



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



Escuela Técnica Superior de Ingeniería del Diseño
Universitat Politècnica de València

Trabajo Fin de Grado

Ingeniería en Electrónica Industrial y Automática

**DESARROLLO DE UN NODO INALÁMBRICO DE BAJO
CONSUMO BASADO EN TÉCNICAS DE IOT PARA
MONITORIZACIÓN DE PATRIMONIO CULTURAL**

Documentos:

1. Memoria
2. Planos
3. Pliego de condiciones
4. Presupuesto
5. Anexos:
 - 5.A1. Código
 - 5.A2. Hojas de características

Autor:

D. Jaime Laborda Macario

Tutor:

D. Ángel Perles Ivars

Valencia, junio de 2018

*Gracias a mi tutor Ángel Perles Ivars
por su gran ayuda durante el desarrollo del proyecto.
A mi familia y a mi pareja por su apoyo incondicional.*

Resumen

De la misma forma que Internet ha conectado a personas de todo el mundo, el Internet de las cosas está pensado para conectar objetos y máquinas a nuestro alrededor. Todo, desde las luces de la calle, hasta los parquímetros, se comunicarán para hacer que nuestras ciudades y hogares sean más seguros, más energéticamente eficientes y en definitiva mejores lugares para vivir. Este trabajo examina la viabilidad de conectar las obras de arte y patrimonio cultural en general a la red de Internet de las cosas, con el objetivo de monitorizarlas y poder conocer cómo afectan los parámetros ambientales a su conservación antes de que se produzca el daño.

Se desarrolla un dispositivo sensor inalámbrico de bajo consumo capaz de medir algunos de los parámetros ambientales que afectan a la conservación y mantenimiento de obras de arte pertenecientes al patrimonio cultural, y los envía inalámbricamente haciendo uso de una red LoRaWAN. Posteriormente estos parámetros podrán ser tratados y analizados en la nube mediante técnicas estadísticas de *big data* para poder obtener sugerencias de actuación o generar alarmas que alerten de cualquier anomalía detectada.

En concreto, se hace un diseño de la electrónica que integra los sensores con el transmisor y de la programación del nodo. También se diseña la red, haciendo la instalación de un punto de acceso a la red LoRaWAN. Por último, se diseña una aplicación web simple para visualizar los datos en tiempo real.

Palabras clave: IoT, Conservación, Patrimonio cultural, Arte, LoRaWAN, ARM Cortex-M0, Bajo consumo, Nodo inalámbrico, LPWAN, Sensor inalámbrico

Summary

Just like the Internet has connected people all around the World, the Internet of Things is built to connect objects and machines around us. Everything from streets lights to parking meters, will communicate to make our cities and homes safer, more energy efficient and ultimately, better places to live. This work examines the feasibility of connecting artworks and cultural heritage in general to the Internet of Things, with the aim of monitoring them and being able to know how environmental parameters affect their conservation before damage occurs.

A wireless low power sensor is developed that measures some of the most important parameters that affect conservation of artworks part of the cultural heritage and sends them wirelessly using LoRaWAN protocol. These parameters can be then treated and analysed in the cloud by statistical techniques of *big data* to get suggestions for action or generate alarms to alert of any anomaly detected.

Specifically, an electronics design is made that integrates the sensors with the transmitter and the programming of the node. The network is also designed, making the installation of an access point to the LoRaWAN network. Finally, a simple web application is design to visualize the data in real time.

Key words: IoT, Conservation, Cultural heritage, art, LoRaWAN, ARM Cortex-M0, Low Power, Wireless node, LPWAN, Wireless sensor

Resum

De la mateixa forma que la Internet ha connectat a persones de tot el món, la Internet de les Coses està pensat per connectar objectes i màquines al nostre voltant. Tot, des de les llums del carrer, fins els parquímetres, es comunicaran per fer que les nostres ciutats i llars siguin més segurs, més energèticament eficients i en definitiva millors llocs per a viure. Aquest treball examina la viabilitat de connectar les obres d'art i patrimoni cultural en general a la xarxa de la Internet de les Coses, amb l'objectiu de monitoritzar-les i poder conèixer com afecten els paràmetres ambientals a la seua conservació abans que es produïska el dany.

Es desenvolupa un dispositiu sensor de baix consum sense fils que mesura alguns dels paràmetres ambientals més importants que afecten a la conservació y manteniment de les obres d'art pertanyents al patrimoni cultural i els envia sense fils fent ús d'una xarxa LoRaWAN. Posteriorment aquests paràmetres podran ser tractats i analitzats en el núvol mitjançant tècniques estadístiques de *big data* per poder obtenir suggeriments d'actuació o generar alarmes que alerten de qualsevol anomalia detectada.

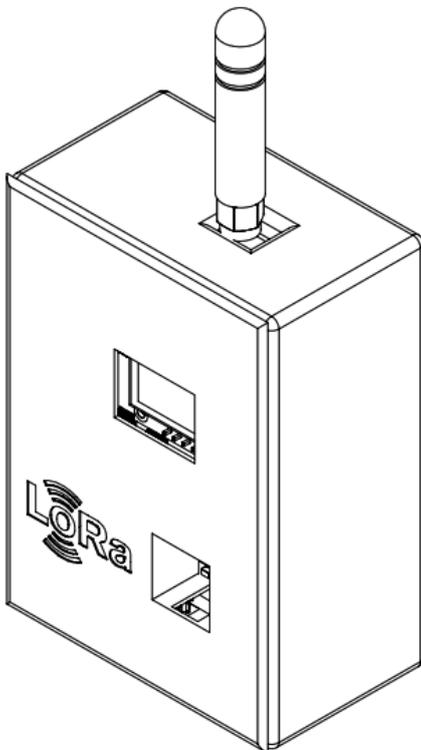
En concret, es fa un disseny de l'electrònica que integra els sensors amb el transmissor i de la programació del node. També es dissenya la xarxa, fent la instal·lació d'un punt d'accés a la xarxa LoRaWAN. Finalment, es dissenya una aplicació web simple per a visualitzar les dades en temps real.

Paraules clau: IoT, Conservació, Patrimoni cultural, art, LoRaWAN, ARM Cortex-M0, Baix consum, Node sense fils, LPWAN, Sensor sense fils



DESARROLLO DE UN NODO INALÁMBRICO DE BAJO CONSUMO BASADO EN TÉCNICAS DE IOT PARA MONITORIZACIÓN DE PATRIMONIO CULTURAL

1. MEMORIA



Autor:

D. Jaime Laborda Macario

Tutor:

D. Ángel Perles Ivars

Valencia, junio de 2018

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para
monitorización de patrimonio cultural

Índice de contenido

1. Objeto	1
2. Antecedentes.....	1
3. Estudio de necesidades, factores a considerar: limitaciones y condicionantes.	2
3.1. Magnitudes a medir	2
3.2. Versatilidad.....	3
3.3. Consumo energético. Baterías	3
3.4. Transmisión inalámbrica. Cobertura.....	4
3.5. Integración en la obra	4
4. Planteamiento de soluciones alternativas y justificación de la solución adoptada.....	5
4.1. Elección de los sensores	5
4.1.1. Sensor de temperatura y humedad.....	7
4.1.1.1. Sensirion SHT31	7
4.1.1.2. Texas Instruments LM35	8
4.1.1.3. Maxim Integrated DS18B20.....	8
4.1.2. Sensor de luminosidad.....	9
4.1.2.1. Fotorresistencia LDR.....	9
4.1.2.2. AMS TSL2591	9
4.1.2.3. Sillicon Labs Si1133	10
4.1.2.4. Rhom Semiconductors BH1750FV.....	10
4.2. Elección de la tecnología inalámbrica.....	11
4.2.1. WiFi.....	12
4.2.2. Bluetooth Low Energy	13
4.2.3. ZigBee.....	14
4.2.4. RFID.....	17
4.2.5. Telefonía móvil GSM/GPRS/3G/LTE	17
4.2.6. LPWAN	19
4.2.6.1. LoRaWAN.....	22
4.2.6.2. Sigfox.....	23
4.2.6.3. DASH7.....	24
4.2.6.4. NB-IoT	25
4.2.6.5. Ingenu RPMA	26
4.3. Microcontrolador	27

Memoria

4.3.1. Arduino.....	28
4.3.2. ARM Cortex M0+.....	30
4.3.3. Expressif ESP32	32
4.4. Transceptores LoRa	33
4.4.1. HopeRF RFM95W.....	34
4.4.2. RAK Wireless RAK811.....	35
4.4.3. Murata CMWX1ZZABZ	36
4.5. Plataforma de programación (IDE)	38
4.5.1. ARM Keil uVision	38
4.5.2. ARM Mbed	39
4.5.3. Arduino IDE.....	39
4.6. Batería	41
4.6.1. Batería de ion de litio (Li-Ion)	41
4.6.2. LiFePo ₄	43
4.6.3. NiCa	44
4.6.4. NiMH	44
4.6.5. Li-FeS ₂	44
4.6.6. Li-SOCl ₂	46
4.6.7. Supercondensadores	46
4.6.8. Tabla comparativa.....	47
5. Descripción detallada de la solución adoptada.....	48
5.1. Desarrollo del sistema sensor inalámbrico	50
5.1.1. Desarrollo electrónico tarjeta sensores	50
5.1.1.1. Diseño del bus I2C	53
5.1.1.2. Sensor temperatura y humedad SHT31	56
5.1.1.3. Sensor luminosidad AMS TSL2591	59
5.1.1.4. Transceptor RFM95.....	60
5.1.1.5. Circuito monitorización de la batería	62
5.1.1.6. Regulador lineal 3,3V	63
5.1.2. Murata LoRaWAN CMWX1ZZABZ	63
5.1.3. Programación.....	64
5.1.3.1. Biblioteca sensor SHT31	64
5.1.3.2. Biblioteca sensor TSL2591	65
5.1.3.3. Prueba sensores	66
5.1.3.4. Aplicación End-Node final.....	66

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para
monitorización de patrimonio cultural

5.2. Pasarela LoRaWAN.....	68
5.2.1. Dragino LG01-S	69
5.2.2. Multitech Conduit	70
5.2.3. Instalación del Multitech Conduit.....	70
5.3. Plataforma cloud recolectora de datos (TTN)	72
5.4. Aplicación final de usuario	75
5.5. Carcasa de protección	79
5.6. Pruebas de cobertura	81
5.7. Dimensionado de la batería y autonomía	83
6. Conclusiones	85
6.1. Sobre el trabajo realizado.....	85
6.2. Mejoras.....	85
7. Bibliografía.....	87

Índice de figuras

Figura 1 - Dominio analógico vs digital.....	6
Figura 2- Sistema de medida (bucle abierto).....	6
Figura 3 - SHT31 montado en placa de evaluación de Adafruit.....	7
Figura 4 - Sonda sensor DS18B20	8
Figura 5 - Fotorresistencia LDR	9
Figura 6 - Diagrama de bloques del sensor TSL2591	9
Figura 7 - Sensor Si1133	10
Figura 8 - Módulo BH1750FV.....	11
Figura 9 - Comparativa tecnología alámbrica e inalámbrica (studiopsis.com)	11
Figura 10 - Logo certificación WiFi.....	12
Figura 11 - Modelo OSI WiFi (monografías.com)	12
Figura 12 - Bluetooth Smart Logo	13
Figura 13 - Transacciones GATT	14
Figura 14 - Topología Malla Zigbee.....	15
Figura 15 - Ejemplo de etiqueta RFID activa.....	17
Figura 16 - Descripción general de las tecnologías empleadas en comunicación móvil.....	18
Figura 17 - Módulo GSM con slot para tarjeta SIM.....	19
Figura 18 - Diagrama cobertura Bluetooth	20
Figura 19 - Diagrama cobertura LPWAN.....	20
Figura 20 - Comparación tecnologías inalámbricas (IoTForAll.com)	21
Figura 21 - Logo LoRa	22
Figura 22 - Tipología de red LoRaWAN	23
Figura 23 - Logo Sigfox.....	23
Figura 24 - Cobertura Sigfox Europa.....	24
Figura 25 - DASH7 Alliance logo.....	25
Figura 26 - Estructura de un microcontrolador(aprendiendoarduino.com).....	27
Figura 27 – Placa desarrollo Arduino Uno.....	28
Figura 28 - Diagrama de bloques ATmega328P	29
Figura 29 - Arduino MKR WAN 1300 con precio	30
Figura 30 - Tarjeta de desarrollo con el módulo ESP-WROOM-32	32
Figura 31 - Módulo HopeRF FRM95W	34
Figura 32 - Módulo RAK811 de RAK Wireless	35
Figura 33 - Placa de desarrollo WisNode-Lora.....	36
Figura 34 - Murata módulo LoRa CMWX1ZZABZ	36

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para
monitorización de patrimonio cultural

Figura 35 - Diagrama de bloques CMWX1ZZABZ.....	37
Figura 36 - B-L072Z-LRWAN1 - STM32L0 Discovery kit LoRa	37
Figura 37 - Ventana principal Keil uVision5 para microcontroladores.....	38
Figura 38 - Compilador ARM Mbed en la nube	39
Figura 39 - Arduino IDE 1.8.5.....	40
Figura 40 - Pack de baterías de un vehículo Tesla Model S	41
Figura 41 - Curva de carga CC-CV batería Li-Ion (portatilmovil.com).....	42
Figura 42 – Curva de descarga de una celda de iones de litio (PowerTechSystems.eu).....	43
Figura 43 - Curva descarga batería LiFePo ₄ (benergytech.com).....	43
Figura 44 - Curva descarga batería NiMH (priuschat.com)	44
Figura 45 - Comparación Li-FeS ₂ con Alcalina (Energizer)	45
Figura 46 - Curva de descarga característica Saft LS14500EX (Saft Batteries) [34]	46
Figura 47 - Supercondensador fino	46
Figura 48 - Diagrama implementación nodo inalámbrico.....	48
Figura 49 – Diagrama tipología con varios nodos	49
Figura 50 - Arduino Uno con Shield controlador de motores	50
Figura 51 - LoRa Sensor Shield con RFM95.....	51
Figura 52 - Montaje final electrónica	51
Figura 53 – Render 3D del layout extraído de Allegro PCB Editor	52
Figura 54 - Diagrama de bloques funcionales electrónica	52
Figura 55 - Esquema eléctrico bus I2C (Wikipedia).....	53
Figura 56 - Implementación resistencias pullup I2C	54
Figura 57 - Definición de pines SHT31.....	56
Figura 58 - Implementación SHT31 en el esquema eléctrico	56
Figura 59 - Corte zona sensores PCB.....	58
Figura 60 - Implementación TSL2591 en el esquema eléctrico	59
Figura 61 - Detalle Solder Bridge en prototipo real.....	59
Figura 62 - Implementación modulo RF	60
Figura 63 - Implementación convertor de niveles con SN74LVC125.....	61
Figura 64 – Conector SMA y UFL y su implementación en el esquema	61
Figura 65 - Calculo de la impedancia característica de la línea.....	61
Figura 66 - Implementación en PCB de la antena	62
Figura 67 - Implementación monitorización batería.....	62
Figura 68 - Implementación LDO 3.3V en el esquema eléctrico.....	63
Figura 69 - Arquitectura del firmware	64

Memoria

Figura 70 - Captura gráficas STMStudio	66
Figura 71 - Relación tasa de propagación con consumo, velocidad y alcance	67
Figura 72 - Diagrama básico LoRaWAN (LoRa Alliance)	68
Figura 73 - Gateway Dragino LG01-S	69
Figura 74 - Miltitech Conduit	70
Figura 75 - Multitech Conduit con transceptor LoRa	70
Figura 76 - MultiConnect Conduit instalado y conectado al router.....	71
Figura 77 - Programa sniffer escaneo de red.....	72
Figura 78 - Configuración packet forwarder	72
Figura 79 – Comunidad The Things Network Valencia	73
Figura 80 - Consola aplicación TTN.....	74
Figura 81 - Vista principal plataforma web	75
Figura 82 - Programación Node-Red aplicación usuario	76
Figura 83 - Gráficas temporales aplicación	76
Figura 84 - Información de los sensores	77
Figura 85 - Payload TTN en crudo	78
Figura 86 - Renderizado 3D del nodo sensor montado en la carcasa	79
Figura 87 - Detalle abertura sensores.....	79
Figura 88 - Renderizado 3D vista explotada	80
Figura 89 - Prototipo completo instalado junto a un cuadro.....	80
Figura 90 - Radio de alcance conseguido	81
Figura 91 - Paquete recibido a 600 metros de la pasarela	82
Figura 92 - Perfil de elevación del terreno.....	82
Figura 93 - Cronograma de los estados del ciclo de envío.....	84

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para
monitorización de patrimonio cultural

Índice de tablas

Tabla 1 - Comparativa Sensores Temperatura y Humedad	7
Tabla 2 - Comparación tecnologías LPWAN (Flespi.com).....	21
Tabla 3 - Comparativa NB-IoT (eSmartCity.es).....	26
Tabla 4 - Variantes RFM95W de HopeRF.....	35
Tabla 5 - Comparativa baterías.....	47
Tabla 6 - Configuración dirección I2C	57
Tabla 7 - Tabla verdad Buffer Triestado.....	60
Tabla 8 - Estado de los módulos en cada estado.....	83
Tabla 9 - Consumos de los integrados.....	83
Tabla 10 - Suma de los consumos.....	84
Tabla 11 - Gasto energético que representa realizar un envío.....	84

Memoria

Glosario de siglas y abreviaturas

ADC	Conversor analógico a digital
API	Interfaz de programación de aplicaciones
BLE	Bluetooth de baja energía
BOM	Bill of materials. Lista de materiales
CAD	Diseño asistido por computadora
DAC	Conversor digital a analógico
DFN	Dual Flat No Lead,
DNP	Do Not Place/Populate (No montar)
EDA	Automatización de diseño electrónico
GND	Ground (0V)
HAL	Capa de abstracción de hardware
I2C	Circuito inter-integrado
IC	Circuito integrado
ID	Número de identificación
IDE	Entorno de desarrollo integrado
IOT	Internet de las cosas
IP	Protocolo de Internet
JSON	Notación para objetos JavaScript
LDO	Regulador lineal de baja caída de tensión
LED	Diodo emisor de luz
LORA	Long Range
LORAWAN	Long Range Wide Area Network
LPLAN	Red de área local de baja energía
LPWAN	Red de área extendida de baja energía
LRWAN1	Placa de evaluación de STMicroelectronics
M2M	Machine to Machine
MCU	Microcontrolador
MISO	Entrada al master, salida al esclavo
MOSI	Salida al master, entrada al esclavo
MQTT	Message Queuing Telemetry Transport
PCB	Tarjeta de circuito impreso
PCBA	Tarjeta de circuito impreso ensamblada
PTH	Tecnología de agujero pasante
QFN	Quad-Flat No-leads
RAM	Memoria de acceso aleatorio
RFID	Identificación por radiofrecuencia
ROM	Memoria de solo lectura
RTC	Reloj de tiempo real
SDK	Kit de desarrollo de software
SMD	Tecnología de montaje superficial
SOC	Sistema en chip
SOM	Sistema en módulo
SPA	Aplicación de una sola página
SPI	Interfaz de periféricos serie
STM	STMicroelectronics

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para
monitorización de patrimonio cultural

TTN	The Things Network
UART	Transmisor-Receptor Asincrono Universal
USB	Bus Serie Universal
VCC	Voltaje común de colector (alimentación positiva IC)
VSS	Alimentación negativa IC

Jaime Laborda Macario
Trabajo fin de grado

Memoria

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

1. Objeto

El objeto del presente proyecto es desarrollar un dispositivo inalámbrico conectado de bajo consumo energético para monitorización de patrimonio cultural. Para ello se hará uso de técnicas de Internet of Things.

El dispositivo será capaz de medir parámetros ambientales que afectan a la conservación y mantenimiento de arte pertenecientes al patrimonio cultural situadas tanto en interiores como exteriores, tales como murales, esculturas y cuadros.

El dispositivo estará conectado a Internet a través de una LPWAN (Low Power Wide Area Network) y será capaz de transmitir los datos tomados por los sensores periódicamente en tiempo real para poder ser consultados y analizados por expertos conservadores de patrimonio, así como generación de alertas si los parámetros sobrepasan un valor umbral definido

El dispositivo estará alimentado por baterías que deberán garantizar una autonomía superior a un año.

2. Antecedentes

El patrimonio histórico es un legado que nos vincula con el pasado y lo actualiza, es la herencia que nos dejaron nuestros antepasados y es de vital importancia no olvidarlo, conservarlo y mantenerlo.

