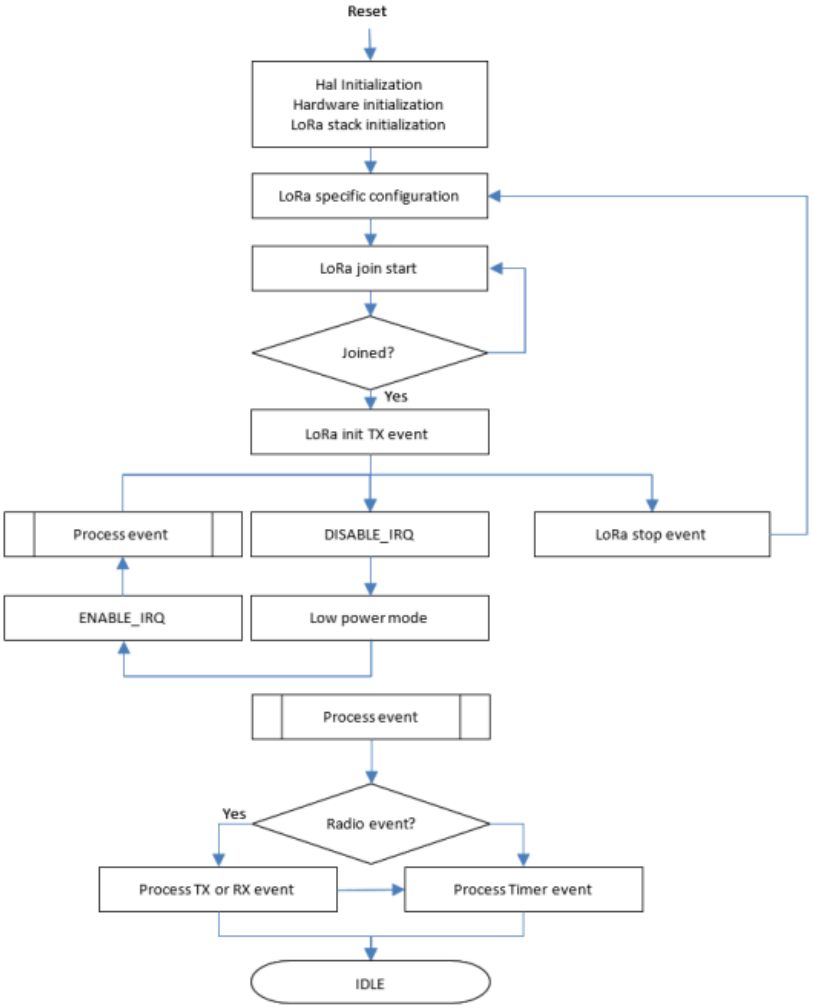


Figura 5.2.5. Modelo operacional dentro del proyecto *LoRaWAN_End_Node* **[19]**.

5.3. Gateway Multitech Conduit IP67

La puerta de enlace o gateway fabricado por Multitech es un dispositivo que incorpora los módulos de comunicación LoRa® y GNSS (ubicación satelital de alta precisión). Estos módulos se encuentran integrados en la tarjeta principal y están protegidos por una estructura metálica de resistencia IP67. La estructura cuenta a su vez con materiales de vidrio que se encargan de recolectar las señales GNSS y LoRa® emitidas en una forma de ondas electromagnéticas. Este componente es alimentado a través de un cable RJ45 el cual se conecta a un transformador de corriente POE.

El transformador cuenta con una entrada de alimentación y dos salidas. La primera salida se conecta con el gateway, mientras que la segunda salida se conecta hacia un ordenador, el cual puede estar ubicado de manera local o remota. Estas salidas se conectan mediante un cable RJ45 como se puede observar en la Figura 5.3.1

Figura 5.3.1. Modelo operacional dentro del proyecto *LoRaWAN_End_Node* .