Lo que se ha venido haciendo estos años ha sido una restauración correctiva de los bienes de patrimonio, es decir, cuando algo requería la atención de un restaurador, se reparaba. Pero en estos casos, como en la medicina, la mejor opción es realizar una conservación preventiva que se anticipe al peligro antes de que este ocurra. Actualmente, la tecnología nos permite hacer un diagnóstico temprano sirviéndonos de sensores que monitoricen en tiempo real todos los parámetros que afectan al mantenimiento de estas obras.

Cada vez se habla más del Internet de Las Cosas; dispositivos y máquinas conectadas vertiendo gran cantidad de datos a la red, capaces de interactuar con el medio y las personas. Hasta ahora, toda la información de los objetos y cosas se perdía, ya que no éramos capaces de recogerla y transmitirla, pero actualmente con las redes de comunicación existentes, es relativamente sencillo transmitir esta información en cualquier momento y lugar.

La monitorización del patrimonio mediante nodos conectados ofrece una fuente importantísima de información para los conservadores, ya que esta gran cantidad de datos permite realizar análisis mediante diversos algoritmos para predecir el comportamiento futuro de dichas obras y así poder actuar con tiempo antes de que se produzca el daño.

Actualmente, hay diversos proyectos trabajando en este ámbito, como el proyecto *Smart Patrimonio* de Telefónica [1] en el que ya se han instalado 60 sensores en la Muralla de Ávila. Pero estas soluciones propietarias ofrecen poca versatilidad y están basadas en una red de comunicaciones cerrada, ya existente y no pensada para este fin. Dichas soluciones hacen uso de redes de gran ancho de banda como WiFi o 3G para la transmisión de los datos, esto hace que el consumo de los dispositivos se eleve, no siendo aptas para sensores inalámbricos pensados para monitorización de patrimonio.

Memoria

3. Estudio de necesidades, factores a considerar: limitaciones y condicionantes.

El primer paso en la realización de cualquier proyecto trata de realizar un estudio de las necesidades que se pretenden cubrir, así como de los diferentes ámbitos que se desean abarcar, con el objetivo de marcarse unos objetivos y especificaciones claros que eviten que el proyecto se desvíe de los temas importantes.

A continuación, se describen con detalle los factores y necesidades a tener en cuenta en la implementación del nodo inalámbrico de bajo consumo.

3.1. Magnitudes a medir

Es importante definir con exactitud las magnitudes que se van a medir con el sensor inalámbrico y estas dependerán de los parámetros realmente importantes a la hora de conservar una obra.

A menudo las obras son expuestas en vitrinas con el objetivo de protegerlas de ataques vandálicos o daños físicos, no obstante, estas vitrinas, normalmente de vidrio, presentan un gran problema y es que sufren de un efecto invernadero por el que los rayos de luz incidentes quedan atrapados haciendo que se cree un microclima de muy altas temperaturas que son muy perjudiciales para los objetos u obras expuestos en el interior. También resulta muy importante, controlar el espectro de la luz incidida, ya que, por ejemplo, la radiación azul sobre los 550nm y la ultravioleta de unos 450nm tienen componentes muy dañinos para las obras y objetos físicos en general [2].

Para dar solución a esta problemática y con el objetivo de percibir las situaciones dañinas para las obras, se medirán los parámetros de **temperatura** y **luminosidad incidente**.

El siguiente agente ambiental que más impacto tiene en la obra es el agua. En concreto el agua en forma vaporosa suspendida en el aire, también conocida como **humedad relativa**. La humedad relativa es una fracción de la máxima cantidad de agua que puede ser concentrada en el aire y a menudo se expresa en porcentaje. Los cambios de humedad relativa en el ambiente son especialmente dañinos si se producen en un espacio corto de tiempo, pues el material de la pieza expuesta no es capaz de adaptarse, especialmente si se trata de madera. Por tanto, es de vital importancia controlar estos cambios en la humedad relativa con el objetivo de poder evitar que estos cambien rápidamente y poder actuar en la obra en cuestión, ya sea llevándola a una sala diferente, o retirándola de la exposición al público para su descanso en un sitio con una humedad mucho más controlada. Además, las altas humedades, pueden favorecer el crecimiento de materia orgánica viva que puede deteriorar y dañar la pieza, cuadro o edificio.

Por tanto, el proyecto se centrará en monitorizar estos tres parámetros: temperatura, humedad y luminosidad, aunque existen otros muchos parámetros secundarios.

El nodo sensor deberá ser capaz de leer los parámetros de los sensores en crudo y enviarlos a una plataforma *cloud* donde serán procesados y analizados en detalle para la generación de gráficas o alertas.

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

3.2. Versatilidad

Un factor diferencial en el desarrollo de este proyecto es la versatilidad del producto final. El nodo inalámbrico ha de ser capaz de adaptarse a varias situaciones que se dan en la monitorización del patrimonio cultural. Cada tipo de patrimonio cultural tiene unas necesidades determinadas y unos parámetros que interesan más a la conservación de estos.

Para ello el nodo ha sido diseñado teniendo en cuenta la versatilidad de uso, y por tanto el mismo se adapta tanto a una obra de arte pictórico tipo cuadro, como a una escultura de piedra o incluso un edificio pictórico como puede ser una iglesia.

El nodo incorporará varios tipos de sensores que harán que pueda adaptarse a las circunstancias de la aplicación final, para ello el nodo hará uso de un microcontrolador cuyo programa podrá adaptarse para leer y transmitir la información de un sensor u otro. La electrónica también será diseñada de manera modular, de tal manera que permita la inclusión de unos sensores u otros para abaratar costes de fabricación. Por ejemplo, si no se requiere el sensor de luminosidad, no se montará en la placa electrónica ahorrando el coste de este componente.

3.3. Consumo energético. Baterías

En el pasado el método tradicional de monitorización de obras de patrimonio cultural consistía en costosos equipos de monitorización a los cuales se les conectaba por cable diversos sensores que eran llevados a las zonas necesarias. Esto representa un problema, en primer lugar, porque obliga a instalar cables en zonas a menudo de difícil acceso, y, en segundo lugar, obliga a realizar un buen diseño de señal para evitar interferencias y ruidos en los cables.

Preocupa especialmente el consumo energético del nodo inalámbrico, ya que en muchos casos este será instalado en sitios de difícil acceso a una toma de corriente, incluso en el exterior. Además, el cambio de baterías representa un gran desembolso económico, ya no por el coste de las baterías en sí, si no por el coste de mano de obra que representa.

El dispositivo, por tanto, será diseñado haciendo uso de modos de ultra-baja energía para minimizar el consumo energético al máximo y se buscará un compromiso entre el envío en tiempo real de los datos y el consumo energético necesario para dicho envío y toma de datos.

Para ello el diseño electrónico se llevará a cabo teniendo en cuenta todo esto, y será un criterio de peso en la elección de los componentes, y en el especial el de los sensores, el bajo consumo energético de los mismos, especialmente en los modos "standby", buscando siempre un compromiso entre funcionalidad y el consumo más bajo posible.

El desarrollo del firmware para el microcontrolador se realizará haciendo uso de los modos de baja energía disponibles, y será un criterio de peso en la elección del microcontrolador que su consumo en modo dormido sea el mínimo posible.

Para la transmisión inalámbrica de la información se seguirá la misma filosofía y por tanto se buscarán métodos nuevos de transmisión inalámbrica de baja energía con aplicación al IoT.

Debido a la versatilidad del dispositivo, será alimentado por pilas no recargables que permitan el funcionamiento autónomo sin intervención del nodo inalámbrico por un periodo considerable de tiempo (en principio superior a uno o dos años). Para ello se realizará un estudio del estado del arte en cuanto a baterías primarias y no primarias se refiere.

3.4. Transmisión inalámbrica. Cobertura

Tradicionalmente, las soluciones existentes en el mercado para sensores de monitorización de parámetros hacían uso de una transmisión alámbrica de los datos usando protocolos propietarios de cada compañía que a menudo son difíciles de integrar junto con otras tecnologías y requieren de la contratación del servicio de soporte de la compañía o empresa suministradora.

Las soluciones inalámbricas que tradicionalmente existen tampoco nos sirven, al usar protocolos de comunicación propietarios y normalmente de bajo alcance, basados en redes de tipo WiFi, Bluetooth o radio frecuencia de 433MHz. Estas soluciones son válidas para un entorno controlado como un museo, pero no nos sirven cuando se trata de monitorizar varios edificios, esculturas o puntos situados en exteriores.

Es por ello por lo que se busca una solución versátil, válida para cualquier situación que se le pueda plantear al conservador de patrimonio. Esta solución pasa por buscar un protocolo de comunicación inalámbrico de largo alcance, que permita una mejora tanto en el consumo energético utilizado para la transmisión, como en el alcance o cobertura de este. Se estudiarán varias tecnologías innovadoras LPWAN para finalmente realizar la integración de una de ellas en el nodo inalámbrico.

3.5. Integración en la obra

Uno de los aspectos a tener en cuenta en el diseño del sensor inalámbrico será su integración en la obra de arte o patrimonio. Para ello se realizará un estudio buscando un compromiso entre el lugar ideal que permite realizar las medidas con mayor exactitud y que a su vez genere el menor impacto visual posible a la exposición al público de la obra.

Las obras de arte deben ser expuestas al público, pues representan un patrimonio que es de todos y es una pena que se releguen a estar en ambientes ocultos, oscuros sin que las personas puedan disfrutar de ellas. No obstante, son piezas muy delicadas, cuya conservación es de vital importancia para que perduren en el tiempo. Actualmente la tecnología disponible nos permite dar solución a este problema. La solución pasa por instalar el equipo electrónico junto a la obra que se quiera monitorizar. A menudo será en el interior de una vitrina, pero también podrá ser en lugares clave donde se sepa por experiencia que representa un problema. También se podrán instalar varios nodos sensores en una sala para ser capaces de controlar aquellas áreas que tengan cambios de temperatura o humedad más drásticos para evitar instalar objetos más sensibles en ellas.

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

4. Planteamiento de soluciones alternativas y justificación de la solución adoptada

En todo proyecto, y en especial en un proyecto de ingeniería, resulta vital realizar un estudio de investigación con el objetivo de conocer el estado del arte sobre las tecnologías que se van a tratar y tomar la decisión idónea para cada tipo de proyecto. Para el proyecto que nos atañe, se han seleccionado una serie de elementos críticos que conviene comparar entre distintas soluciones para elegir la adecuada.

En primer lugar, se realizará un estudio del estado del arte en cuanto a sensores comerciales se refiere y las ventajas o desventajas que presenta la elección de cada uno, posteriormente se compararán varias tecnologías de comunicación inalámbrica para ver cual es la ideal para esta aplicación. En tercer lugar, se hará un estudio tanto de los microcontroladores existentes en el mercado como de las plataformas disponibles para su programación y el transceptor a usar para implementar la comunicación sin cables. Por ultimo se compararán diversas químicas de baterías tanto primarias como secundarias comparándolas entre si con opciones comerciales con el objetivo de seleccionar la que mejor se adapte.

4.1. Elección de los sensores

Un *sensor* es todo aquello que tiene una propiedad sensible a una magnitud del medio y que, por tanto, al variar esa magnitud también lo hace la propiedad a la que es sensible. Puede entenderse, por tanto, que un sensor es un dispositivo que convierte una forma de energía en otra.

Comúnmente llamamos sensor a lo que se conoce como *transductor*, que es un elemento encargado de transformar una propiedad física o química en una variable eléctrica que posteriormente podremos cuantizar.

Hay una serie de características técnicas que definen a un sensor y que los diferencian entre sí:

- Rango de medida: valores máximos y mínimos entre los que puede medir el sensor
- Precisión: grado de concordancia entre sucesivas medidas
- Exactitud: capacidad del sensor de dar un resultado próximo al valor verdadero de la medida
- Linealidad: tipo de respuesta del sensor ante una perturbación lineal
- Resolución: mínima variación de la magnitud de entrada que puede detectarse a la salida
- Rapidez de respuesta: define la frecuencia máxima a la que podrá ser usado

La forma más general de clasificar los sensores, independientemente del parámetro que estos midan, es por la forma en la que nos devuelven el parámetro medido, es decir, su salida.

Se diferenciará entre:

- **Sensores analógicos:** cuando su salida sea representada en un valor analógico, es decir, un valor físico infinito.
- **Sensores digitales:** por el contrario, nos devuelve una señal digital binaria codificada que representa la magnitud o variable que está midiendo. En el mundo físico no existen señales digitales, la digitalización es una invención humana y, por tanto, un sensor digital ha de transformar o convertir la magnitud física en una señal binaria codificada (nivel alto o bajo). Este proceso se conoce como digitalización.

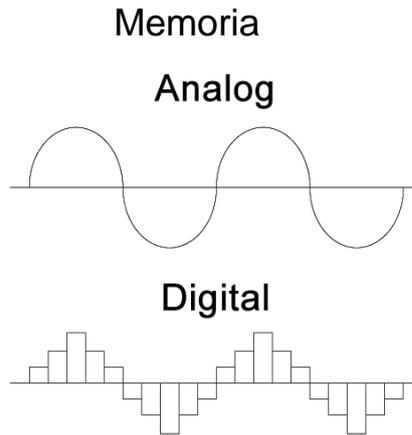


Figura 1 - Dominio analógico vs digital

Lo que se conoce comúnmente como sensor es más bien un sistema de medida más o menos complejo. Un sistema de medida se compone principalmente de 3 elementos: el propio **sensor o transductor**, un **amplificador**, ya que los transductores suelen generar una señal analógica muy débil, un filtro analógico que nos ayudará a atenuar el ruido eléctrico que la propia señal pueda contener, y por último, un Conversor de Analógico a Digital (ADC), que podrá no estar presente si la señal va a ser tratada y procesada mediante un sistema analógico.

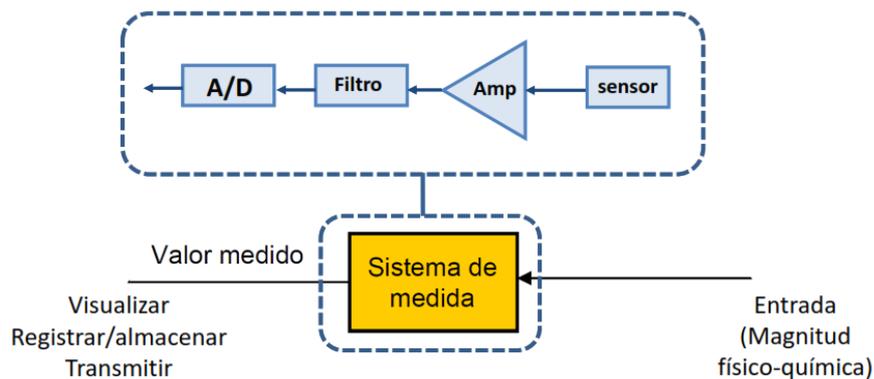


Figura 2- Sistema de medida (bucle abierto)

Un ADC (del inglés *Analog to Digital Converter*) es un circuito electrónico capaz de convertir una variable analógica como un valor de tensión, en una codificación digital capaz de ser procesada por un sistema digital, como un ordenador o un microcontrolador.

El presente proyecto no pretende desarrollar la electrónica del sensor en sí, si no diseñar a nivel de aplicación. Y por ello se seleccionará un sensor comercial tipo circuito integrado para embeber en placa. Estos sensores representan la opción ideal si se pretende medir temperatura ambiental y permite al diseñador centrarse en la aplicación final y no tanto en el desarrollo de la electrónica de instrumentación del sensor, ahorrando tiempo y recursos. Además, resulta difícil competir con un sensor de este tipo, ya que representan una opción muy probada y vienen calibrados de fábrica y listos para usar. Estos sensores utilizan una interfaz digital por lo que la temperatura es transmitida sin riesgos de interferencias.

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

4.1.1. Sensor de temperatura y humedad

En la actualidad podemos encontrar distintas tecnologías para medir temperatura. Todas son válidas, pero algunas son más adecuadas que otras en según qué circunstancias.

A continuación, se comparan distintos sensores digitales en formato circuito integrado. Dichos circuitos a menudo integran un sensor de temperatura y un sensor de humedad, ya que, aunque no dependen un parámetro del otro, es necesario medir la temperatura para compensar la medida de humedad, y por tanto los fabricantes optan por incluir las dos salidas en un mismo integrado.

Fabricante	Sensirion	Texas Instruments	Maxim Integrated
Referencia	SHT31	LM35	DS18B20
Tipo de Sensor	T&HR Digital	Temperatura	Temperatura
Rango de Humedad	0-100%	N/A	N/A
Rango de Temperatura	10-60°C	-55 a 150 °C	0
Tipo de Salida	i2c (2 direcciones)	Analógico lineal	Digital 1 hilo
Exactitud HR	±2% typ	N/A	N/A
Exactitud T	±0.3°C typ	±1 typ	±0.5 max
V Alimentación	2.4 a 5.5V	4 a 30V	0
Consumo activo (uA)	1500	60	1500
Consumo standby (uA)	0,7	N/A	100
Dimensiones	3x3mm	5x5mm	N/A
Precio	5,98	1,72	0

Tabla 1 - Comparativa Sensores Temperatura y Humedad

4.1.1.1.Sensirion SHT31

El sensor SHT31 de Sensirion es uno de los más utilizados en aplicaciones de este tipo debido a su alta resolución y precisión en la medida. Tiene una precisión de ±2% en la medida de humedad y ±0.3 en la de temperatura, lo que se traduce en unas medidas muy fiables para la mayoría de los usos.



Figura 3 - SHT31 montado en placa de evaluación de Adafruit

Memoria

Incluye un filtro PTFE físico opcional para evitar que se ensucie de polvo a la vez que permite que las medidas de humedad funcionen correctamente.

Funciona mediante el bus I2C por lo que resulta muy fácil de implementar. También dispone de una salida de alarma en la que se puede configurar un umbral. Cuando el sensor detecte que se ha salido del rango configurado activará la señal de alarma que puede ser utilizada para generar una interrupción en el microcontrolador.

4.1.1.2. Texas Instruments LM35

A diferencia del sensor digital de Sensirion, el LM35 de Texas Instruments tiene una salida analógica y un rango de precisión entre -55 y 150 grados centígrados. El LM35 es un sensor ampliamente utilizado, pero presenta la desventaja de tener una salida analógica donde la tensión de salida es proporcional a la temperatura. Esto obliga a usar un ADC para convertir los valores a digital.

Por el contrario, también tiene un error mayor de hasta $\pm 1^{\circ}\text{C}$, y no mide la humedad relativa, pero a diferencia del SHT31, su precio es muy bajo y está disponible en formato de agujero pasante.

4.1.1.3. Maxim Integrated DS18B20

Se trata de un sensor muy famoso en la comunidad *Maker DIY* debido a su bajo costo y altas prestaciones, además de su formato cápsula hermética que permite la medición de temperatura en líquidos. Funciona bajo un protocolo de comunicación serie asíncrono (sin señal de datos) que tan solo precisa de un cable de datos. Tiene una precisión de ± 0.5 grados centígrados y su precio ronda los dos euros. Tampoco dispone de medida de humedad.



Figura 4 - Sonda sensor DS18B20

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

4.1.2. Sensor de luminosidad

4.1.2.1. Fotorresistencia LDR

Se trata de posiblemente la solución más sencilla. El uso de una resistencia dependiente de la luz permite disponer de un valor analógico de tensión proporcional a la intensidad lumínica incidente. No obstante, se trata de una resistencia variable que no está calibrada y, por tanto, ha de ser calibrada a nivel de aplicación. Además, al tratarse de una resistencia, se la debe alimentar para obtener un valor de voltaje y por tanto disipa una potencia en funcionamiento que no es despreciable.

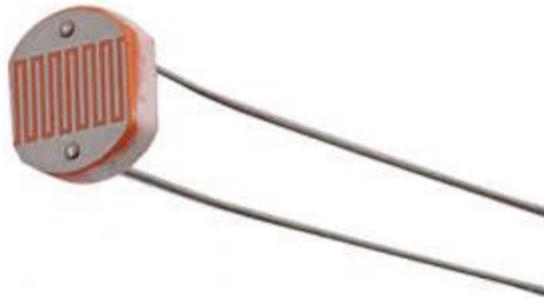


Figura 5 - Fotorresistencia LDR

4.1.2.2. AMS TSL2591

Los sensores de luminosidad hacen uso de fotodiodos para medir la intensidad lumínica incidente. En el caso del **TSL2591** de AMS Semiconductors, que contiene dos fotodiodos, están conectados a dos ADC que integra las corrientes de los dos fotodiodos simultáneamente. Los resultados de la conversión son transferidos a un registro digital desde donde pueden ser leídos por I2C. En cuanto los valores son leídos de los registros, el dispositivo empieza otra conversión

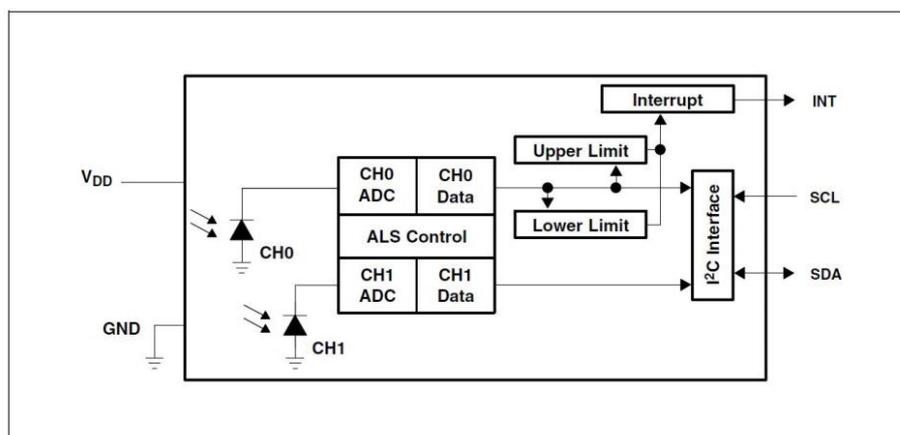


Figura 6 - Diagrama de bloques del sensor TSL2591

El uso de sensores que nos den el valor a medir de forma digital simplifica enormemente la conexión de estos con un microcontrolador al no ser necesario el tratamiento analógico de la señal, ni ningún circuito externo, además de que la señal resultante es inmune a ruido al tratarse de una señal digital.

Memoria

Este sensor dispone de un comparador que genera una interrupción cuando el valor de luminosidad detectado es mayor o menor a un umbral que puede ser configurado por el usuario mediante la escritura del registro I2C correspondiente.

4.1.2.3. Silicon Labs Si1133

El sensor de Silicon Labs resulta una opción muy interesante, debido a que, además de medir la luminosidad de luz visible incidida, también nos permite medir el índice de luz ultravioleta con gran exactitud. Este parámetro resulta muy útil para la monitorización de obras de arte, ya que estas se ven muy afectadas por la componente ultravioleta de la luz, especialmente solar, que resulta muy dañina.

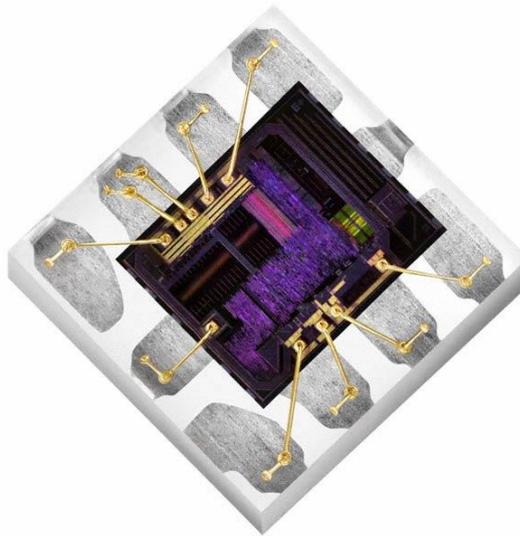


Figura 7 - Sensor Si1133

Este sensor fue considerado en un principio y se llegó a montar en la placa desarrollada. No obstante, los registros I2C resultaban complicados de utilizar, por lo que se desestimó su uso. Además, resultaba complicado de montar de forma manual en el prototipo debido a su reducido tamaño. El sensor está soportado por la placa electrónica para futuras actualizaciones.

4.1.2.4. Rhom Semiconductors BH1750FV

Ampliamente utilizado en la comunidad *Maker*, el BH1750FV es un sensor de luminosidad ambiente de bajo costo con una interfaz digital I2C. Tiene un rango de luminosidad muy alto, pero no permite un ajuste de la ganancia interna y, aunque su uso resulta muy sencillo, no ofrece el nivel de configuración y versatilidad deseado.

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

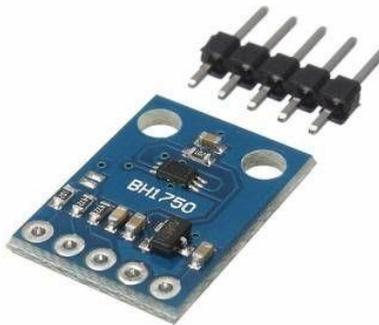


Figura 8 - Módulo BH1750FV

4.2. Elección de la tecnología inalámbrica

La elección de la tecnología inalámbrica a usar para la transmisión de datos del sensor a la aplicación final es una decisión crítica en el desarrollo de este proyecto, ya que de ella dependerá la versatilidad del sensor y permitirá que pueda usarse en exteriores o solo en interiores controlados.

Las opciones alámbricas se han descartado totalmente y ni si quiera serán analizadas en esta memoria por considerarse que no cumplen con la filosofía de versatilidad que se le exige que tenga, según se explica en apartados anteriores. Además, a menudo resulta difícil el llevar cableado a zonas que no han sido preparadas para ello con el correspondiente impacto estético que conlleva el conducir dichos cables por canaletas o similares. No obstante, la tecnología inalámbrica ofrece una serie de ventajas que no cumple la inalámbrica, como puede ser la confianza o confiabilidad de la conexión, la velocidad o la seguridad.

Wired vs. Wireless [Consumer / Personal use applications]		
	Wired	Wireless
Convenience	☆☆	☆☆☆☆☆
Reliability	☆☆☆☆	☆☆☆
Speed	☆☆☆☆	☆☆☆☆
Security	☆☆☆	☆☆☆
Ease of Setup	☆☆☆☆	☆☆☆

Figura 9 - Comparativa tecnología alámbrica e inalámbrica (studiopsis.com)

Teniendo en cuenta todo ello, se ha seleccionado un amplio abanico de tecnologías inalámbricas actuales que se estudian a continuación, con la intención de elegir la que mejor se ajuste al proyecto.

Memoria

4.2.1. WiFi

El WiFi es una tecnología inalámbrica que permite la interconexión de dispositivos electrónicos habilitados para ello, tales como ordenadores, teléfonos inteligentes, televisores, videoconsolas, reproductores de música, etc. Esta tecnología define un estándar que permite la conexión de estos dispositivos entre sí y/o la conexión de estos a la red de Internet, todo ello de manera inalámbrica.



Figura 10 - Logo certificación WiFi

El WiFi surge en 1999 por la necesidad de establecer un mecanismo de conexión inalámbrica compatible entre distintos dispositivos, ya fueran ordenadores personales, asistentes digitales personales (PDA) u otros. Varias empresas entre las que destacan Nokia o 3Com se unieron para crear lo que actualmente se conoce como Alianza WiFi, un colectivo de empresas que estandarizan y regulan el uso de esta tecnología inalámbrica, asegurando que todos los equipos con el sello WiFi puedan trabajar juntos sin problemas, independientemente del tipo de dispositivo o fabricante.

El IEEE 802.11 [3] es la norma que define y regula tanto la capa física como la capa MAC (*Medium Access Control*) del modelo OSI para la tecnología WiFi. El modelo OSI (*Open System Interconnection*) o modelo de interconexión de sistemas abiertos es un modelo de referencia para los protocolos de la red de arquitectura en capas, este modelo se compone de 7 niveles o capas que utiliza para definir una red de interconexión. La tecnología inalámbrica WiFi es exactamente igual en lo que respecta al modelo OSI que la tecnología LAN tradicional, a excepción de las capas 1 y 2 (MAC y PHY), que son las definidas en la IEEE 802.11. Esto en la práctica nos dice que lo único que diferencia una red WiFi de una red Ethernet tradicional por cable es como se transmiten las tramas o paquetes de datos y su codificación, el resto es totalmente idéntico, lo que garantiza la compatibilidad de todos los servicios de redes locales entre sí.



Figura 11 - Modelo OSI WiFi (monografias.com)

Existen diversos tipos de WiFi basados en el mismo estándar, el clásico que trabaja en una frecuencia de 2.4GHz y permite transmitir datos a una velocidad de hasta 300Mbit/s y el más actual conocido como WiFi 5G o simplemente como WiFi 5 por operar en la banda de frecuencias de los

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

5GHz. Este nuevo estándar de capa física permite una mayor tasa de transmisión y un mayor ancho de banda al operar en una frecuencia dos veces mayor y utilizar canales relativamente limpios y sin interferencias al no existir otras tecnologías como sí ocurre con la banda de 2.4GHz (Bluetooth, Zigbee, WUSB...). Por el contrario, este nuevo estándar tiene un menor alcance que el que trabaja a 2.4GHz debido a que la frecuencia es mayor.

El WiFi permite un ancho de banda muy grande, pero a costa de un gran gasto de energía para transmisión y demodulación de los datos. Esto lo hace poco ideal para dispositivos alimentados a baterías ya que obliga a incluir baterías muy grandes (del orden de los 2000mAh como las que se utilizan en los teléfonos inteligentes de última generación) para autonomías que no sobrepasan el día de uso. Además, al operar en la banda de 2.4GHz (UHF o *Ultra High Frequency*), los dispositivos que hacen uso de la tecnología WiFi tienen un alcance de no más de 100-200 metros en el mejor de los casos debido a la propia física de estas longitudes de ondas. Esto a menudo obliga a instalar repetidores de señal para ampliar el rango.

Todo esto, hace que se descarte esta tecnología inalámbrica para este proyecto, al tratarse de un dispositivo autónomo alimentado de baterías que, a menudo, se instala en zonas de difícil acceso, lo cual obliga a que las baterías deban durar un tiempo superior a un año.

4.2.2. Bluetooth Low Energy

Para solucionar el problema de alto consumo del WiFi se considera el BLE (Bluetooth Low Energy) o también conocido como *Bluetooth Smart*. BLE es una versión ligera del Bluetooth clásico tradicional y fue introducido como parte del Bluetooth 4.0. Aunque comparte muchas cosas en común con el Bluetooth tradicional, BLE tiene una arquitectura completamente diferente.



Figura 12 - Bluetooth Smart Logo

Este estándar Bluetooth es usado en los conocidos como *wearables* o llevables en español, ya que permite una interconexión con un dispositivo central (normalmente un teléfono inteligente con mayor capacidad de conexión), con un consumo muy bajo de energía que permite reducir el tamaño de estos dispositivos al no ser necesarias grandes baterías. Todo ello con una duración de baterías muy aceptable que a menudo ronda el mes de uso.

BLE entra dentro de la clasificación de red de área personal inalámbrica o *personal area network (PAN)* en inglés y sus ventajas respecto al Bluetooth tradicional pasan por una reducción de consumo energético y coste manteniendo un rango de comunicación similar. Esta tecnología puede convivir con el Bluetooth tradicional en el mismo dispositivo sin necesidad de una antena extra, ya que opera en el mismo espectro de frecuencia de 2.400–2.4835 GHz.

Según se extrae de [4], BLE tiene una arquitectura basada en el *Generic Access Profile (GAP)*, que controla las conexiones del Bluetooth y determina que dispositivo pueden o no interactuar con otro. El estándar define dos tipos de dispositivos o roles que cada dispositivo puede adoptar:

Memoria

- **Peripheral** o Servidor (GATT Server) es aquel dispositivo que transmite datos a otro que actúa como central de procesamiento. Un ejemplo de un dispositivo de tipo *peripheral* sería un pulsometro, una etiqueta de proximidad, etc.
- **Central**, entendido como cliente (GATT Client), es aquel que recibe los datos que envía el servidor. Normalmente tiene mucha más potencia de procesado y memoria. Un ejemplo puede ser un teléfono móvil inteligente o una tableta.

El dispositivo *peripheral* transmite un dato en un intervalo de tiempo definido (cuanto más corto sea este periodo, más energía gastará) para anunciar que está disponible para la conexión. El dispositivo master o cliente también puede enviar un *scan response request* para que todos los dispositivos periféricos que estén dentro de cobertura respondan con un *scan response data*.

Una vez realizada la conexión se realizan las transacciones GATT, donde todas las conexiones son iniciadas por el dispositivo master (*GATT Client*) para ser contestadas con los datos correspondientes por el dispositivo esclavo (el sensor).

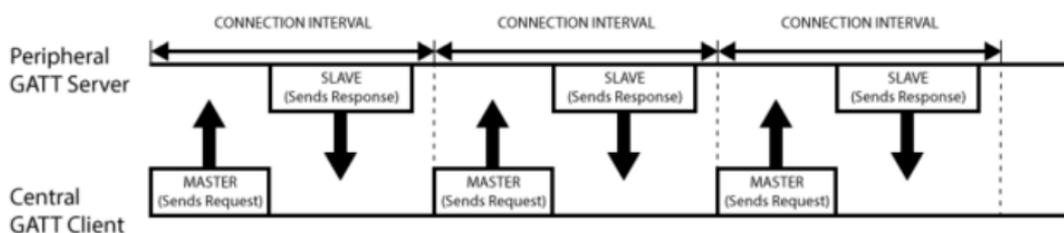


Figura 13 - Transacciones GATT

No es el objeto de esta memoria explicar en profundidad como funciona este estándar de transmisión, por lo que nos quedaremos con su topología y con las razones por las cuales ha sido desestimado su uso en el proyecto.

Este protocolo de transmisión inalámbrica está pensado para dispositivos sencillos no inteligentes (sensores que tan solo transmiten datos y no realizan un gran procesado de los mismos) que han de estar siempre conectados a un dispositivo master o cliente que es el que hace la recolección y procesado de los datos. Este dispositivo *master* también será el que tenga más posibilidades de conectividad para subir los datos a la red de Internet si fuera necesario. Nos quedaremos por tanto con la idea de que para hacer uso de BLE es necesario disponer de un dispositivo auxiliar en el radio de cobertura de los sensores, que ronda los 100 metros en campo abierto en el mejor de los casos. Por la naturaleza del proyecto, no podemos asegurar que los sensores vayan a estar separados entre sí menos de 100 metros, ya que, de ser así, esto implicaría, por ejemplo, el instalar un dispositivo master en cada sala de un museo para monitorizar todas las obras de arte que en ella se encuentran.

4.2.3. ZigBee

ZigBee es como se conoce a la especificación de un conjunto de protocolos de red de malla de bajo consumo que al igual que el Bluetooth, está basada en el estándar **IEEE 802.15.4** [5] de redes inalámbricas de área personal (*Wireless personal area network*, WPAN). Su objetivo son las aplicaciones que requieren de comunicaciones seguras con baja tasa de envío de datos y una maximización de la vida útil de sus baterías, ya que ZigBee ha sido diseñado para proporcionar un alto rendimiento en aplicaciones con una tasa de transmisión baja, donde el consumo de energía es una consideración importante.

Definida como “una tecnología inalámbrica de corto alcance y bajo consumo” que se define como “una solución inalámbrica de baja capacidad para aplicaciones en el hogar como la seguridad

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

y la automatización” [6]. Por tanto, su objetivo no es obtener velocidades muy altas ya que tan solo ofrece una velocidad de transmisión de hasta 250Kbps con un rango de hasta 75 metros, si no la de crear sensores y actuadores, en principio destinados al hogar, alimentados con pilas y con un consumo de energía muy bajo.

Opera en las bandas de frecuencia de radio sin licencia, conocidas como bandas libres. Estas son, entre otras, las bandas de 2.4GHz (también utilizada para WiFi y Bluetooth), pero también en las de 868MHz (Europa) y 915MHz (Estados Unidos). Aunque lo normal es que usen la banda de 2.4GHz, para conseguir una cobertura de uso mundial.

Zigbee es un protocolo muy enfocado a la domótica, que nació prácticamente para ello. Las soluciones comerciales que hacen uso de la red ZigBee suelen ser productos para el hogar como bombillas LED inteligentes con temperatura de color ajustable, mandos remotos para controlarlas, control de brillo para paredes, sensores de gas, de temperatura, de movimiento, de apertura de ventanas y puertas, entre otros.

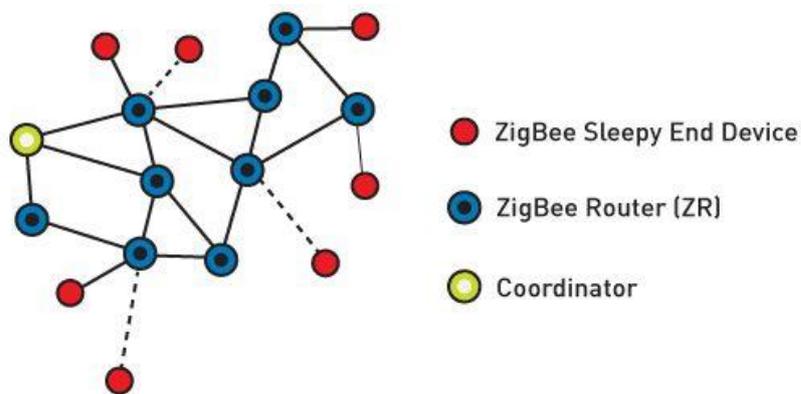


Figura 14 - Topología Malla Zigbee

ZigBee funciona bajo una tipología de malla, aunque se pueden utilizar otras topologías como estrella u árbol. ZigBee define 3 tipos de dispositivos distintos:

- **Dispositivo final ZigBee (end device):** Es aquel que actúa como dispositivo de aplicación final e implementa los sensores, actuadores u etiquetas inteligentes. Dicho dispositivo está dormido la mayor parte del tiempo con un consumo de energía cercano a cero, y se despierta para transmitir datos o escuchar periódicamente. No realiza tareas de enrutamiento.
- **Router ZigBee:** Dispositivo donde van conectados los *end-devices*, es el encargado de enrutar las conexiones y datos al dispositivo final que se conecta. Un router puede ser conectado a otro router y/o *end-devices* (hasta 255 dispositivos por router)
- **Coordinador:** Toda red ZigBee precisa de un coordinador o master, que se encargará de manejar, procesar y controlar las conexiones de los demás dispositivos. Al coordinador solamente irán conectados uno o más *routers*, pero nunca un *end-device* y solo puede haber una figura de coordinador por red.

De esta manera, ZigBee permite crear redes de malla muy versátiles y escalables para cualquier situación. El ejemplo clásico de ZigBee en domótica de hogar pasa por tener un dispositivo coordinador central y un enrutador en cada habitación de la casa con varios sensores u actuadores conectados al mismo.

Memoria

No obstante, como ha podido quedar representado, se trata de un protocolo de red bastante complejo, que implica tener una red muy definida y controlada. Esto lo hace una tecnología muy versátil, pero por otro lado complica sustancialmente la creación y mantenimiento de la red. A todo esto, se le suma el corto alcance que tiene, al operar normalmente en la red de 2.4GHz y tratarse de una red WPAN. Pensando en la aplicación que nos atañe, la parte de un consumo eficiente y reducido de energía optimizado para dispositivos alimentados por baterías la tendríamos resuelta, pero no la parte de largo alcance (del orden de varios kilómetros) que nos vemos obligados a resolver si se requiere de una monitorización de obras al exterior. Es por ello por lo que tampoco se considerará la tecnología ZigBee.

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

4.2.4. RFID

El RFID o *Radio Frequency IDentification*, es una tecnología de transmisión de datos inalámbrica ampliamente utilizada para la identificación tanto de personas como de cosas.



Figura 15 - Ejemplo de etiqueta RFID activa

Existen principalmente dos tipos de identificación por radiofrecuencia, las etiquetas pasivas, que carecen de alimentación propia y son alimentadas por el lector, y las etiquetas activas, las cuales emiten una señal de radio frecuencia pudiendo comunicarse a una distancia mucho mayor y realizar una mayor cantidad de procesamiento de datos.

Los sistemas RFID activos operan típicamente en la banda de alta frecuencia (UHF) y ofrecen coberturas de hasta 100 metros. Se emplean generalmente en objetos grandes que requieran de un rastreo a largas distancias como paquetes de mensajería voluminosos o importantes. Para que la comunicación tenga lugar, las balizas o etiquetas activas, que emiten cada pocos segundos o pocas veces al día, deben encontrarse dentro del rango de cobertura de una antena lectora de la señal RFID. Normalmente es posible tanto identificar la baliza como posicionarla con el objetivo de conocer su localización. No obstante, los sistemas de RFID están habitualmente enfocados a localización e identificación y no tanto a la lectura de sensores.

4.2.5. Telefonía móvil GSM/GPRS/3G/LTE

La telefonía móvil [7] es conocida ya por todos, ya que se remonta a finales de la década de 1950, cuando el científico soviético Leonid Ivanovich Kupriyanovich, desarrolló un sistema de comunicación móvil utilizando ondas de radio y fue capaz de alcanzar una distancia de hasta 30km. El sistema fue desarrollado por múltiples compañías, entre ellas la famosa Motorola [8], comenzando con una modulación analógica conocida como **NMT** (*Nordic Mobile Telephone*) entre 1980 y 1995, para posteriormente ser desarrollados los sistemas de segunda generación o sistemas digitales **GSM** (*Global System Mobile Communication*, 1992), seguidas por los sistemas de tercera generación **UMTS** y los modernos sistemas de cuarta generación, conocidos como 4G o LTE de *Long Term Evolution-Advance*.