Una de las principales ventajas es que integra una aplicación que permite configurar las distintas características relacionadas con la red LoRaWAN® Figura 5.3.2.

Esta interfaz se puede acceder configurando el puerto de conexiones de red local desde el ordenador. Se ha de introducir la IP y la Subnet Mask del gateway de valores 192.168.2.1 y 255.255.255.0 respectivamente. Después se ha de acceder a la dirección IP 192.168.2.1 por medio de un navegador web, el cual abrirá una página de sesión por defecto, que se puede personalizar en la sección de administración. Una vez dentro de la aplicación se pueden configurar los siguientes parámetros LoRAWAN®:

- Network Settings, útil para cambiar el plan de frecuencia de banda y las características de transmisión de la red.
- Key Mangement, sirve para registrar cada una de las claves que poseen los nodos finales dentro del proceso de comunicación.
- Gateways, en donde se administran las puertas de enlace dentro de la red LoRaWAN®.
- Devices, útil para ver los dispositivos conectados a la red.
- Devices groups, útil para agrupar diferentes dispositivos.
- Packets, donde se pueden visualizar los paquetes de datos recibidos.
- Downlink Queue, útil si se quiere establecer la comunicación bidireccional entre el gateway y los dispositivos finales.

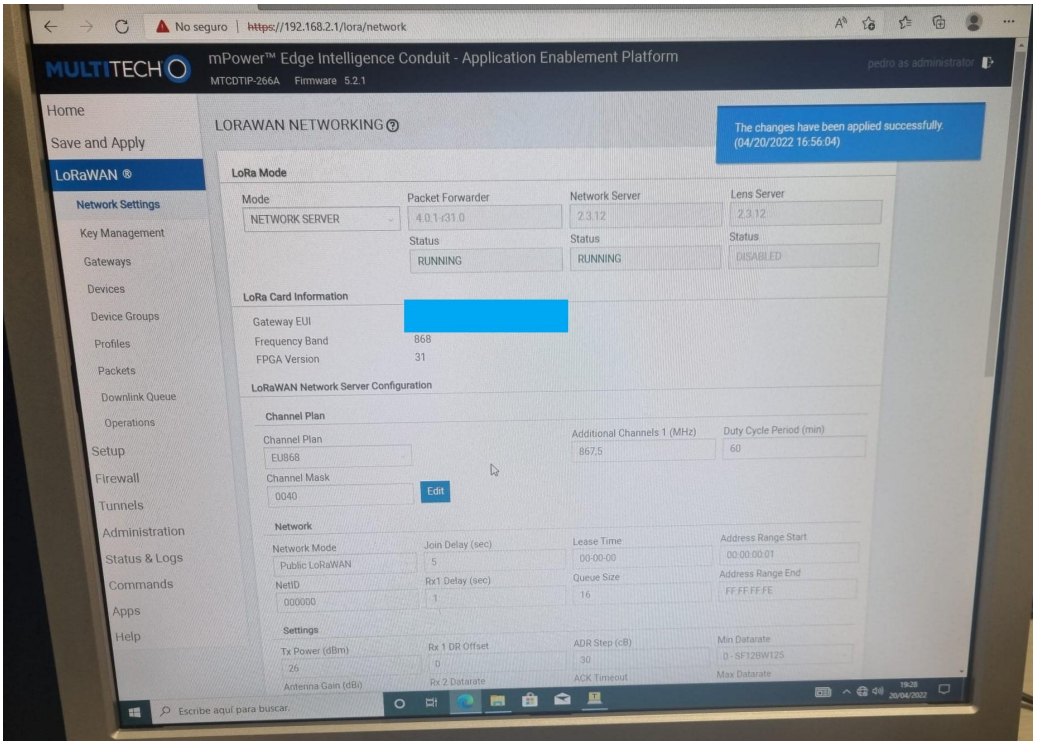


Figura 5.3.2.Capa de aplicación del Multitech Conduit IP67 .

Cabe destacar que el gateway posee un apartado llamado firewall que limita el acceso a determinadas IP 's en caso de que la conexión a la red sea pública. Es decir que si se tuviera que instalar estos sistemas en lugares poco accesibles, al acceder a la red pública de la UPC se vería limitado el acceso a los desarrolladores de aplicaciones IT de la ETSEIB que se encuentren dentro de una IP determinada.

5.4. Conexión entre sistemas

Con el objetivo de establecer la comunicación entre ambos componentes realizamos la programación en código c en el STM32WL55JC1 y el registro del mismo componente dentro del gateway Multitech Conduit IP67. La sesión de red que se establezca durante la comunicación puede ser de dos tipos, ABP (Activación por Personalización) o OTAA (sobre el aire). En nuestro caso debido a que la conexión OTAA brinda flexibilidad porque cada vez que se enlaza el dispositivo la dirección se genera automáticamente (device address).

Esta conexión se establece a través de solicitudes de unión (join request) que se envían al join server cuando el dispositivo se encuentra iniciando el proceso de unión. Además, a la hora de definir la clase de comunicación se ha elegido la clase “A” debido a que es el tipo de clase que se puede alimentar por batería, por lo que posee poco consumo de energía.

En primer lugar, para la configuración de la sesión del dispositivo STM32WL55JC1 se configuraron las líneas de código del fichero *“se-identity.h”*. En este archivo es configurable todos los parámetros para la creación de la sesión. Dentro de este archivo se encuentra la definición del APP EUI (JOIN EUI en código) y el APPKEY. Por un lado, el APP EUI, es la dirección única (ID) del servidor de aplicaciones, es el destino de los mensajes enviados por los dispositivos, única para el servidor de aplicaciones y cada uno de estos servidores debería tener su propio APP EUI. Por otro lado el APP KEY es la llave de encriptación entre la fuente del mensaje DEV EUI y la destinación del mensaje APP EUI.

A nivel físico de componentes tenemos el DEV EUI y el GATEWAY EUI que son códigos únicos establecidos por los fabricantes de la pieza, por ejemplo “00:80:E1:15:00:0A:9F:BF”. Este código está pegado en cada uno de los componentes que fabrica STM. Por último, los valores del network key (NWK KEY), app session key (APP_S_KEY) y network session key (NWK_S_KEY) se han mantenido iguales al del APP KEY. En la Figura 5.4.1. y en la 5.4.2 se pueden observar las líneas de código que

definen los parámetros comentados anteriormente.

```
 * When set to 0 DevEui is automatically set with a value provided
 */
#define STATIC_DEVICE_EUI                                    0

/*!
 * end-device IEEE EUI (big endian)
 */
#define LORAWAN_DEVICE_EUI                                   { 0x00,

/*!
 * App/Join server IEEE EUI (big endian)
 */
#define LORAWAN_JOIN_EUI                                     { 0x01,
```