Memoria

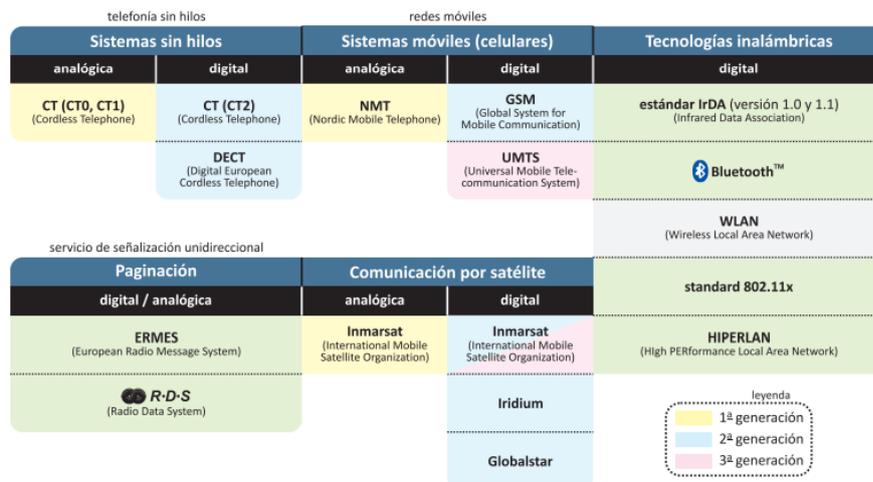


Figura 16 - Descripción general de las tecnologías empleadas en comunicación móvil

Las redes móviles operan en frecuencias de microondas (del orden de GHz) conocidas como UHF, *Ultra High Frequencies*. El espectro de frecuencias utilizadas cada vez es mayor, debido al aumento de banda ancha por la creciente demanda de contenido multimedia (audio y video). Los primeros sistemas analógicos tan solo permitían la transmisión inalámbrica de voz, más tarde, con la llegada del sistema digital, las bandas de radio también permitían la codificación y transmisión de datos, pero a velocidades bajas. Por último, con la aparición de las nuevas generaciones de telecomunicación móvil, las bandas son capaces de transmitir contenido multimedia a grandes velocidades.

Dichas redes de telecomunicación modernas precisan de una infraestructura muy grande, costosa y difícil de mantener. Es por ello por lo que dichas redes y frecuencias pertenecen a grandes empresas de telecomunicación a menudo reguladas por los gobiernos. En España las infraestructuras pertenecen a Telefónica, Vodafone y Orange, principalmente. Estas empresas son las encargadas de crear, mantener y actualizar una infraestructura de comunicación inalámbrica móvil para cubrir una amplia cobertura que se logra a través de repetidores o estaciones base colocadas en formación de celdas o *clusters*.

Para hacer uso de la tecnología móvil, ya sea GSM o redes más modernas, es necesario contar con un SIM (*Subscriber Identity Module*), conocido normalmente en su factor de tarjeta. Esta tarjeta contiene los datos básicos de identificación necesarios para la conexión del dispositivo a la red móvil y otros datos como claves de autenticación en la red. El terminal o dispositivo móvil solo podrá ser utilizado con la activación de la tarjeta SIM en el operador específico.

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural



Figura 17 - Módulo GSM con slot para tarjeta SIM

El sistema de transmisión inalámbrica móvil obliga al dispositivo a estar permanentemente en busca de cobertura de red, y a estar identificado en todo momento. Esto representa un gran gasto de energía, también en la transmisión inalámbrica de datos.

El gran gasto de energía que representa el uso de esta tecnología para un dispositivo que se requiere que tenga una autonomía de años, unido a la poca flexibilidad que nos ofrece una red con una infraestructura propietaria de pago, nos llevan a descartar la telefonía móvil inalámbrica para el desarrollo de este proyecto. No obstante, sigue siendo una tecnología válida como centralita o puerta de enlace con otra tecnología inalámbrica, especialmente en situaciones donde no se disponga de una conexión a Internet a través de WiFi o cable. Esto es algo que se verá más adelante, cuando se expliquen las redes LPWAN.

4.2.6. LPWAN

El desarrollo del Internet de las Cosas en esta última década ha promovido el surgimiento de nuevas tecnologías de comunicación inalámbrica pensadas y optimizadas para estos nuevos estándares de comunicación. Dichas tecnologías han surgido como una necesidad de cubrir las carencias que presentan las tecnologías tradicionales como el WiFi o las redes móviles, que han sido estudiadas con anterioridad. Estos nuevos estándares de redes ofrecen una mejora en las capacidades para implementar y desarrollar proyectos orientados a usar una arquitectura IoT, con unas características idóneas para ello [9].

Las redes LPWAN o *Low Power Wide Area Networks* son aquellas orientadas a dispositivos inalámbricos de bajo consumo con un gran alcance, pero con un ancho de banda muy bajo. Es una tecnología de transmisión inalámbrica creada para aplicaciones *machine-to-machine* (M2M) y muy enfocada al Internet de las cosas (IoT) y, por tanto, al estar concebidas para este entorno IoT, las tecnologías LPWAN ofrecen unas prestaciones muy interesantes que pretenden introducirse como nuevos estándares de comunicación inalámbrica entre máquinas.

Generalmente pueden enviar y recibir paquetes de datos de 10 a 1000 bytes a unas velocidades de 200Kbps y unos rangos que varían entre 2 y 1000 kilómetros [10], aunque esto varía dependiendo de la tecnología específica, pero la filosofía siempre es la misma. La mayoría de las redes de este tipo tienen una tipología de estrella similar al WiFi donde cada dispositivo o *endpoint* se conecta directamente a un punto central de acceso llamado pasarela o *gateway*. Por norma

Memoria

general entenderemos que las redes LPWAN no ofrecen una conectividad IP punto a punto. Esto significa que, aunque el destino final de los paquetes de datos sea una red basada en un protocolo TCP/IP como lo es la red de Internet, el dispositivo sensor que envía los paquetes normalmente no hará uso de una conexión directa TCP/IP y, por tanto, el paquete tendrá que ser decodificado y convertido mediante el uso de una pasarela. Esto se verá más adelante con profundidad cuando se explique el funcionamiento de LoRaWAN.

A menudo se tiende a comparar a las tecnologías LPWAN con las tecnologías conocidas como LPLAN (*Low Power Local Area Network*). Estas son principalmente el Bluetooth o Zigbee y se asemejan a las LPWAN en lo que respecta al bajo consumo, pero la gran diferencia reside en el alcance. Como puede verse en la Figura 18 y Figura 19 la topología de red es prácticamente la misma y también su implementación, pero las LPLAN tan solo tiene una cobertura física de pocas decenas de metros, mientras que en las LPWAN estaríamos hablando de coberturas de varios kilómetros.

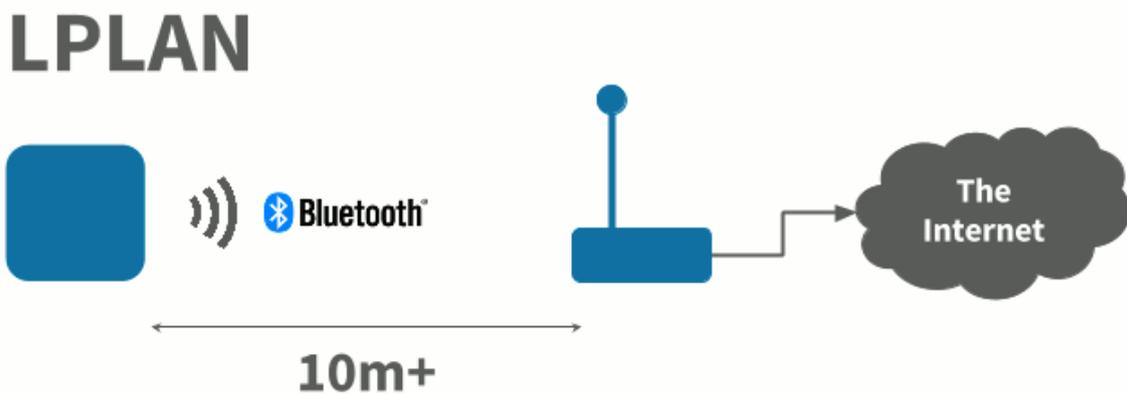


Figura 18 - Diagrama cobertura Bluetooth

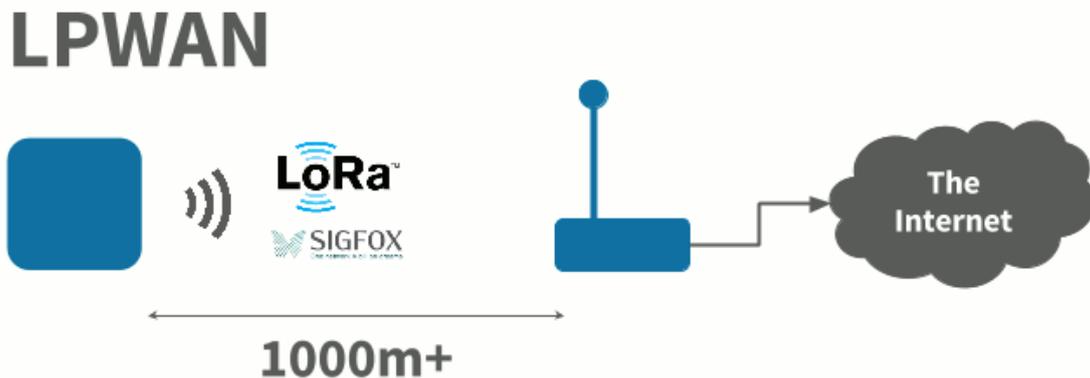


Figura 19 - Diagrama cobertura LPWAN

El aumento de cobertura respetando el bajo consumo solo puede conseguirse a un precio, y este es el de disminuir el ancho de banda disponible y por tanto la cantidad de datos que puede ser transmitida, esto puede verse en el gráfico comparativo de la *Figura 20* que relaciona el ancho de banda disponible respecto al alcance o cobertura del que se beneficia. Observamos que las tecnologías LPWAN disfrutan de un rango de cobertura por encima de tecnologías como WiFi, Bluetooth BLE, ZigBee o RFID, como ya se ha visto en apartados anteriores, pero por el contrario disponemos de un ancho de banda muy limitado con velocidades de transmisión muy lentas.

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

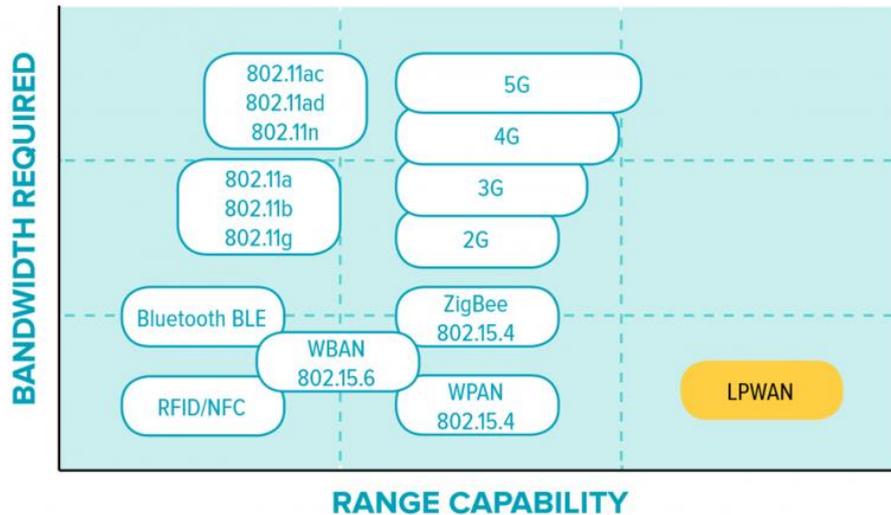


Figura 20 - Comparación tecnologías inalámbricas (IoTForAll.com)

No podremos, por tanto, transmitir contenido multimedia en alta calidad como si podemos hacer usando WiFi o redes móviles modernas 4G/LTE, ni audio de alta fidelidad como si nos permite Bluetooth, pero será la tecnología ideal si tan solo precisamos de transmitir unos pocos bytes de información por hora con un radio de cobertura muy alto y esto es precisamente lo que nos interesa para la aplicación que nos atañe.

Teniendo en cuenta los parámetros que se desean medir: temperatura, humedad relativa del aire y luminosidad incidente, se estima que son suficientes 10-20 bytes para representar dichas variables. Además, sabiendo que son variables de entorno que varían muy despacio podemos fijar un intervalo de envío relativamente alto (unos 15-30 minutos). Es por ello por lo que se hará uso de una de una transmisión inalámbrica LPWAN para la aplicación que se describe en el proyecto.

	LoRa	sigfox	NB-IoT	LTE	INGENU	WEIGHTLESS	LinkLabs
	LoRa / LoRaWAN	Sigfox	NB-IoT	LTE-M	RPMA	Weightless-P	Symphony Link
Origin	France	France	USA (Global)	USA (Global)	USA	UK	USA
Proprietary or open	LoRa – proprietary LoRaWAN – open	Net – proprietary Devices – open	Open	Open	Proprietary	Open	Proprietary
Cellular	No	No	Yes	Yes	No	No	No
Spectrum	Unlicensed	Unlicensed	Licensed	Licensed	Unlicensed	Unlicensed	Unlicensed
Range, km	urban: 2-5 rural: 15	urban: 3-10 rural: 30-50	urban: 1-5 rural: 10-15	urban: 2-5	urban: 1-3 rural: 25-50	urban: 2	urban: 2-5 rural: 15
Speed, uplink / downlink	50 kbps / 50 kbps	300 bps / –	250 kbps / 250 kbps	1 Mbps / 1 Mbps	634 kbps / 156 kbps	100 kbps / 100 kbps	100 kbps / 100 kbps
Power consumption	●●●	●	●	●●●	●●	●	●●
Security	●●	●●	●●●	●●●	●●●	●●●	●●●
Availability of devices	●●	●●●	●●	●	●●	●	●●
Price*	●●	●	●●	●●●	●●●	●	●●
Areas of application	Precision farming, manufacturing automation, pipeline monitoring	Predictive maintenance, capacity planning, demand forecasting	Electric metering, manufacturing automation, retail PoS	tracking objects, wearables, energy management, utility metering, city infrastructure	Digital oilfield, connected cities, usage-based insurance, agriculture	Smart grid, healthcare, automotive, smart cities, asset tracking	Industrial control systems, lighting control, alarm systems
Supporting companies	IBM, Semtech, Cisco, HP, Orange, Kerlink, Actility	STMicroelectronic, Texas Instruments, Atmel, Silicon Labs	Huawei, Ericsson, Qualcomm, Vodafone	Verizon, AT&T, Nokia	Ingenu	Accenture, Sony Europe, uniik, ARM, Telensa	Link Labs



Tabla 2 - Comparación tecnologías LPWAN (Flespi.com)

Memoria

Son varias las tecnologías LPWAN que existen y al tratarse de una tecnología muy reciente, es previsible que en un futuro existan y convivan muchas más. A continuación, realizaremos una comparación de las principales tecnologías de transmisión inalámbrica *long-range* existentes en el momento de la redacción de esta memoria, que se ven resumidas en la comparación de la *Tabla 2*

4.2.6.1. LoRaWAN

LoRa es una modulación inalámbrica apta para aplicaciones long-range low power low-data rate desarrollada por Semtech.



Figura 21 - Logo LoRa

LoRaWAN viene de LoRa Wide Area Network, y se refiere a una especificación para redes de baja potencia y área amplia, LPWAN (Low Power Wide Area Network). LoRaWAN es un protocolo optimizado para dispositivos alimentados por baterías y está concebido para dispositivos que requieran conexión a Internet of Things.

La LoRa Alliance recoge las especificaciones del protocolo de red de LoRaWAN en un documento llamado *LoRaWAN Specification* [11] que define los detalles y limitaciones del protocolo, así como la manera de implementarlo y trabajar con él. Dicho documento define los modelos de capas 1, 2 y 3 del modelo OSI. La primera capa, PHY, correspondiente a la implementación física LoRa, contiene los parámetros físicos tales como frecuencia de transmisión, tipo de modulación propietaria de Semtech, etc... Mientras que las capas 2 y 3, capas MAC y NET definen el direccionamiento físico y la distribución ordenada de las tramas y flujo de red, y el enrutamiento de los paquetes dentro de la red.

Se trata de un protocolo de red libre, por lo que su implementación resulta muy flexible. El usuario puede definir su propio protocolo de red basado en LoRaWAN para hacer uso de la tecnología de modulación LoRa, pudiendo así, crear una red privada de una manera muy sencilla.

Tiene una tipología de red en estrella (*Figura 48*) donde los nodos de aplicación o sensores se conectan a una o varias pasarelas o *gateways* haciendo uso de la modulación de radio frecuencia LoRa, a su vez, las pasarelas, están conectadas al servidor de red LoRaWAN mediante TCP/IP convencional. La red es quien elige de que pasarela coger el paquete y cual descartar.

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

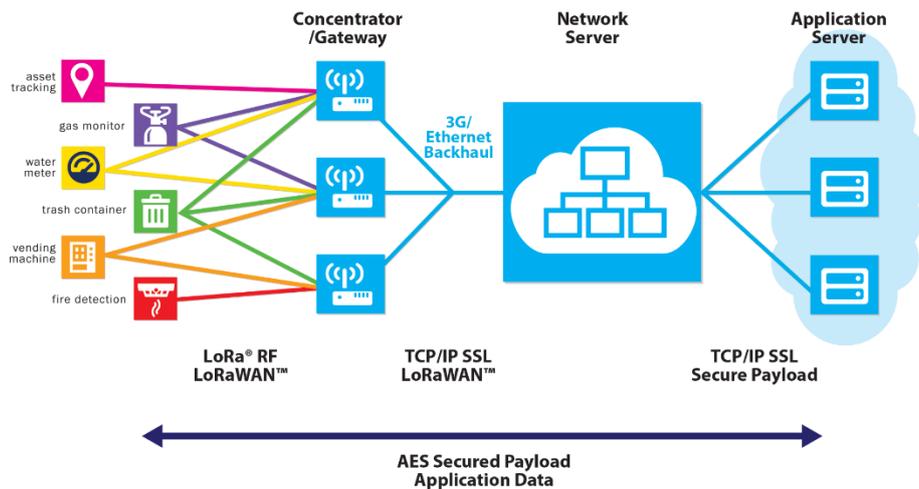


Figura 22 - Tipología de red LoRaWAN

A menudo, para entender las tipologías de red en estrella, se hace la comparación con una clase donde la pasarela (profesor) habla para los dispositivos sensores (los alumnos en la clase) y viceversa. Se trata de una comunicación asimétrica, en tanto que todos los alumnos de la clase podrían estar intentando comunicarse con el profesor al mismo tiempo, pero el profesor no sería capaz de escuchar a todos al mismo tiempo. Es por ello, por lo que LoRaWAN, y todas las tecnologías LPWAN en general, escuchan en varios canales o bandas al mismo tiempo.

4.2.6.2. Sigfox

Sigfox es una de las tecnologías de comunicación más importantes actualmente en el sector IoT junto con LoRaWAN, pero a diferencia de esta, se trata de una tecnología propietaria de la empresa con el mismo nombre. Está basada en una modulación DBPSK (modulación por desplazamiento de fase) para la subida de paquetes hacia el servidor, y en una modulación GFSK (modulación por desplazamiento de frecuencia) para los downlink o envíos de paquetes desde la plataforma hacia los dispositivos. Con esto, Sigfox funciona muy bien transmitiendo datos desde sensores hacia la plataforma, pero no es eficaz al revés. Se trata por tanto de una tecnología claramente enfocada a sensores que solo requieran de envíos de datos hacia la nube y no requieran de “gran” ancho de banda para recepción. No obstante, esto no quiere decir que no lo permita.



Figura 23 - Logo Sigfox

Trabaja transmitiendo sobre las bandas sub-GHz de licencia libre (868MHz para Europa y 900MHz para EE. UU.), al igual que LoRaWAN y hace uso de la comunicación *Ultra Narrow Band*.