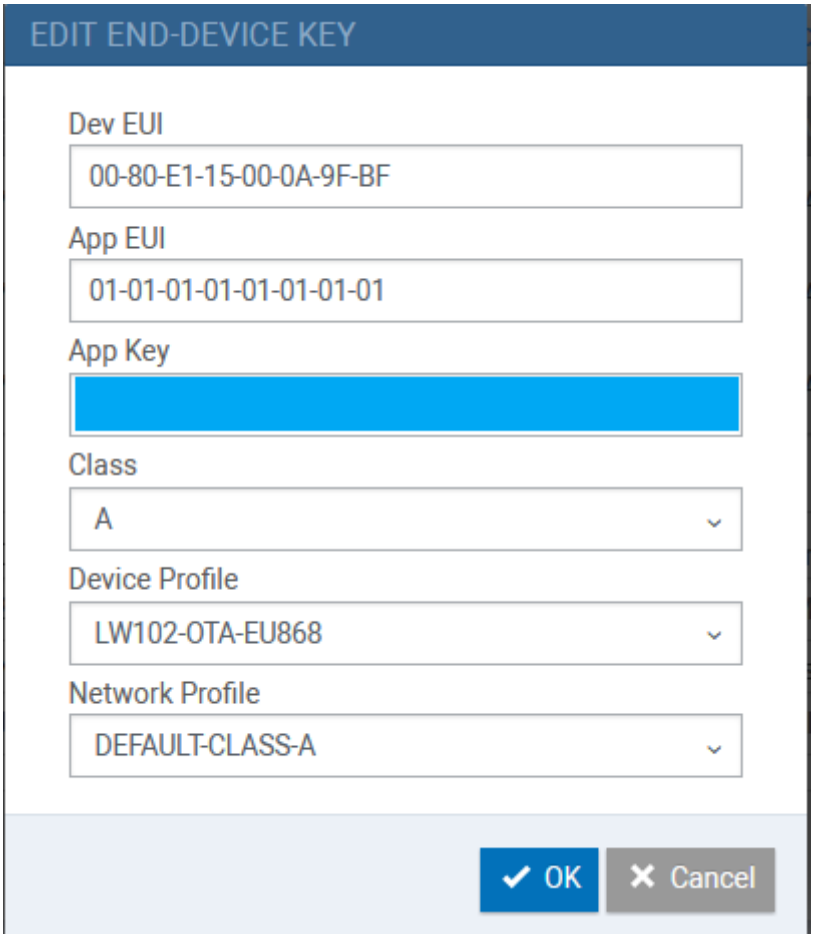
5.4.1. Líneas de código que definen el device EUI **[Keil µVision]**.

```
 */
#define LORAWAN_DEVICE_ADDRESS

/*!
 * Application root key
 */
#define LORAWAN_APP_KEY

/*!
 * Network root key
 */
#define LORAWAN_NWK_KEY

/*!
 * Forwarding Network session key
 */
#define LORAWAN_NWK_S_KEY

/*!
 * Application session key
 */
#define LORAWAN_APP_S_KEY
```

5.4.2. Líneas de código que definen las llaves de la sesión **[Keil µVision]**.

Del mismo modo, se ha de configurar el gateway, que cuenta con servidor LoRaWAN® local, el Join Server Local y el servidor de aplicaciones integrado en la interfaz

MULTITECH. Con estos servidores se establece la conexión con el componente STM32WL55JC1. En la Figura 5.4.3. se puede observar cuáles son las variables a añadir dentro del gateway. Estas variables describen las características a nivel de red LoRaWAN que se han configurado para establecer la conexión.



5.4.3. Configuración del dispositivo STM32WL55JC en el gateway **[Laboratorio de Electrónica]**.

Por último, para comprobar el estado de conexión del dispositivo y del gateway se han consultado dos herramientas, el puerto serial USB (software de Tera Term) y la interfaz del gateway multitech. En el gateway se podía observar con más detalle la información de los paquetes que se habían sido recibidos por parte de todos los dispositivos STM32WL55JC1 conectados, lo cual permitía ver en tiempo real el envío de los distintos dispositivos configurados a lo largo del proyecto (Figura 5.4.4.). Mientras que en el putty se observa en la Figura 5.4.5. que los dispositivos STM32WL55JC1 siempre emiten señales de temperatura mientras se encuentran alimentados por una fuente (fuente de 5V).

Packets

Device EUI	Freq	Datarate	SNR	RSSI	Size	FCnt	Type	Tx/Rx Time	Details
00-80-e1-15-00-0a-99-cf	869.525	SF10BW125	-	-	43	0000000F	DnUnc	12:17:19	
00-80-e1-15-00-0a-99-cf	867.500	SF12BW125	4	-86	28	00000019	UpUnc	12:17:17	
00-80-e1-15-00-0a-9b-b5	867.500	SF12BW125	8	-54	28	00000024	UpUnc	12:16:14	
00-80-e1-15-00-0a-99-cf	868.500	SF12BW125	9	-85	30	00000018	UpUnc	12:15:48	
00-80-e1-15-00-0a-9b-b5	868.300	SF12BW125	12	-55	28	00000023	UpUnc	12:15:48	

5.4.4. Paquetes recibidos por gateway emitidos por el STM32WL55JC **[Gateway Multitech]**.

COM9 - Tera Term VT

File Edit Setup Control Window Help

```
90s101:temp: 27
90s105:TX on freq 867300000 Hz at DR 0
90s108:SEND REQUEST
91s754:MAC txDone
92s788:RX_1 on freq 867300000 Hz at DR 0
92s986:IRQ_RX_TX_TIMEOUT
92s986:MAC rxTimeOut
93s750:RX_2 on freq 869525000 Hz at DR 2
94s090:MAC rxDone

###### ========== MCPS-Confirm =============
100s109:VDDA: 254
100s109:temp: 28
100s113:TX on freq 867700000 Hz at DR 0
100s116:SEND REQUEST
101s762:MAC txDone
102s795:RX_1 on freq 867700000 Hz at DR 0
102s994:IRQ_RX_TX_TIMEOUT
102s994:MAC rxTimeOut
103s758:RX_2 on freq 869525000 Hz at DR 2
104s303:MAC rxDone

###### ========== MCPS-Confirm =============
```

5.4.5. Paquetes de temperatura emitidos por el STM32WL55JC y registrados en el ordenador **[Tera Term]**.

5.5. Comentarios

Dentro de la experimentación se configuró la función que repite el proceso de unión cada vez que el dispositivo se reinicie o vuelva a realizar una actividad después de estar apagado o suspendido. La razón se basa en que al hacer la experimentación se pudo observar que los datos de sesión de red LoRaWAN® quedan grabados en la Memoria No Volátil (NVM). Por tanto, si en algún momento se busca configurar otra red LoRaWAN® los datos pueden ser guardados dentro de la NVM.

```
92
93  /*!
94   * LoRaWAN force rejoin even if the NVM context is restored
95   * @note useful only when context management is enabled by CONTE
96   */
97  #define LORAWAN_FORCE_REJOIN_AT_BOOT                true
98
99  /*!
100  * User application data buffer size
101  */
102 #define LORAWAN_APP_DATA_BUFFER_MAX_SIZE            242
103
104 /*!
105  * Default Unicast ping slots periodicity
106  *
107  * \remark periodicity is equal to 2^LORAWAN_DEFAULT_PING_SLOT_P
108  *         example: 2^4 = 16 seconds. The end-device will open a
```

5.5.1. Líneas de código que define el proceso de reintento de unión al arranque **[Keil µVision]**.

Después que se comprobó el establecimiento de la red LoRaWAN dentro del gateway, se agregó dentro del proyecto la migración de la IP del gateway. Es decir, se cambió la IP del servidor de Semtech (192.168.2.1) hacia una IP pública de la red de UPC. Por tanto, como esta red es potencialmente víctima de un ataque de fuerza bruta por parte de cualquier ordenador UPC o externo, se decidió poner un firewall con el fin de que solo unos cuantos dispositivos puedan obtener el privilegio de entrar a la dirección IP. Por último, el gateway irá conectado directamente a un switch de la facultad sin necesidad de utilizar el transformador POE.

A nivel del firewall, una vez dentro de la aplicación, se configuró el acceso solamente a las direcciones IP que tengan en su estructura el código correspondiente a los ordenadores del ala derecha de la planta 9 en la ETSEIB. Estos ordenadores pertenecen a los del departamento de electrónica, en donde se encuentran los despachos de los profesores y los laboratorios de investigación electrónica.