Al ser una tecnología propietaria, Sigfox provee de la infraestructura de red para utilizar la tecnología. Esto engloba tanto las infraestructuras físicas como antenas, pasarelas, routers y repetidores, como los servidores para la propia red y la plataforma de gestión en la nube y por tanto, dispone de una tarifa de suscripción para los equipos o aplicaciones que requieran de conexión a su red, que en el momento de la redacción de esta memoria está en torno a 1\$ por suscripción

Memoria

anual [12]. Se trata de un modelo similar al que puedan tener las compañías de telefonía móvil, aplicando una tarifa mensual a sus clientes por conectarse y usar su red.

Sigfox limita tanto el tamaño de los paquetes de información que puedes transmitir, como el número de paquetes que te permite enviar por día por normativa, ya que al trabajar en un espectro de frecuencia sin licenciar, la ley establece un máximo de uso de red. En cuanto al tamaño del paquete, Sigfox lo limita a 12 bytes de información, que junto con 12 bytes más correspondientes a un *header* e identificador de dispositivo y otros metadatos más, componen un total de 25 bytes por mensaje u envío. El límite de envíos diarios está fijado en 140 mensajes tardando una media de 2 segundos en estar disponibles en la plataforma, es decir, que la velocidad de envío está fijada en unos 100 bps para envíos, y bastante más lentas para la subida de información.

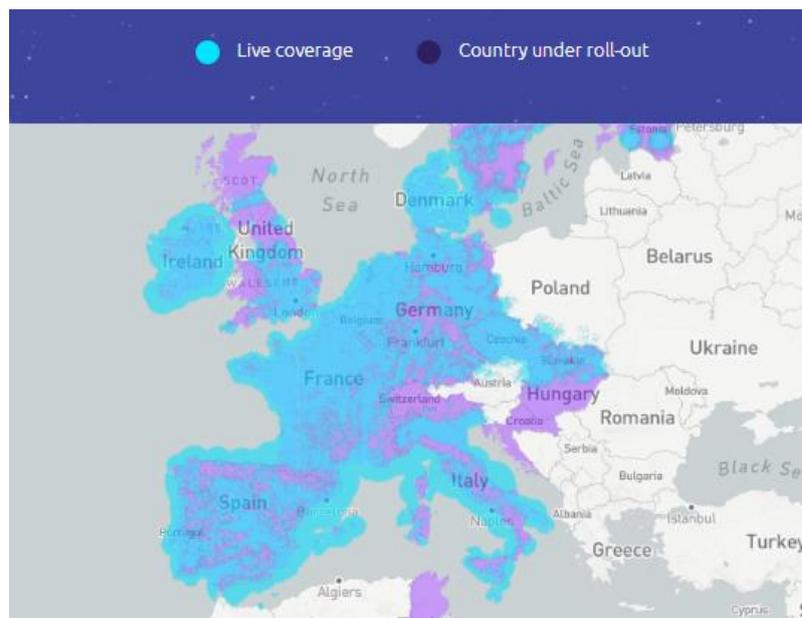


Figura 24 - Cobertura Sigfox Europa

El alcance teórico varía entre los 30 y 50 km en entornos rurales y hasta los 10 km en entornos urbanos [9] al repetidor más cercano. No obstante, la cobertura de Sigfox pretende ser global y en su página web está disponible un mapa de cobertura. [13] (Figura 24)

4.2.6.3. DASH7

La tecnología de comunicación *DASH7 Alliance Protocol (D7AP)* es un protocolo de red LPWAN que proporciona conectividad de alcance medio para actuadores y sensores de baja potencia. DASH7, al igual que LoRaWAN es *open source* y opera en las bandas ISM libres por debajo del Gigahercio.

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural



Figura 25 - DASH7 Alliance logo

Utiliza una modulación de banda estrecha utilizando GFSK (modulación por desplazamiento de frecuencias) y promete obtener una menor latencia que el resto de las tecnologías debido a su ciclo de funcionamiento, que se basa en comprobar periódicamente el canal a la espera de información desde el servidor. Esto la hace ideal para su uso junto a actuadores, ya que permite que las órdenes lleguen rápidamente desde el servidor de procesamiento en la nube al actuador, característica que con otras tecnologías no se da. Por ejemplo, en LoRaWAN, el servidor ha de esperar a que el dispositivo envíe información a la plataforma, para transmitir un mensaje de *downlink*, ya que el nodo solo abre una ventana de escucha cada vez que envía (En dispositivos clase A, ver especificaciones LoRaWAN para más información) [11]. En contraposición, dicha escucha permanente presenta un aumento en el consumo de energía, cuestión de vital importancia en la aplicación que se nos presenta.

D7AP promete un alcance teórico de hasta 5 kilómetros en entornos urbanos con unas tasas de transferencia de decenas de kbps. [9]

4.2.6.4.NB-IoT

Narrowband IoT es una tecnología LPWAN diseñada para una comunicación eficiente entre dispositivos distribuidos, con una gran duración de baterías y diseñado para aplicaciones tanto urbanas como agrícolas. [14]

NB-IoT es una especificación de red desarrollada por 3GPP y estandarizada en junio de 2016 que pretende imponerse como una solución LPWAN de grado industrial que garantice un servicio de calidad preparado para el futuro de las comunicaciones modernas. Para ello Narrowband IoT opera en un espectro de frecuencias licenciadas para garantizar la operabilidad del servicio y está basada en la red celular LTE, ofreciendo una cobertura adicional de 20 dB sobre el GSM. Se puede entender como la respuesta de las operadoras móviles al futuro de las comunicaciones M2M para el IoT, ya que trabaja bajo las mismas frecuencias e infraestructuras existentes para la tecnología LTE, pero garantizando unas comunicaciones eficientes con bajo consumo energético y manteniendo la seguridad en las comunicaciones de las redes LTE.

Se trata de una tecnología todavía en pruebas pero que se está desplegando actualmente en España (concretamente en Valencia y Madrid) por parte de Vodafone [15] y promete que dará mucho que hablar en un futuro cercano.

La filosofía es parecida a la que ofrece Sigfox, con una suscripción de pago para hacer uso de la infraestructura de red, pero operando bajo el estándar de la LTE. Los dispositivos o sensores precisan de una tarjeta de identificación SIM similar a la que se usa para los teléfonos móviles que

Memoria

les brinda de una conexión bidireccional cifrada y pueden convivir hasta 100.000 en una misma celda o repetidor.

La ventaja de esta tecnología es precisamente el uso de la red GSM o LTE ya existente y la operabilidad en un espectro de frecuencia licenciado que evita las colisiones de envío que se producen en otras tecnologías LPWAN de banda estrecha en espectros de licencia libre.

	LoRa	Sigfox	NB-IoT
cobertura	160dB	157 dB	164dB
tecnología	propietaria	propietaria	Abierto LTE
espectro	no licenciada	no licenciada	Licenciada (LTE /otra)
restricciones de envío	si	si	no
restricciones potencia transmitida	si (14dBm = 25mW)	si (14dBm = 25mW)	no (23 dBm = 200mW)
velocidad de bajada	<0.1 kbps	<10kbps	0,5-200kbps
velocidad de subida	<0.1 kbps	<10kbps	0,3-180 kbps
vida batería (200b/día)	10+	10+	15+
seguridad	baja	baja	muy alta

Tabla 3 - Comparativa NB-IoT (eSmartCity.es)

Ya existen actualmente proyectos enfocados a las ciudades inteligentes haciendo uso de esta tecnología, como el proyecto valenciano *TSwasTe*, un sensor de llenado de contenedores de bajo coste diseñado para facilitar un despliegue rápido, económico y flexible para la gestión de residuos urbanos en las ciudades. Está equipado con un sensor de ultrasonidos y es capaz de monitorizar diferentes parámetros del contenedor. Además, se puede reprogramar remotamente a través de la red, gracias a la bidireccionalidad de la comunicación NB-IoT. [16]

4.2.6.5. Ingenu RPMA

Random phase multiple Access (RPMA) es una tecnología de comunicación inalámbrica propietaria desarrollada por la empresa californiana Ingenu y diseñada específicamente para dispositivos IoT. Opera principalmente en Estados Unidos en la banda ISM de 2.4GHz, como el WiFi. Esto en un principio lo hace más sensible a pérdidas e interferencias, hecho que la empresa soluciona instalando dos antenas en sus dispositivos. La empresa gestiona la infraestructura tanto física como de red al igual que Sigfox y a diferencia de LoRaWAN.

A pesar de usar la banda de 2.4GHz, la empresa promete una de las mejores coberturas por punto de acceso con tan solo 1 punto de acceso por cada 17 puntos de acceso necesarios para cubrir el mismo radio haciendo uso de LoRa o 70 en el caso de Sigfox. [17]

La ventaja principal de usar el espectro de 2.4GHz es que está disponible mundialmente para el uso libre. Esto permite a las compañías desarrollar una sola versión de su producto para ser comercializado internacionalmente y facilitar así la expansión global, a diferencia de otras tecnologías LPWAN que hacen uso de bandas Sub-GHz.

RPMA también es capaz de soportar una transmisión bidireccional de forma efectiva, incluso con velocidades mayores que otras tecnologías competidoras. Todo indica a que todas estas ventajas son a costa de un mayor consumo de energía en la transmisión.

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

Ingenu dispone de su propia plataforma y una API Restful para facilitar la implementación de la tecnología en la aplicación final, aunque no es accesible a cualquiera, y por el momento obligan a tener un proyecto sólido para poder hacer uso de la tecnología.

4.3. Microcontrolador

En lo que respecta a la elección de un microcontrolador las opciones existentes en el mercado son infinitas, y es que son muchos los fabricantes de semiconductores que ofrecen varios rangos de microcontroladores en su gama de productos.

Un microcontrolador puede entenderse como un circuito integrado programable que es capaz de ejecutar órdenes grabadas en una memoria. No hay que confundir un microcontrolador con un microprocesador, puesto que, aunque ambos pueden formar parte de un sistema embebidos, sus diferencias son bastante notorias. Un microcontrolador se diseña de tal manera que tenga todos los componentes que son necesarios para su funcionamiento en el propio circuito integrado y por tanto no necesita de otros componentes especializados para su funcionamiento. Un microprocesador, en cambio, es una unidad lógica de procesamiento que en la actualidad ya integra muchos periféricos de entrada y salida pero que precisa de una circuitería externa como memorias donde almacenar el programa y donde guardar variables de ejecución de acceso aleatorio, buses de comunicación o circuitos integrados de gestión y suministro de energía.

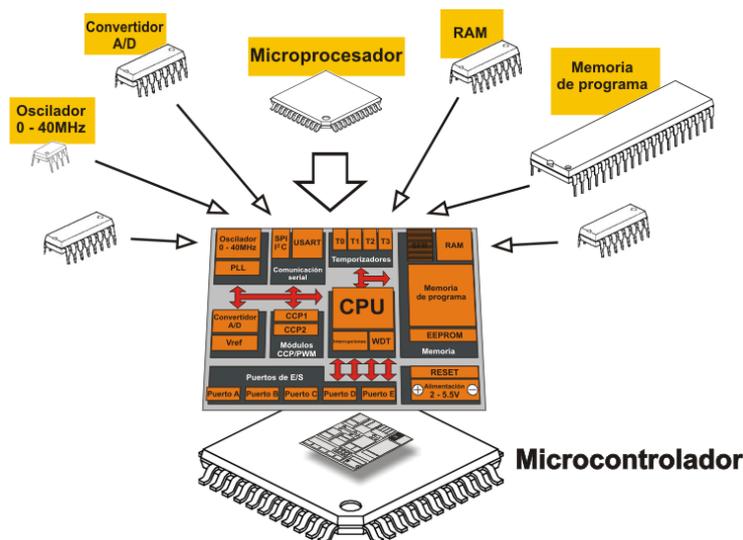


Figura 26 - Estructura de un microcontrolador(aprendiendoarduino.com)

A diferencia de un microprocesador, un microcontrolador tendrá por norma general mucha menor potencia de procesamiento, no permitiendo generalmente cargar un sistema operativo complejo con gestión de procesos. En cambio, será la opción ideal en aplicaciones que no demanden una potencia de procesamiento brutal, que requieran de uno o varios pocos de procesos sencillos con un gasto energético muy eficiente para dispositivos embebidos. Un microcontrolador moderno integra una memoria FLASH para almacenamiento de las instrucciones o programas, una memoria programable tipo EEPROM para guardado de datos que no se desea que se pierdan y una memoria de acceso aleatorio (RAM) para almacenamiento de variables y procesos en ejecución, además de muchos periféricos como relojes/contadores, buses de datos para comunicación con otros periféricos como I2C, SPI o puertos serie (UART) y periféricos de entrada y salida de propósito general configurables. También cuentan con varios canales de lectura analógica (ADC),

Memoria

convertidores digitales a analógicos generadores e incluso transeptores de comunicación inalámbrica. Hoy día encontramos un microcontrolador en casi cualquier dispositivo electrónico.

Con anterioridad a la invención de los microcontroladores, los equipos electrónicos debían de diseñarse con puertas lógicas discretas y circuitos analógicos complejos que eran difíciles de desarrollar y depurar y poco versátiles a la hora de introducir cambios o mejoras. Gracias a estos pequeños dispositivos, la electrónica pasa de ser un problema de diseño y cálculos matemáticos a estar orientado a la programación de instrucciones. Esto permite que los desarrollos electrónicos sean mucho más versátiles al poder ser reprogramados de nuevo para cambiar, mejorar o incluso añadir nuevas funcionalidades.

Existen microcontroladores casi para cualquier aplicación que se pueda imaginar. Algunos enfocados a la adquisición y procesado digital de señales (DSP), otros diseñados expresamente para comunicaciones y microcontroladores específicos para vehículos (ECU) o aviones y por último los enfocados a un propósito general. Pueden ser clasificados por varios factores, pero lo más común es clasificarlos por tamaño de palabra y arquitectura.

Cuando nos planteamos un proyecto, es importante definir que necesidades vamos a tener ya que esto influirá en el tipo de microcontrolador que vamos a utilizar. El mercado de microcontroladores actual es infinito y es fácil perderse, por tanto, es tarea del ingeniero electrónico el hacer un estudio de mercado y seleccionar aquel microcontrolador que más se adecue a las necesidades del proyecto concreto. Dentro de la gran cantidad de microcontroladores integrados que existen, se ha optado por estudiar a grandes rasgos 3 opciones por sus diferentes arquitecturas que se pasan a describir a continuación.

4.3.1. Arduino

Arduino no es un microcontrolador en sí, si no una plataforma de desarrollo basada en un microcontrolador de la familia AVR de Atmel (ahora Microchip). Arduino es un proyecto, comunidad y compañía *open source* y *open hardware* que acerca y facilita el uso de la electrónica y la programación de sistemas embebidos. Arduino fue concebido en 2005 como un programa para estudiantes en el Interaction Design Institute en Ivrea, Italia, con el objetivo de proporcionar una forma fácil y económica a principiantes y profesionales para crear dispositivos que pudieran interactuar con su entorno mediante sensores y actuadores [18].



Figura 27 – Placa desarrollo Arduino Uno

Arduino surge del proyecto *Wiring* y el proyecto original se compone de una placa electrónica programable basada en un microcontrolador con la circuitería necesaria para ser programado fácilmente a través de un ordenador y un entorno de desarrollo integrado (IDE) con una biblioteca de funciones sencillas para programar fácilmente el microcontrolador. Además, facilita la conexión de sensores, periféricos y actuadores gracias a sus conectores universales. Pero la versatilidad del proyecto Arduino reside en su comunidad, y es que, gracias a ser de código libre, actualmente existen cientos de placas compatibles y programables mediante Arduino siguiendo la misma filosofía de acercar la electrónica a personas no familiarizados con la ingeniería.

La placa de desarrollo Arduino más conocida, usada y replicada es la *Arduino Uno* (Figura 27). Está basada en el chip *ATmega238P*, un microcontrolador CMOS de 8 bits de la familia AVR de Atmel (ahora Microchip), basado en la arquitectura RISC (*Reduced instruction set computing*) [19].

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

No es el más potente de la familia, pero sí el más equilibrado en cuanto a relación precio/potencia, y dispone de todo lo necesario para una gran cantidad de sistemas embebidos sencillos que no requieran de una gran potencia de procesamiento. Dispone de 131 instrucciones, 32Kbytes de memoria flash para almacenamiento de programas, con capacidad de programación de un bootloader, y 2Kbytes de memoria RAM. Además de disponer una cantidad muy interesante de periféricos como 3 timers (dos de 8bits y otro de 16), un reloj de tiempo integrado, un convertidor analógico a digital de 8 canales con 10 bits de resolución capaz de hacer quince mil conversiones por segundo, buses SPI e I2C y una UART. También dispone de 23 pines de propósito general que pueden ser configurados como entrada o salida, dos de ellos con capacidades de generar interrupciones externas por hardware y otras muchas más configuraciones y capacidades que pueden ser encontradas en la hoja de datos correspondiente (Figura 28). La placa *Arduino Uno* oficial puede ser comprada a un precio muy asequible de 20€ [20].

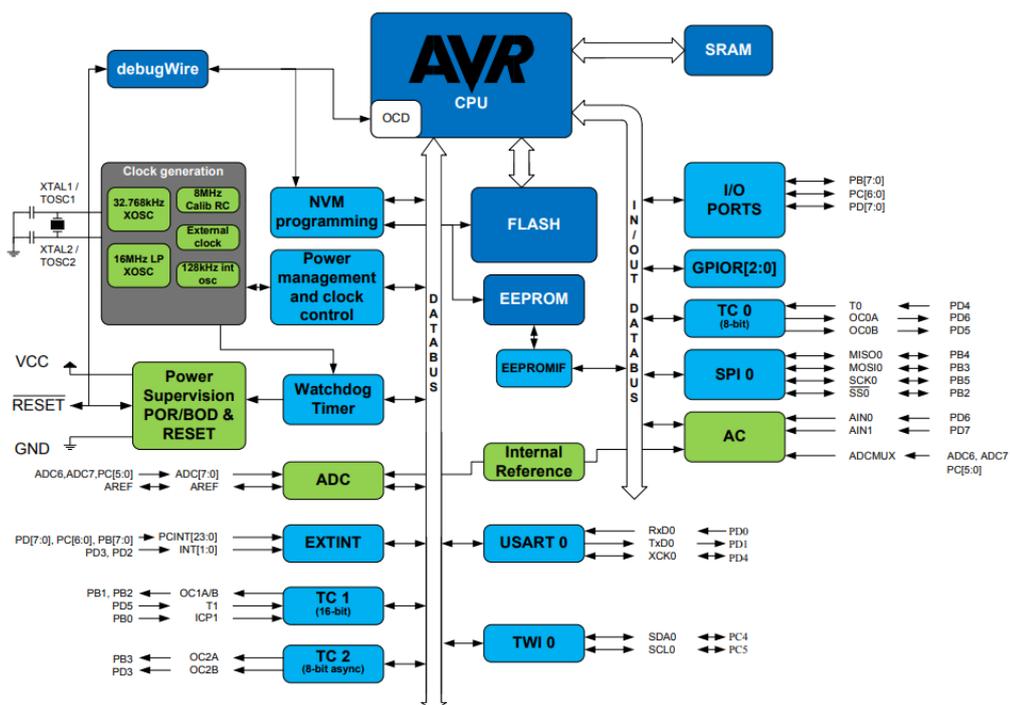


Figura 28 - Diagrama de bloques ATmega328P

No obstante, Arduino sigue creciendo y actualmente dispone de muchísima variedad de placas de desarrollo oficiales y no oficiales compatibles con su IDE y bibliotecas, en concreto, recientemente se está enfocando en el *Internet de las Cosas* y dispone de más de una decena de placas y módulos con conectividades inalámbricas como **WiFi** (MKR 1010), **Sigfox** (MKR 1200), **LoRa** (MKR 1300), **GSM** (Arduino MKR 1400) o **Narrow Band IoT** (MKR 1500), entre otras. Estas tarjetas de desarrollo están basadas en procesadores ARM de la familia SAM, de bajo consumo energético con modos *sleep* muy eficientes y transceptores de radio frecuencia inalámbrica que permiten disponer de un sensor comunicándose a la red de manera muy fácil y barata en pocas horas.

Memoria



Figura 29 - Arduino MKR WAN 1300 con precio

Además, al ser un proyecto con licencia abierta, Arduino provee de toda la documentación referente al diseño (esquemas electrónicos, planos de circuitos impresos y archivos de fabricación) de las placas que pone en el mercado, permitiéndonos aprender de ellas, entender cómo funcionan y utilizarlas en nuestros propios diseños customizados. Arduino distribuye sus proyectos con una licencia Creative Commons la cual además permite incluso que se haga un uso comercial haciendo uso de sus desarrollos sin pedir permiso [21].

4.3.2. ARM Cortex M0+

El ARM[®] Cortex[®]-M0+ tampoco es un microcontrolador como tal, sino uno de los procesadores más pequeño disponible de 32-bit y un gasto energético increíblemente eficiente para dispositivos pequeños orientados a los mercados actuales [22].

ARM es una arquitectura RISC de 32 bits concebida originalmente por *Acorn computers* para su uso en ordenadores personales, que permite una reducción de costes, calor y energía debido a que requiere de un menor número de transistores que los procesadores CISC, típicos en la mayoría de los ordenadores personales. Esta simplicidad los hace ideales para aplicaciones de baja potencia que funcionen con baterías como los teléfonos móviles, tabletas y relojes inteligentes. Actualmente la mayoría de los teléfonos móviles inteligentes, por no decir todos, montan procesadores con una arquitectura ARM. Para entender cómo funciona la arquitectura ARM, hay que saber que esta es licenciable, lo que significa que el negocio de ARM como empresa es la venta de núcleos IP¹, que se utilizan para crear tanto microcontroladores como procesadores o CPUs basados en estos núcleos y, por tanto, ARM no fabrica ni diseña microcontroladores o microprocesadores como tal, sino que vende licencias para hacer uso de los núcleos diseñados. Posteriormente será tarea de un fabricante de semiconductores como por ejemplo *STMicroelectronics* o *Microchip*, el integrar estos núcleos IP en silicio junto con varios periféricos para formar un microcontrolador.

Por tanto, el ARM Cortex-M0+ no es un microcontrolador, ni si quiera un microprocesador, sino un núcleo hardware funcional que describe un procesador con unas características determinadas y que los fabricantes o montadores de circuitos integrados pueden integrar en sus diseños previo pago de una licencia correspondiente. Sin embargo, se hace esta aclaración, para dar paso a explicar los microcontroladores de la serie **STM32L0** del fabricante *STMicroelectronics* y en concreto el microcontrolador **STM32L072CZ**. Esta serie de STM está enfocada a ofrecer unos

¹ Un **Núcleo IP** es un conjunto de bloques funcionales de un circuito electrónico, cuyo uso está autorizado a otras empresas por el diseñador original. En general se trata de una parte de un procesador compleja.

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

microcontroladores *entry-level* de 32-bit con un ultra bajo consumo de energía y una relación de potencia muy eficiente. Esta serie de microcontroladores está basada en el núcleo ARM® Cortex®-M0+ anteriormente mencionado, montado junto con una serie de periféricos que varía en función del microcontrolador concreto, como convertidores analógicos a digital de 12-bits para sensores, convertidores digitales a analógicos de dos canales para audio, periféricos de comunicación y diferentes modos de ultra-bajo consumo (modos *sleep*) para ofrecer la mayor duración posible para dispositivos alimentados de manera autónoma [23].

Existen otros muchos núcleos funcionales ARM, cada uno centrado en ofrecer unas funcionalidades determinadas. Otro ejemplo es el integrado en el conocido microcontrolador **STM32F407**, también de STMicroelectronics, el núcleo **ARM Cortex™-M4**. El cual se caracteriza por ofrecer un alto nivel de integración y potencia con una gran cantidad de periféricos, trabajando a una frecuencia de hasta 168MHz y con un conjunto de instrucciones específicas para su uso como DSP². Posee 1Mbyte de memoria Flash y hasta 192KBytes de RAM. Además de ofrecer hasta 136 pines de propósito general trabajando a una frecuencia de 84MHz e incluso un módulo hardware dedicado para controlar una pantalla LCD o un puerto paralelo para una cámara. Toda esta potencia bruta lo hace ideal para su uso como procesador de señales y controlador de procesos industriales.

No obstante, para nuestra aplicación específica nos centraremos en analizar el microcontrolador **STM32L072CZ**, que se caracteriza por ofrecer una conectividad USB 2.0, operando a una frecuencia máxima de reloj de 32MHz, con 192Kbytes de flash, 20KBytes de RAM y 6KBytes de memoria EEPROM. Además de las siguientes características: [24]

- Hasta 84 GPIOs
- Fuente de alimentación entre 1.65V y 3.6V, que permite ser alimentada directamente por baterías sin necesidad de regulador
- 0.29uA de consumo en su modo más dormido
- 93 uA/MHz en modo normal
- ADC de 12 bits con hasta 16 canales
- DAC de 12-bit con dos canales
- Direct Memory Access (DMA)³ con soporte ADC, I2C, SPI, USART, DAC y timers
- 11 periféricos de comunicación
- 11 relojes o *timers* de 16-bit
- Depurador⁴ embebido

Por tanto, se concluye con que esta familia de microcontroladores STM resulta una opción muy interesante a considerar debido sobre todo a su ultra-bajo consumo que lo hace ideal para aplicaciones donde se requiera de un dispositivo capaz de operar con una sola carga de batería un gran número de años. Y, por tanto, se selecciona el microcontrolador **STM32L072CZ** para el desarrollo del nodo sensor como unidad recolectora y procesamiento de los parámetros de los sensores. Su labor será la de leer los datos de los sensores, procesarlos si es necesario y enviarlos haciendo uso de la modulación adecuada. En la *Sección 5.1.2* se explicará en profundidad como se integra el microcontrolador con los sensores y el transceptor de radio frecuencia LoRa.

² Un **procesador digital de señales** (DSP, por sus siglas en inglés), es un microcontrolador con una serie de instrucciones y hardware dedicado optimizado para operaciones numéricas a muy alta velocidad, ideal para el tratamiento de señales analógicas a altas frecuencias.

³ El **Direct memory Access (DMA)** es una cualidad de los ordenadores o microcontroladores que permite a ciertos subsistemas hardware acceder a la memoria principal del sistema sin representar un cargo en la CPU.

⁴ Un **depurador** (en inglés, *debugger*), es un conjunto de instrucciones de una CPU que permite que el programa se ejecute paso por paso para facilitar la depuración de código de programa en las fases de desarrollo.

Memoria

4.3.3. Expressif ESP32

Expressif Systems es una compañía multinacional dedicada al desarrollo, diseño y fabricación de circuitos integrados. De reciente creación (2008) y con sede en Shanghai, China, *Expressif* cuenta con oficinas en China, India y Europa dedicadas a la investigación y desarrollo de soluciones integradas IoT para dar solución a la gran demanda de dispositivos de comunicación inalámbrica. Su modelo de negocio, a diferencia de sus competidores más directos, se basa en ofrecer un número reducido de soluciones, muy versátiles y a un precio muy bajo debido a su gran volumen de fabricación. En concreto *Expressif Systems* tan solo ofrece tres SoCs⁵ con algunas pocas variantes y unos pocos módulos integrados, que le permite ofrecer un gran soporte a desarrolladores manteniendo un precio muy competitivo, al poder fabricarlos en muy grandes cantidades, bajando el costo por unidad. Los circuitos integrados de la compañía se caracterizan por disponer de un transceptor WiFi/Bluetooth embebido, facilitando la integración de los sistemas.



Figura 30 - Tarjeta de desarrollo con el módulo ESP-WROOM-32

Los tres circuitos integrados de *Expressif Systems* son el ESP8266 (un núcleo con WiFi), el ESP32 (doble núcleo con WiFi + BLE) y el ESP8089 (tan solo transceptor WiFi).

En concreto nos centraremos en el ESP32 que ofrece un transceptor integrado WiFi y Bluetooth Low Energy con un procesador potente de doble núcleo en un paquete de montaje superficial reducido de tan solo 5x5mm con 48 pines. Estas son sus características más interesantes extraídas de su hoja de datos:

- 520KBytes de memoria RAM
- 448KBytes de memoria ROM que obligan a disponer de una Flash SPI externa para almacenamiento de programas
- Compatible con IEEE 802.11 b/g/n hasta 150Mbps
- Configuración como punto de acceso y como estación
- Compatible con Bluetooth 4.2 BLE
- 34 pines de propósito general
- 2 canales de Convertidor digital a analógico de 8-bit a gran velocidad
- 18 canales de ADC de 12-bit
- Interfaz I2S para audio
- UART, I2C, SPI, CAN, eMMC, touch sensors...

Además, el fabricante nos ofrece un módulo certificado que integra el SoC ESP32 con una unidad Flash SPI externa de 4MBytes para almacenamiento de programa y una antena de 2.4GHz integrada en el circuito impreso para el correcto funcionamiento del WiFi y BLE. Lo cual facilita

⁵ Un **sistema en chip** o SoC (del inglés *System On a Chip*) se trata de una tecnología de fabricación que integra una gran parte de los módulos que componen un sistema electrónico o computador en un único circuito integrado o chip.

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

enormemente la inclusión de este SoC a los desarrolladores, al no tener que hacer un diseño de la parte de radio frecuencia, que a menudo resulta muy complejo, y permitir reducir el *time-to-market* gracias a la inclusión de un módulo ya certificado.

El integrado dispone de varios modos de bajo consumo o modos *sleep*, incluso con la integración de un RTC ⁶ de bajo consumo para establecer periodos de baja inactividad. Pero en la práctica lo cierto es que resulta difícil conseguir un bajo consumo aceptable aun bajando la frecuencia del procesador y haciendo uso de un solo núcleo, y aunque no es imposible, resulta dificultoso el hacer un consumo eficiente llegándose incluso a calentar en exceso. [25]. Es por ello por lo que no es la solución ideal si se desea diseñar un dispositivo alimentado a baterías, incluso si éstas son recargables, ya que en el mejor de los casos obtendríamos autonomías de pocas semanas.

Existen tarjetas de desarrollo con el ESP32 preparadas para transmitir usando LoRa, como la *ESP32 LoRa IoT Board* de Heltec, la cual integra un ESP32 con el transceptor SX1276 e incluso una pantalla OLED.

4.4. Transceptores LoRa

La tecnología de radio frecuencia utilizada para el desarrollo del proyecto va a ser LoRa y para hacer uso de esta tecnología, existen varias soluciones en el mercado. En esta sección se hace una comparación de distintos transceptores o *transceivers* para hacer uso de esta modulación. Es de vital importancia hacer una buena elección del transceptor adecuado para las necesidades que se requieran.

Transceiver o transceptor viene de la combinación de las palabras *Transmisor* y *Receptor* (o *Transmitter* y *Receiver*, en inglés). Se trata de un dispositivo que cuenta por tanto con un transmisor y un receptor de radio frecuencia que comparten parte de la circuitería (normalmente la antena y el driver de salida) y se encuentra dentro de la misma caja/paquete o circuito integrado. Dado que determinados elementos del circuito son compartidos y se utilizan tanto para la transmisión como para la recepción, la comunicación solo puede ser *semidúplex*, lo que significa que pueden enviarse señales en ambos sentidos, pero no simultáneamente. Resulta necesario y mucho más eficiente hacer uso de un dispositivo transceptor si se requiere una comunicación bidireccional. Un ejemplo de transceptores son los chips WiFi que integran todos los dispositivos móviles actuales.

En el caso de LoRa, que es una modulación propietaria de la empresa Semtech, el transceptor usado en nodos es el **SX1276/77/78/79**. Se trata de un modem integrado en un factor de forma pequeño patentado con la técnica de modulación LoRa y un *link budget*⁷ máximo de 168dB y una sensibilidad⁸ de -148dBm. Esta solución de transceptor es usada en prácticamente todos los módulos que existen en el mercado para modulación LoRa. En este capítulo estudiaremos en profundidad algunos de esos módulos existentes en el mercado, ya que el uso de uno de estos módulos simplifica enormemente la tarea de diseño y desarrollo, permitiendo olvidarse del desarrollo de radio frecuencia, centrándose en la programación y otras partes del diseño electrónico más precisas. Es por ello por lo que se opta por hacer uso de una solución embebida que permita al Ingeniero electrónico olvidarse del diseño RF para centrarse en el diseño de los sensores y la

⁶ Un *Reloj en tiempo real* (en inglés, real-time clock, **RTC**)

⁷ **Link Budget** o literalmente, presupuesto de enlace, es un cálculo utilizado para determinar los requerimientos de energía necesarios para un enlace particular y tiene en cuenta tanto las pérdidas como las ganancias producidas en ese enlace. Es ampliamente utilizado para medir la calidad de un sistema inalámbrico. Expresada en escala logarítmica.

⁸ La sensibilidad en radio frecuencia determina el nivel de señal más débil que el receptor es capaz de recibir con una reproducción aceptable de la señal modulante original.

Memoria

conexión de los demás módulos del sistema y facilite la obtención de un producto certificado que cumpla con las normas de emisión y radiación correspondientes. No obstante, en un futuro, y pensando en una solución comercial, lo óptimo para una reducción de costes, sería implementar todo el sistema en un único módulo o circuito impreso más integrado.

4.4.1. HopeRF RFM95W

Se trata de un módulo de radiofrecuencia basado en el integrado SX1276 de Semtech descrito en la sección anterior. Este módulo integra todo lo necesario para que el chip SX1276 funcione correctamente como el cristal oscilador, el conmutador de la antena, más conocido como *RF switch*⁹, desacoplo de la alimentación y varias resistencias, condensadores, bobinas e impedancias para definir la correcta adaptación de la antena a la frecuencia deseada. [26]



Figura 31 - Módulo HopeRF FRM95W

Este módulo se conecta a un microcontrolador mediante la interfaz de comunicación SPI o *Serial Peripheral Interface*. Este bus serie o estándar de comunicación es usado principalmente para la transferencia de información entre circuitos integrados en equipos electrónicos. Se trata de un bus síncrono, pues consta de una señal de reloj (SCK), además de una señal de comunicación desde el dispositivo master (normalmente el microcontrolador) al dispositivo esclavo (*Master Output – Slave Input* o MOSI) y una señal de datos del dispositivo esclavo al master (*Master Input – Slave Output* o MISO). Además, se le añade una línea de control (*chip select* o SS) para poder acoplar varios dispositivos al mismo bus de comunicaciones y poder activar o desactivarlos individualmente. Es uno de los buses más utilizados junto con el I2C, pero a diferencia de este, el SPI ofrece una comunicación *Full Duplex* a mayor velocidad. Por el contrario, precisa de más pines para el control de varios dispositivos al no tener una filosofía de direcciones como el protocolo I2C, explicado anteriormente en la *Sección 4.1*. Por tanto, este módulo permite integrar comunicación LoRa con tan solo usar un puerto SPI del microcontrolador. No obstante, todo el manejo de la comunicación ha de ser efectuada por el microcontrolador. [27]

Existen varias versiones de este módulo, las cuales se diferencian en algunas características que se resumen en la *Tabla 4*. Se podrán elegir características como el rango de *Spreading Factor* con el que puede transmitir, la sensibilidad y lo que es más importante, la frecuencia. Es importante seleccionar el módulo de la frecuencia correcta que deseemos usar, ya que la red de impedancia para la antena estará diseñada para esa frecuencia específica, que en el caso de Europa para LoRa es normalmente de 868MHz.

Aunque finalmente este módulo transceptor no se ha usado en el proyecto, si que se ha experimentado satisfactoriamente usando un Arduino Uno. El módulo ha sido integrado en la tarjeta electrónica diseñada junto con los sensores, que se verá más adelante. No obstante, este módulo

⁹ La función de un *RF switch* o conmutador de radio frecuencia es un dispositivo encargado de enrutar la señal de radio frecuencia o incluso de conmutar entre varias señales a un mismo path.

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

no tiene una certificación de compatibilidad electromagnética europea, lo cual dificulta la salida en un futuro del producto al mercado.

Part Number	Frequency Range	Spreading Factor	Bandwidth	Effective Bitrate	Est. Sensitivity
RFM95W	868/915 MHz	6 - 12	7.8 - 500 kHz	.018 - 37.5 kbps	-111 to -148 dBm
RFM97W	868/915 MHz	6 - 9	7.8 - 500 kHz	0.11 - 37.5 kbps	-111 to -139 dBm
RFM96W/RFM98W	433/470MHz	6- 12	7.8 - 500 kHz	.018 - 37.5 kbps	-111 to -148 dBm

Tabla 4 - Variantes RFM95W de HopeRF

4.4.2. RAK Wireless RAK811

El RAK811 es un módulo LoRa desarrollado por la compañía china RAK Wireless también basado en el SX1276 de Semtech. La diferencia respecto al de HopeRF reside en que este además del integrado SX1276 con su oscilador y toda la circuitería auxiliar para hacerlo funcionar, dispone de un microcontrolador de STMicroelectronics conectado a él. Es microcontrolador se comunica por SPI con el SX1276 y además integra todo el stack del protocolo LoRaWAN para usarlo como dispositivo de Clase A y Clase C. Esto permite que pueda ser integrado de manera extremadamente fácil, reduciendo las horas de desarrollo que conlleva implementar el stack¹⁰ en un microcontrolador. Además, permite hacer uso de un microcontrolador mucho más sencillo, y es que tan solo precisa de un puerto serie (UART) con el que enviar comandos AT¹¹ al módulo. Su consumo energético también es muy eficiente, al integrar un microcontrolador STM32L1 Cortex-M3 de muy bajo consumo con varios modos dormido que en la práctica funcionan muy bien.



LoRa Module RAK811

- ✧ Embedded LoRaWAN protocol class A&C, easily connects to LoRa technology gateway.
- ✧ LoRaWAN Protocol Stack ready in system.
- ✧ Full certification by region.
- ✧ Long Range - Greater Than 15 km.
- ✧ Simple ASCII command set.
- ✧ Serial Port AT Command, Automatic Modification of Frequency Points Supported.

 Buy Now

Figura 32 - Módulo RAK811 de RAK Wireless

Además, la compañía RAK Wireless dispone de una placa de evaluación conocida como WisNode-LoRa, con el formato *shield*¹² de Arduino para desarrollar la aplicación de usuario de manera fácil y económica haciendo uso del RAK811 (Figura 33).

¹⁰ Un **stack** en este caso se refiere a una pequeña capa de abstracción o conjunto de instrucciones para hacer uso de una tecnología específica. En este caso, el stack de LoRaWAN se trata de una máquina de estados para manejar el protocolo de comunicación de la capa 3 del modelo OSI.

¹¹ Los **comandos AT**, también conocidos como *conjunto de comandos Hayes* es un estándar abierto de comandos para configurar y parametrizar modems.

¹² Un **shield o escudo** de Arduino es un formato estándar para añadir encima de un Arduino Uno y del que hacen uso muchos diseñadores, sobre todo en el mundo Maker.

Memoria

Adicionalmente, este módulo viene completamente certificado para hacer uso de el en un producto comercial y reducir el *time-to-market*.

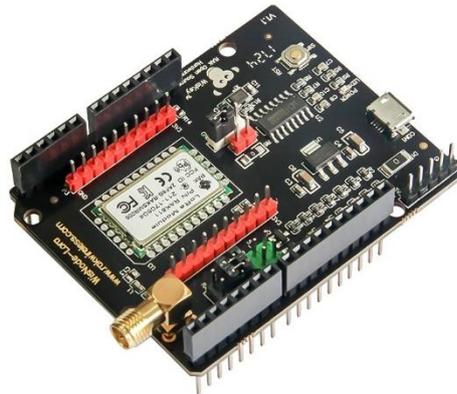


Figura 33 - Placa de desarrollo WisNode-Lora

4.4.3. Murata CMWX1ZZABZ

Murata es una empresa manufacturera de componentes electrónicos con base de operaciones en Kyoto, Japón. Su volumen de negocio principal es la fabricación de componentes pasivos, principalmente condensadores cerámicos de gran calidad y con mucha reputación en el mercado. No obstante, recientemente se está introduciendo en el sector del Internet de las Cosas y la radio frecuencia con varios módulos de comunicación inalámbrica como módulos WiFi, Bluetooth y LPWA (Sigfox y LoRa por el momento). Dichos módulos son de una excepcional calidad en cuanto a construcción y diseño y representan un estándar en la industria debido al gran soporte técnico y la buena reputación de la compañía.