6. Conclusiones

Una vez realizado el estudio e implementación del presente proyecto, se dispone de la información suficiente para afirmar que la red LoRaWAN® se ha desarrollado con éxito dentro del campus de la ETSEIB.

A futuro sería recomendable establecer más dispositivos agrupados y que puedan enviar señales de alta importancia.

Por ejemplo, se podrían implementar las siguientes aplicaciones:

- La temperatura en el interior de zonas climatizadas para la integración IoT con los sistemas HVAC.
- La potencia de red emitida por las placas fotovoltaicas para la integración IoT con la red general mediante conmutación energética.

7. Agradecimientos

Agradezco a Antonio Calomarde por ser mi tutor durante este proyecto de investigación. Por otro lado, dedico este trabajo a mi familia por estar conmigo apoyándome durante mi estancia en la ETSEIB.

8. Bibliografía

Referencias bibliográficas

[1] ALIANZA LORA ®. LoRaWAN®. *LoRaWAN®, LoRaWAN® para desarrolladores, biblioteca de recursos, LoRa*. Fremont, 2021. [https://blog.lora-alliance.org/, 17 de febrero de 2022].

[2] THE THINGS NETWORK. LORA. *Técnica de modulación LoRa y preguntas frecuentes a la hora de implementar LoRa*. Amsterdam, 2022. [https://www.thethingsnetwork.org/docs/lorawan/what-is-lorawan/, 17 de febrero de 2022].

[3] FINDSTACK. PERSPECTIVAS B2B. *Estadísticas del IoT para el 2022*. Londres, 2021. [https://findstack.com/es/internet-of-things-statistics/, 25 de febrero de 2022].

[4] ZIBTEK®. Arquitectura IoT. *Componentes de una arquitectura IoT*. Utah, 2020. [https://www.zibtek.com/blog/iot-architecture/, 5 de marzo de 2022].

[5] ISO. ISO/IEC 7498-1:1994. *Modelo OSI referencia.* Ginebra, 2000. [https://www.iso.org/standard/20269.html, 5 de marzo de 2022].

[6] LESICS. CÓMO FUNCIONA EL INTERNET. Youtube, 2019. [https://www.youtube.com/watch?v=x3c1ih2NJEg&ab_channel=Lesics, 9 de marzo de 2022].

[7] WIKIPEDIA. DEFINICIÓN COLOQUIAL DE UN PROTOCOLO DE COMUNICACIÓN. [https://es.wikipedia.org/w/index.php?title=Protocolo_de_comunicaciones&oldid=141578147, 9 de marzo de 2022].

[8] SUNFIRE TESTING. LORA FCC GUÍA DE CERTIFICACIÓN. *Características técnicas de una red* LoRaWAN® *operativa.* Illinois, 2018. [https://www.sunfiretesting.com/LoRa-FCC-Certification-Guide/#:~:text=LoRa%20 , 9 de marzo de 2022].

[9] MOUSER. EQUIPO STM32WL55JC1. *Ficha técnica y precio del dispositivo final.* Texas, 2022.

[https://www.mouser.es/ProductDetail/STMicroelectronics/NUCLEO-WL55JC1?qs=zW32dvEIR3uxo98jZ7%2FN0g%3D%3D, 15 de marzo de 2022].

[10] THE THINGS NETWORK. LISTA DE GATEWAYS COMERCIALES. *Estudio de mercado de gateways.* Amsterdam, 2020. [https://www.thethingsnetwork.org/docs/gateways/start/list/, 15 de marzo de 2022].

[11] CISCO. MEJORES PRÁCTICAS PARA INSTALAR UN PUNTO DE ACCESO INALÁMBRICO. San José, 2020. [https://www.cisco.com/c/en/us/td/docs/wireless/access_point/outdoor/video/ap-best-practices.html, 23 de marzo de 2022].