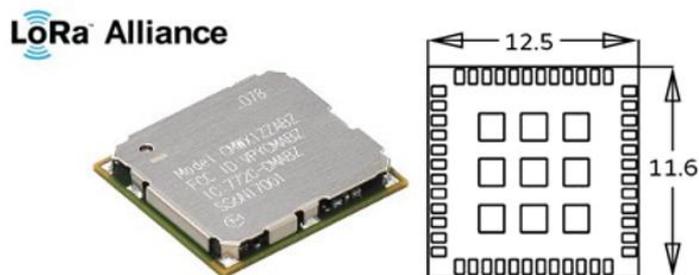


Figura 34 - Murata módulo LoRa CMWX1ZZABZ

En concreto, la solución LoRaWAN por parte de Murata se conoce como **CMWX1ZZABZ** e integra el transceptor **SX1276** de Semtech junto al microcontrolador **STM32L072CZ** (Figura 48). A efectos prácticos, se trata de un microcontrolador STM32L0 de bajo consumo dotado de una conectividad LoRa gracias al chip de Semtech. Por tanto, ofrece todo lo necesario para diseñar un sensor inalámbrico conectado mediante LoRaWAN, con un tamaño muy compacto de tan solo 12.5x11.6mm (Figura 34), siendo necesario unos pocos condensadores de desacoplo y alguna resistencia para hacerlo funcionar. [28]

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

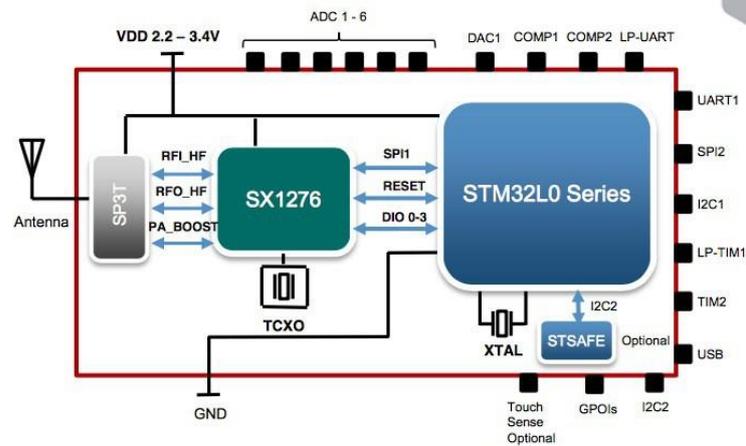


Figura 35 - Diagrama de bloques CMWX1ZZABZ

El módulo está certificado para su uso tanto en Estados Unidos como en Europa, es decir, para las frecuencias de 868MHz y 915MHz y tiene un precio que ronda los 10\$ en grandes cantidades.



Figura 36 - B-L072Z-LRWAN1 - STM32L0 Discovery kit LoRa

Por otra parte, STMicroelectronics fabrica un kit de desarrollo implementado este módulo de Murata, conocido como B-L072Z-LRWAN1 LoRa Discovery kit (Figura 36).

Resulta muy versátil trabajar con este kit de desarrollo debido a la flexibilidad de conexiones que ofrece, los LEDs de depuración, sus conectores hembra para shield de Arduino y su programador y depurador incorporados mediante USB. Además, ofrece un buen soporte por parte de STMicroelectronics y es compatible con las bibliotecas HAL estudiadas en las asignaturas de Informática Industrial II del Grado de Electrónica y que serán descritas en secciones posteriores. Es por ello por lo que se decide utilizar este módulo en el desarrollo de este nodo sensor inalámbrico.

4.5. Plataforma de programación (IDE)

Un entorno de desarrollo integrado (IDE) o plataforma de programación es un software que integra un conjunto de aplicaciones o módulos para el desarrollo, programación y depuración de firmware tanto para microcontroladores como microprocesadores.

Es importante no confundir un IDE con SDK (Software Development Kit), pues un entorno de desarrollo integrado dispone de una interfaz normalmente gráfica de usuario para programar, compilar, desarrollar y añadir bibliotecas y cabeceras, etc. Mientras que un SDK tan solo es un conjunto de herramientas para compilar y programar, precisando así de una interfaz visual adicional o ventana de comandos. Un IDE se puede entender como una interfaz gráfica junto con un SDK.

Aclarado esto, se procede a hacer una comparación de 3 entornos de desarrollo seleccionados dentro de las decenas que existen. Cabe destacar, que según la arquitectura o tipo de microcontrolador con el que vayamos a trabajar será interesante elegir uno u otro. Se seleccionan estos tres ya que los tres han sido probados con la tarjeta de desarrollo B-L072Z-LRWAN1 satisfactoriamente y cada uno de ellos tiene sus ventajas y sus inconvenientes.

4.5.1. ARM Keil uVision

Keil fue la primera compañía en ofrecer un compilador C para uno de los primeros microcontroladores, el 8051 de Intel. Hasta el momento, los microcontroladores eran programados mediante lenguaje ensamblador. Actualmente Keil se ha convertido en uno de los estándares en la industria para la arquitectura ARM, debido a su gran fiabilidad, su soporte técnico al desarrollador y su depurador integrado. Keil uVision es enseñado actualmente en cientos de universidades del mundo.

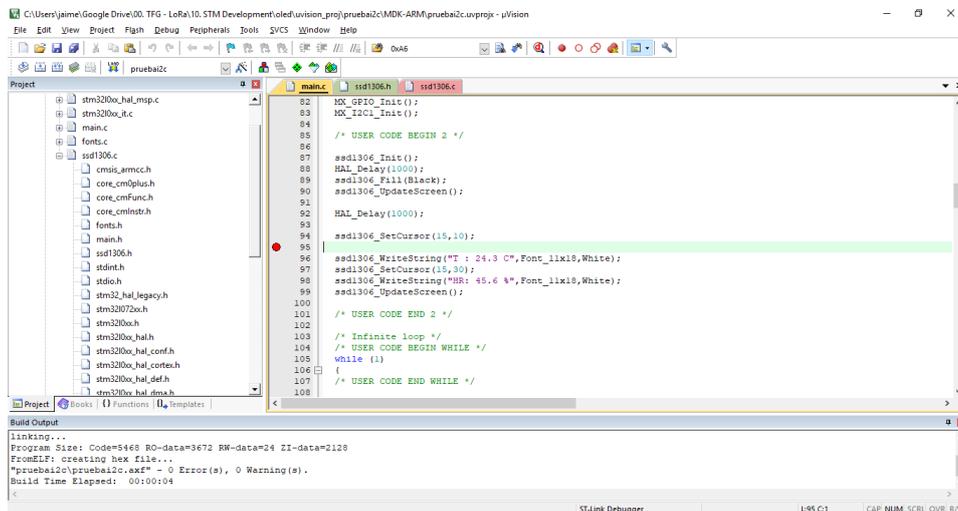


Figura 37 - Ventana principal Keil uVision5 para microcontroladores

Keil uVision IDE combina un editor de código fuente con un compilador de lenguaje C o C++ optimizado para ARM y un depurador propietario de Keil de gran potencia, compatible con la arquitectura ARM. Keil ofrece configurar muchos parámetros y resulta muy flexible. Keil uVision dispone de una licencia de uso gratuito con una limitación de 32Kbytes por programa. No obstante, esta limitación desaparece cuando registras el producto, también de manera gratuita y para microcontroladores pequeños como la serie STM32L0.

Haremos uso de uVision 5 debido a que ha sido aprendido en las asignaturas del Grado de Electrónica Industrial Automática, y a que ofrece una gran versatilidad para el desarrollador, permitiendo configurar una gran cantidad de parámetros. A parte, la herramienta de configuración

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

de microcontrolador basada en las bibliotecas HAL, **CubeMX** de STM, que será vista en capítulos posteriores, es capaz de exportar el proyecto a Keil uVision 5, lo cual facilita enormemente la tarea de desarrollo, ya que una de las tareas más laboriosas es la de configurar el proyecto del IDE correctamente. También STM dispone de ejemplos para la tarjeta B-L072Z-LRWAN1 realizados en Keil uVision con soporte total para el stack de red LoRaWAN.

4.5.2. ARM Mbed

ARM Mbed es una plataforma creada por ARM en 2009 y soportada por STMicroelectronics, que provee de un sistema completo para el desarrollo de aplicaciones integradas para el Internet of Things. Tiene soporte para más de 100 tarjetas de desarrollo y más de 400 componentes y herramientas para editar código, desarrollar y testear aplicaciones, además de herramientas tanto de servidor como de cliente para comunicarse con los dispositivos. [29]

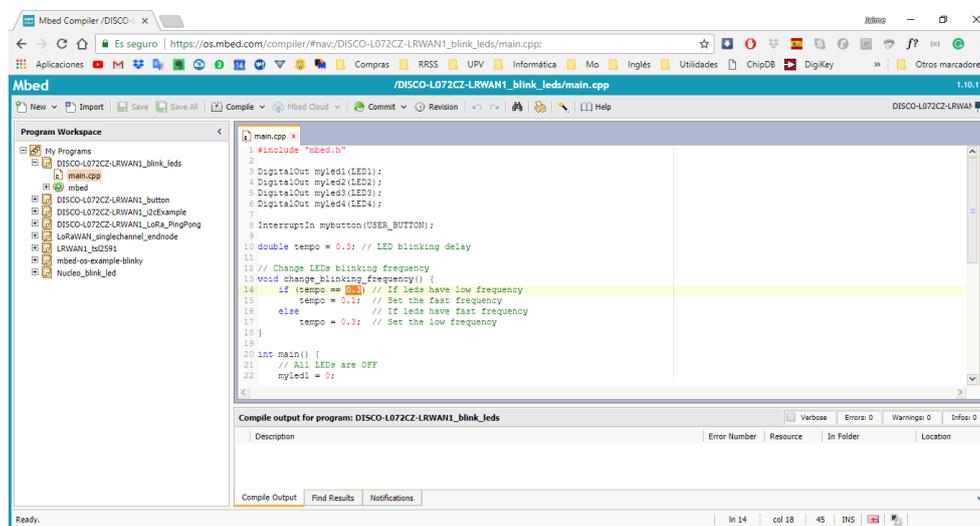


Figura 38 - Compilador ARM Mbed en la nube

ARM Mbed ofrece una herramienta de desarrollo online con un compilador y editor de código fuente en la nube (Figura 38). Además, todas las bibliotecas compatibles están en la nube y los códigos fuente que se desarrollan se almacenan automáticamente en sus servidores, resultando en una integración muy buena. El SDK de ARM Mbed también puede ser descargado en local para funcionar con editores de texto como Atom con Platformio. No obstante, ARM Mbed online no dispone de depurador y los programas son descargados en formato IntelHex¹³ después de ser compilados. Esto resulta farragoso ya que obliga a estar usando una utilidad externa para subir los programas y a hacer uso de un puerto serie para trazar las variables en tiempo de ejecución al no disponer de depurador.

Adicionalmente, por el momento ARM Mbed, aunque si es compatible con la tarjeta B-L072Z-LRWAN1, no cuenta con soporte para el stack de red de LoRaWAN, lo cual obliga a tener que implementar la máquina de estados de red, dificultando mucho la tarea de desarrollo.

Por todas estas razones, se ha descartado usar este IDE, aunque cabe destacar que tiene un futuro muy prometedor.

4.5.3. Arduino IDE

¹³ **IntelHex** es un formato de archivo para la programación de microcontroladores, EPROMS y otros circuitos integrados, datado de los años 70. Consiste en un archivo de texto con valores hexadecimales que codifican los datos junto con su dirección de memoria correspondiente.

Memoria

Arduino IDE es actualmente uno de los entornos de desarrollo para microcontroladores más usados y conocidos del mundo, debido sobre todo a la gran comunidad de desarrolladores existente y la inmensa cantidad de bibliotecas y tarjetas de desarrollo compatibles.

El IDE de Arduino es bastante simple y se inspira en el IDE de Processing. Contiene un editor de texto para escribir el código fuente del programa, un área de mensajes y consola de texto con fondo negro y una simple barra superior con 6 botones para compilar, subir y desplegar la consola serie. Además, tanto instalar y usar una biblioteca en Arduino IDE, como cambiar de tarjeta o microcontrolador, resulta muy sencillo.



The screenshot shows the Arduino IDE 1.8.5 window. The title bar reads "Blink Arduino 1.8.5". The menu bar includes "Archivo", "Editar", "Programa", "Herramientas", and "Ayuda". The toolbar contains icons for opening files, saving, uploading, and running. The main editor area displays the following code:

```
Blink $
by Cory Newman

This example code is in the public domain.

http://www.arduino.cc/en/Tutorial/Blink
*/

// the setup function runs once when you press reset or power the
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is t
  delay(1000); // wait for a second
  digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making
  delay(1000); // wait for a second
}

Compilando programa...
Detecting libraries used...
"C:\Users\jaime\AppData\Local\Arduino15\packages\arduino\tools\arm
```

The status bar at the bottom indicates the board is "B-L072Z-LRWAN1, 32 MHz, Smallest Code, EU868 en COM3".

Figura 39 - Arduino IDE 1.8.5

No obstante, Arduino IDE no cuenta con un depurador y hay muchas configuraciones de parámetros que se encuentran ocultas en archivos de configuración y, aunque no suele ser necesario el realizar cambios, resulta dificultoso si se requiere.

Es compatible con la tarjeta B-L072Z-LRWAN1 e incluso soporta el stack de LoRaWAN, pero no está maduro y estable el desarrollo de las bibliotecas para esta tarjeta y se encuentran ciertos fallos o *bugs* en algunas de sus funcionalidades.

No obstante, Arduino IDE ha sido de gran utilidad en el desarrollo del proyecto para testear funcionalidades aisladas del sistema como la comunicación I2C con los sensores, que de otra manera hubiera resultado más difícil y hubiera requerido de más tiempo de desarrollo.

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

4.6. Batería

Es importante hacer una estimación del consumo medio que va a tener nuestro dispositivo bajo condiciones normales de funcionamiento, para poder hacer un buen dimensionado de la batería y elegir correctamente la tecnología ideal para cada caso. Para ello es importante saber los consumos de cada subsistema del dispositivo en los distintos modos de funcionamiento para realizar una ponderación de consumo medio con el que poder dimensionar la capacidad de la batería según la autonomía que se requiera.

Asimismo, la elección de la batería es uno de los aspectos cruciales en un dispositivo inalámbrico, ya que, por la propia naturaleza del dispositivo, este no tiene acceso a un cable que le provea de alimentación, y tendrá que ser autónomo por un largo periodo de tiempo. La autonomía de un dispositivo de este tipo también puede ser conseguida mediante sistemas de *energy harvesting*, o sistemas recolectores de energía como energía solar, eólica, cinética o incluso electromagnética. Y es que, debido al consumo tan bajo de estos dispositivos, se puede conseguir que estos sean alimentados con una pequeña célula solar o incluso con las propias ondas de radio que viajan en el aire. No obstante, estos sistemas son complejos y requieren de un estudio previo que se sale de la temática de este proyecto y que bien podrían constituir otro trabajo fin de grado.

Sin embargo, la elección de baterías para proyectos conectados a IoT puede resultar difícil, ya que hay muy diversos modelos para cada tipo de aplicación. Existen decenas de tecnologías químicas en lo que respecta a baterías, unas recargables mediante métodos tradicionales y otras no. En este capítulo se estudian y comparan varias tecnologías, unas muy novedosas y otras más tradicionales.

4.6.1. Batería de ion de litio (Li-Ion)

Las baterías de iones de litio son dispositivos diseñados para el almacenamiento de energía eléctrica que emplea como electrolito una sal de litio que consigue los iones necesarios para la reacción electroquímica reversible. Es decir, se trata de pilas recargables.



Figura 40 - Pack de baterías de un vehículo Tesla Model S

Tiene unas propiedades muy interesantes como la ligereza de sus componentes unida a su elevada capacidad energética que le permite conseguir densidades energéticas¹⁴ de entre 250 y 730 Wh/L. Junto con una elevada resistencia a la descarga, un bajo efecto memoria¹⁵ y su capacidad

¹⁴ La **densidad energética** de una batería se define como la energía que puede almacenar dividido entre el volumen de esta. Se suele expresar en Wh/L

¹⁵ El **efecto memoria** en las baterías es un fenómeno que reduce la capacidad de las baterías con cargas incompletas,

Memoria

para funcionar con un elevado número de ciclos, esta tecnología de baterías se ha impuesto como el estándar más usado en aplicaciones de la industria electrónica de consumo como ordenadores portables, teléfonos móviles inteligentes y tabletas. También su uso es cada vez más creciente en bancos de baterías para coches eléctricos y acumuladores de electricidad para instalaciones fotovoltaicas donde están superando tecnológicamente a las viejas y pesadas baterías de plomo-acido.

Sin embargo, las baterías de iones de litio son difíciles de cargar, y precisan de una regulación de voltaje-corriente muy precisa normalmente compuesta por una fase de precarga donde se le aplica una corriente constante muy pequeña para llevarlas hasta un voltaje seguro de unos 3.2V. Posteriormente entran en una fase de carga rápida donde se les aplica una corriente constante de normalmente $1C^{16}$ hasta llegar a un voltaje por celda de 4.2V, momento en el cual pasan a una fase de voltaje constante o carga lenta, donde se les mantiene en un voltaje de 4.2V hasta que la corriente disminuye a un 10% de la corriente de fase de carga rápida (normalmente un 10% de $1C$). Es en este momento, cuando la celda se considera totalmente cargada. (Figura 41).

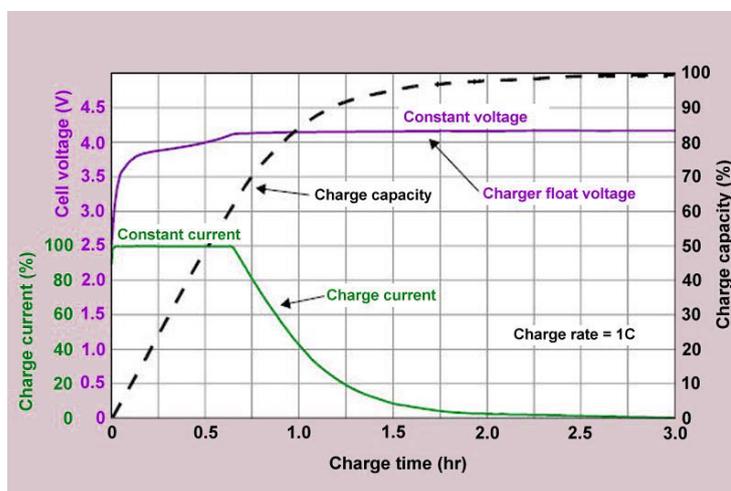


Figura 41 - Curva de carga CC-CV batería Li-Ion (portatilmovil.com)

A menudo las celdas de iones de litio no se usan solas, y normalmente precisan de una configuración serie paralelo para llegar a los requisitos de voltaje y corriente necesarios en cada aplicación. No obstante, debido a la gran variación de voltaje que sufre la química de este tipo de baterías (Figura 42), las celdas han de ser balanceadas durante su carga y descarga para garantizar el buen funcionamiento y durabilidad de cada una de las celdas y extender su vida. Es por ello, por lo que a menudo precisan de complejos circuitos de balanceo de carga y descarga. Además, debido a su química y a los inflamables materiales utilizados en su construcción, resultan peligrosas y pueden llegar incluso a explotar si se sobrecalientan o se hace un uso indebido de ellas, para evitar resulta obligatorio el uso de circuitos de protección contra sobre-corriente, cortocircuito y sub-tensión, ya que si se descargan por debajo de 3 V se pueden dañar.

Las baterías de iones de litio actualmente son la solución ideal para dispositivos portables con baterías pequeñas como los conocidos *wearables* cuyas baterías suelen estar entorno a los 50-100mAh y habitualmente rondan las dos o tres semanas de autonomía en el mejor de los casos. No obstante, nuestra aplicación requiere de al menos varios años de autonomía, por tanto, se descarta la opción de usar batería de iones de litio.

bien por el uso o por unas malas cargas. Una manera de prevenirlo es realizar cargas completas.

¹⁶ $1C$ significa una vez la capacidad de la batería. Es decir, si la batería tiene una capacidad de 100mAh, será cargada a una corriente constante en fase de carga rápida de 100mA.

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

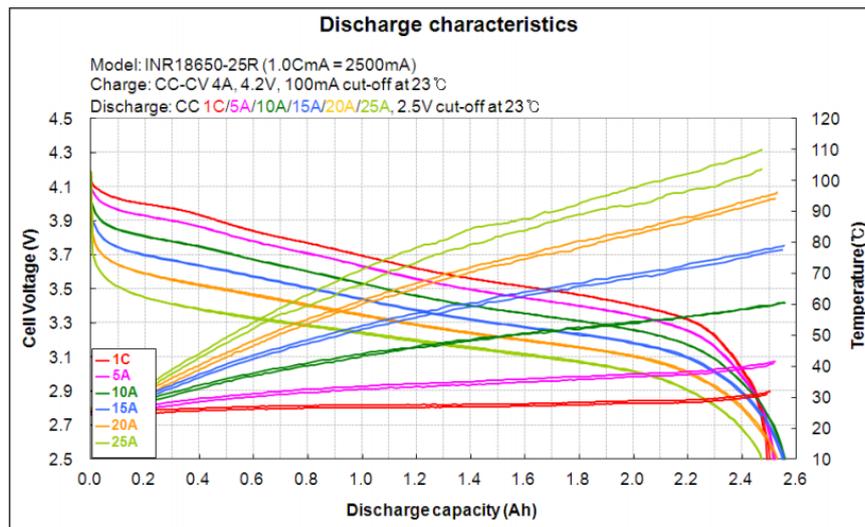


Figura 42 – Curva de descarga de una celda de iones de litio (PowerTechSystems.eu)

4.6.2. LiFePO₄

Las baterías de litio-ferrofosfato, batería LFP o LiFePO₄ es un tipo de batería recargable muy similar a una batería de iones de litio, pero a diferencia de esta, el cátodo es de fosfato de hierro-litio (LiFePO₄).