[12] ZULOAGA IZAGUIRRE, A. y ASTARLOA CUÉLLAR, A., 2008. Sistemas de procesamiento digital. Barcelona: Delta. ISBN 9788492453030.

[13] Hassan, Abbas & Awad, Ali Ismail., 2018. Urban Transition in the Era of the Internet of Things: Social Implications and Privacy Challenges. IEEE Access. PP. 1-1. 10.1109/ACCESS.2018.2838339.

[14] BEHRTECH. BEHRTECH BLOG. *Seis tipos principales de tecnología inalámbrica IoT y sus mejores casos de uso*. Toronto. [https://behrtech.com/blog/6-leading-types-of-iot-wireless-tech-and-their-best-use-cases/, 14 de abril de 2022].

[15] Petajajarvi, Juha & Mikhaylov, Konstantin & Pettissalo, Marko & Janhunen, Janne & Iinatti, Jari., 2017. Performance of a low-power wide-area network based on LoRa technology: Doppler robustness, scalability, and coverage. International Journal of Distributed Sensor Networks. Vol. 13. 1-16. 10.1177/1550147717699412.

[16] STM. NUCLEO-WL55JC. *Placa de desarrollo STM32 Nucleo-64 con STM32WL55JCI MCU, SMPS, admite conectividad Arduino y morpho*. Ginebra. [https://www.st.com/en/evaluation-tools/nucleo-wl55jc.html, 17 de abril de 2022].

[17] STM. STM32 32-BIT ARM CORTEX MCUS. *Familia de microcontroladores de 32 bits basados en el procesador ARM Cortex M*. Ginebra. [https://www.st.com/en/microcontrollers-microprocessors/stm32-32-bit-arm-cortex-mcus.html, 19 de abril de 2022].

[18] STM. STM32WL SERIES. *Portafolio STM32WL*. Ginebra. [https://www.st.com/en/microcontrollers-microprocessors/stm32wl-series.html,2 de

mayo de 2022].

[19] STM. NOTA EXPLICATIVA. *Cómo construir una aplicación LoRa® con STM32Cube*. Ginebra. [https://www.st.com/resource/en/application_note/an5406-how-to-build-a-lora-application-with-stm32cubewl-stmicroelectronics.pdf,4 de mayo de 2022].

[20] THE THINGS NETWORK. DEVICES CLASSES. Tipos de clases dentro de la red *LoRaWAN®*. Ginebra. [https://www.thethingsnetwork.org/docs/lorawan/classes/,4 de mayo de 2022].

[21] SEMTECH. What are LoRa® and LoRaWAN®?. *Información técnica sobre la red LoRa® and LoRaWAN®* . California. [https://lora-developers.semtech.com/documentation/tech-papers-and-guides/lora-and-lorawan/#:~:text=A%20proprietary%20spread%2Dspectrum%20modulation,KHz%20(for%20downlink%20channels).,5 de mayo de 2022].

[22] VISUAL ELECTRIC. How LoRa Modulation really works. *Información sobre la modulación LoRa®*. California. [https://www.youtube.com/watch?v=jHWepP1ZWTk&ab_channel=VisualElectric,9 de mayo de 2022].

[23] L. Vangelista, "Frequency Shift Chirp Modulation: The LoRa Modulation," in IEEE Signal Processing Letters, vol. 24, no. 12, pp. 1818-1821, Dec. 2017, doi: 10.1109/LSP.2017.2762960.

[24] Kim, Dong-Hoon & Lee, Eun-Kyu & Kim, Jibum. (2019). Experiencing LoRa Network Establishment on a Smart Energy Campus Testbed. Sustainability. 11. 1917. 10.3390/su11071917.

[25] Liando, Jansen & Gamage, Amalinda & Tengourtius, Agustinus & Li, Mo. (2019). Known and Unknown Facts of LoRa: Experiences from a Large-scale Measurement Study. ACM Transactions on Sensor Networks. 15. 1-35. 10.1145/3293534.