Presentan una densidad energética algo menor (sobre un 30%) a las más comunes de óxido de litio cobalto (las conocidas comúnmente como iones de litio descritas anteriormente), pero resultan más baratas de fabricar. [30]

La gran ventaja de las baterías LFP reside en su curva de descarga, y es que al igual a las baterías recargables basadas en níquel, las baterías LiFePO₄ tienen un voltaje de descarga muy constante, situado cerca de los 3.2 V nominales. Este voltaje se mantiene casi invariable durante toda la descarga hasta que la pila se vacía por completo (Figura 43)

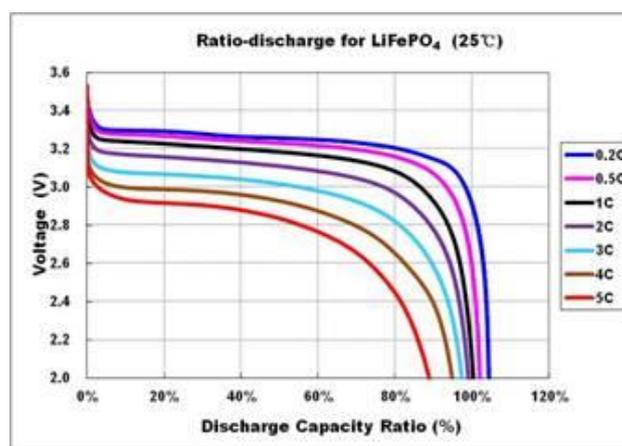


Figura 43 - Curva descarga batería LiFePO₄ (benenergytech.com)

Esto permite a la batería entregar la máxima capacidad disponible hasta su descarga completa y permite simplificar enormemente o incluso suprimir por completo la necesidad de circuitería de regulación de voltaje, aumentando así la eficiencia del sistema al no tener caídas de tensión e

Memoria

ineficiencias debidas a la regulación de voltaje mediante convertidores DC/DC o reguladores lineales. Las baterías LFP también resultan más seguras en su uso, debido a la mayor estabilidad química y térmica y, por tanto, es más difícil que una celda de litio-ferrofosfato entre en combustión en el caso de una mala carga o uso.

4.6.3. NiCa

Las baterías de níquel-cadmio son recargables pero cada vez se usan menos debido a su efecto memoria y al uso de cadmio que es muy contaminante. Actualmente han pasado a ser sustituidas por las NiMH, a pesar de que la vida de las NiCa es mayor y oscila entre los 1000 y 5000 ciclos de carga.

A diferencia de las pilas de litio, las de níquel dan un potencial nominal de 1.2V por celda.

4.6.4. NiMH

Son un tipo de baterías recargables que utilizan un ánodo de de oxihidróxido de níquel (NiOOH), como en la batería de níquel cadmio, pero cuyo cátodo es de una aleación de hidruro metálico. Esto permite eliminar el cadmio que es caro y representa un peligro para el medio ambiente. De esta manera, permite una mayor capacidad de carga con el mismo tamaño y peso y un menor efecto memoria, es por ello por lo que ha sustituido a las baterías NiCa casi totalmente, hasta la aparición de la tecnología de litio, que está empezando a dejar a la tecnología NiMH obsoleta.

Cada pila o celda puede proporcionar un voltaje de 1.2V y una capacidad de entre 800 y 2900mAh con una densidad energética que puede llegar incluso a los 300Wh/L. La carga es sencilla

Con su similitud en cuanto a la tensión nominal que proporciona una pila NiMH, estás muchas veces se usan como sustitutas a las pilas alcalinas de 1.5V convencionales, y resultan ser una opción recargable a estas. Es por ello por lo que se utilizan en productos que funcionan con pilas convencionales, aunque cada vez hay menos y la tendencia es a equipar a los dispositivos con baterías embebidas.

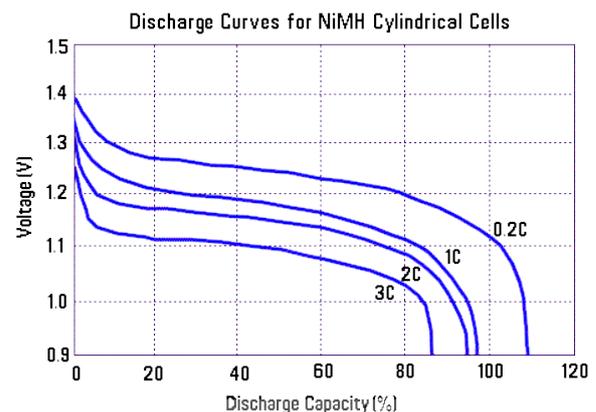


Figura 44 - Curva descarga batería NiMH (priuschat.com)

4.6.5. Li-FeS₂

Las baterías de disulfuro de hierro son un tipo de baterías no recargables que son directamente compatibles con las conocidas baterías alcalinas, ya que ofrecen la misma tensión de 1.5V por celda (aunque puede llegar a 1,83V cuando está completamente cargada), pero respetando los tamaños, pero con una mejora en la capacidad, resistencia interna y curva de descarga muy notable, al emplear una tecnología basada en litio. La Li-FeS₂ existe en tamaños estándar como AA o AAA, y ofrece una autodescarga¹⁷ muy pequeña (de unos 15 años) [31], lo cual es ideal para dispositivos que se desea que funcionen durante varios años.

¹⁷ La **autodescarga** es un fenómeno que se da en baterías por el cual, debido a las reacciones químicas internas, se reduce la capacidad de carga de la batería sin conexión alguna entre los electrodos, esto disminuye la capacidad de la batería sin ser usada.

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

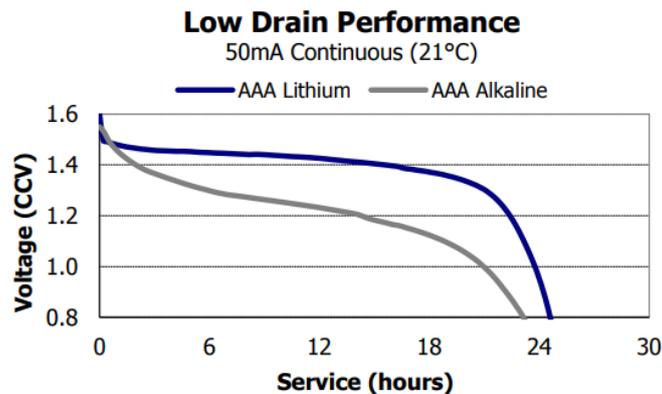


Figura 45 - Comparación Li-FeS₂ con Alcalina (Energizer)

Por su disponibilidad en una gran cantidad de formatos y tamaños, estas pilas serán utilizadas para el desarrollo del prototipo de este proyecto, ya que la tarjeta de desarrollo B-L072Z-LRWAN1 dispone de un portapilas para 3 baterías de tamaño AAA¹⁸ lo cual permite alimentar el sistema de manera inalámbrica. Esto debería ser una solución temporal para pruebas, ya que dicho sistema de alimentación no está correctamente optimizado para ofrecer un rendimiento eficiente.

La batería seleccionada es una **Energizer L92-FR03** [32]. Tiene una capacidad nominal de 1200mAh con un peso de 7.6 gramos y un precio que ronda los 3€ No obstante, en una aplicación final, con un tamaño más reducido se podría hacer uso de otra tecnología óptima.

¹⁸ El formato estándar de pila **AAA** también conocido en su versión alcalina como LR03 tiene un tamaño de 10.5mm de diámetro y 44.5mm de longitud.

Memoria

4.6.6. Li-SOCI2

Las baterías primarias de cloruro de litio-tionilo son pilas no recargables compuestas por un ánodo de litio y un cátodo líquido compuesto por un carbono poroso relleno de cloruro de tionilo (SOCl₂). [33]

Actualmente, son las baterías no recargables con el más alto nivel de potencia por celda (3,6 V), la mayor densidad energética (de hasta 1280Wh/L [33]) y la menor ratio de autodescarga, lo cual les permite tener una vida de hasta 20 años.

Otra de las ventajas que presenta es su rango de temperatura en funcionamiento, que va desde 60 grados Celsius negativos hasta 85 grados positivos (-60 a 85°C), incluso con alguna batería especial que va hasta los 105°C. Esto les permite ser válidas para aplicaciones industriales donde estén sometidas a unas condiciones muy desfavorables de temperatura, humedad y estrés mecánico.

Su precio es algo elevado y ronda los 8€ para una batería de tamaño AA con una capacidad de 2,6Ah (precio de Farnell en mayo de 2018).

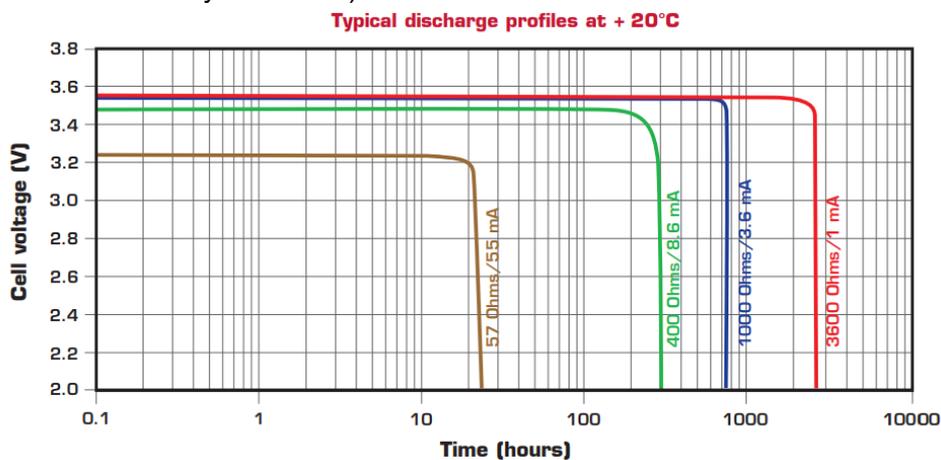


Figura 46 - Curva de descarga característica Saft LS14500EX (Saft Batteries) [34]

4.6.7. Supercondensadores

Los supercondensadores o *super-capacitors* no son baterías como tal. Se trata de dispositivos electroquímicos derivados de los condensadores, que son capaces de sustentar una densidad de energía inusualmente alta en comparación con los condensadores normales. Estamos hablando de capacidades del orden de mil veces mayor que la de un condensador electrolítico de alta capacidad.

Los supercondensadores, al igual que los condensadores, tienen la capacidad de cargarse y descargarse muy rápidamente, pudiendo entregar picos relativamente grandes de corrientes. Esto hace que normalmente sean usados junto con otras fuentes de energía, como en el caso del *Energy Harvesting*, donde los dispositivos son alimentados en modo dormido de bajo consumo energético por fuentes como la solar a través de pequeñas células fotovoltaicas (como las calculadoras), térmica a través de células *Peltier*, o incluso electromagnéticas gracias a la energía que viaja por el aire en forma de ondas. Normalmente estas fuentes de energía aportan una corriente muy pequeña que va cargando un supercondensador para que, cuando el dispositivo despierte y demande un gran pico de corriente para transmitir inalámbricamente, sea el supercondensador quien lo aporte.



Figura 47 - Supercondensador fino

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

4.6.8. Tabla comparativa

Vistas ya las tan variables tecnologías de químicas disponibles en el mercado, se realiza una tabla comparativa de todas ellas con el fin de facilitar la elección de la tecnología de almacenamiento ideal para el proyecto concreto. No obstante, esto son datos teóricos extraídos de diversas hojas de datos de baterías comerciales proporcionadas por los propios fabricantes bajo unas condiciones ideales que pueden variar de un fabricante a otro, y realmente conviene hacer un estudio de comparación práctica real con distintas baterías comerciales para ver cómo se comportan demandándoles el consumo energético requerido por la aplicación real y bajo condiciones no tan optimistas.

Tecnología	Recargable	Voltaje por celda	Temperatura de operación	Densidad energética	Autodescarga	Seguridad
LiCoO ₂ ¹⁹	500 veces	3.7 V (variable)	-20 a 60 °C	577 Wh/L	6%/mes	Altamente inflamable
LiFePo ₄	500 veces	3.2 V (estable)	-20 a 60 °C	220 Wh/L	3%/mes	Segura
NiCd	500 veces	1.2 V		50 Wh/L	30%/mes	Contamina
NiMH	1000 veces	1.2 V	-10 a 65 °C	150 Wh/L	20%/mes	
Li-FeS ₂	No recargable	1.5 V (estable)	-40 a 60 °C	520 Wh/L ²⁰	1%/año	Segura
Li-SOCl ₂	No recargable	3.6 V (muy estable)	-55 a 85 °C	830 Wh/L	1%/año	Segura

Tabla 5 - Comparativa baterías

Se concluye, por tanto, que pese a tener un precio bastante más alto que las demás tecnologías, resulta una tecnología muy interesante la de las químicas de las baterías primarias Li-FeS₂ y las Li-SoCl₂. Por su gran capacidad, su baja autodescarga y su voltaje nominal estable, que permite simplificar o eliminar la regulación de energía.

¹⁹ Datos extraídos de Panasonic NCR18650. Ver hoja de datos [45]

²⁰ Calculado según parámetros extraídos Energizer L92-FR03. = Densidad $\left(\frac{\text{Wh}}{\text{L}}\right) = \frac{\text{Capacidad(Ah)} \times \text{Vnominal (V)}}{\text{Volumen AAA(L)}}$

Memoria

5. Descripción detallada de la solución adoptada

En este capítulo se pasa a describir en detalle el desarrollo realizado, explicando las distintas partes que lo componen.

El desarrollo se ha dividido principalmente en 4 partes

- Desarrollo electrónico hardware del nodo sensor inalámbrico
- Programación firmware del microcontrolador
- Despliegue de red LoRaWAN
- Creación de aplicación de usuario para muestra de datos

Se define la topología general del sistema. Esto indicará la manera de actuar en el desarrollo del trabajo. Es de vital importancia realizar un diagrama general del sistema para definir las diferentes partes y actuar sobre él.

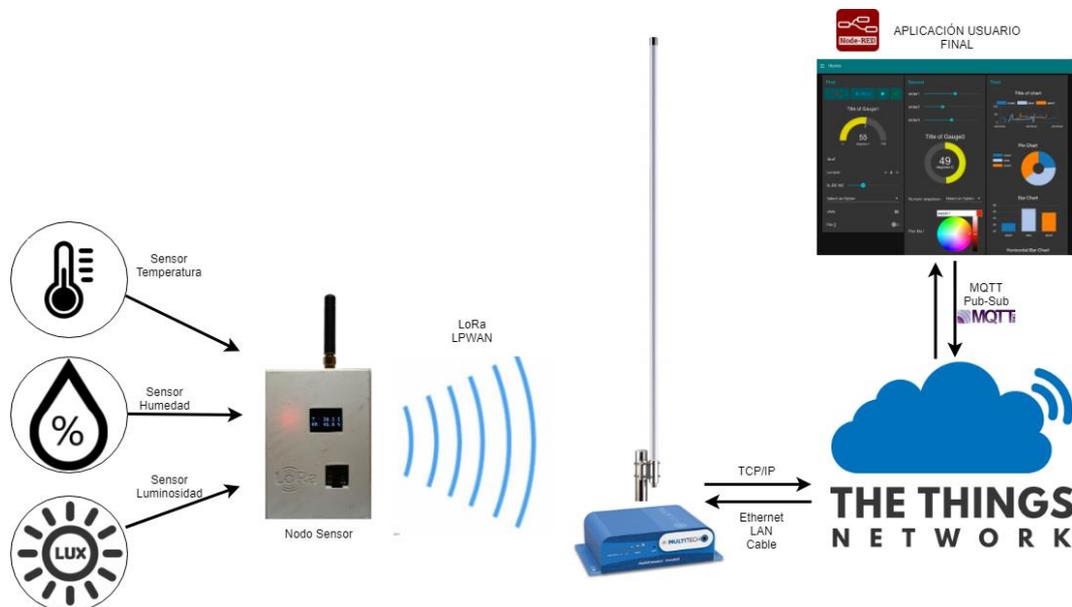


Figura 48 - Diagrama implementación nodo inalámbrico

En primer lugar, dispondremos del nodo sensor inalámbrico que integra los sensores de temperatura, humedad y luminosidad. El nodo tiene como misión recoger estos datos, procesarlos si es necesario, codificarlos y enviarlos mediante ondas de radio. El mensaje viajará por el aire a través de ondas electromagnéticas hasta llegar a la antena omnidireccional del punto de acceso. Hasta este momento, la información viaja inalámbricamente por el aire. El punto de acceso, también llamado *reenviador de paquetes* tiene la misión de transferir ese mensaje con modulación LoRa, en un mensaje TCP/IP que envía a una plataforma web a través de una conexión a Internet por cable LAN. Una vez el mensaje está en Internet, la plataforma LoRaWAN, lo procesa y decodifica y publica un *topic* o evento en un servidor MQTT. Este servicio MQTT es el encargado de llevar el mensaje final decodificado a la aplicación final de usuario.

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

En el caso de existir más de un nodo sensor inalámbrico, se define una tipología de red de estrella, donde todos los nodos reportan información a un punto de acceso central (*Figura 49*)

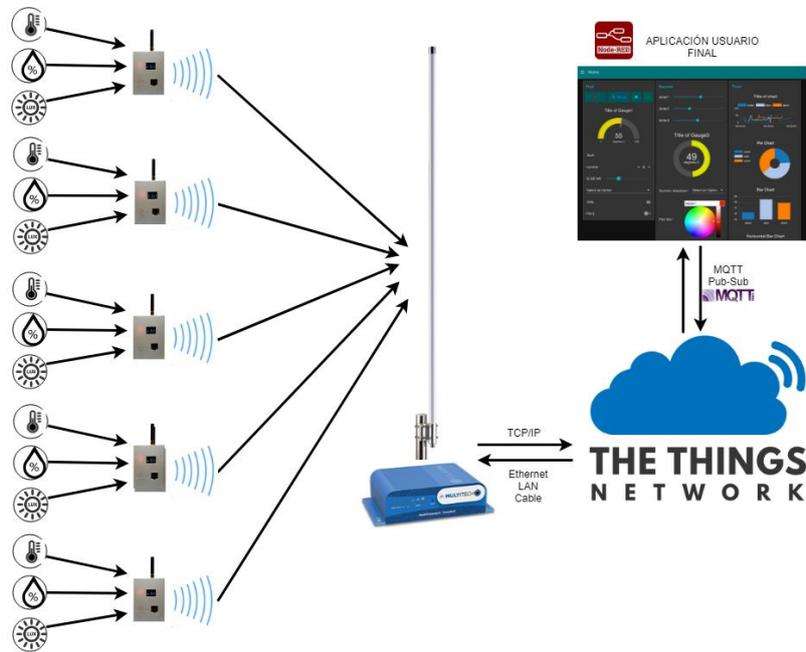


Figura 49 – Diagrama tipología con varios nodos

A continuación, se describe con detalle cada uno de los subsistemas del proyecto.

Memoria

5.1. Desarrollo del sistema sensor inalámbrico

El trabajo principal en el desarrollo del proyecto ha sido el desarrollo, tanto hardware, como software, del nodo inalámbrico en sí. Este desarrollo ha implicado el estudio de la placa de evaluación de STM y el módulo integrado de Murata, el diseño de esquemático para la tarjeta electrónica que integra los sensores y demás funcionalidades y la programación del microcontrolador STM32L0 de bajo consumo.

5.1.1. Desarrollo electrónico tarjeta sensores

El diseño electrónico hardware aborda uno de los mayores retos en el desarrollo del proyecto, debido a la naturaleza de este, pues resulta de vital importancia hacer un buen diseño electrónico tanto a nivel de esquema eléctrico como de *layout* de la PCB en un producto que se requiere que tenga un consumo mínimo. Es importante seleccionar bien los componentes y los pasivos para que no haya excesivas fugas de corrientes parásitas.

La tarjeta desarrollada ha sido concebida como un prototipo o banco de pruebas para testear diferentes tecnologías y conceptos y, por tanto, se trata de una versión no apta para una producción final.

Se ha desarrollado una tarjeta electrónica que se añade a la tarjeta de evaluación LRWAN1, con el formato estándar de *Shield* para Arduino Uno. En el ecosistema de Arduino, se conoce como *Shield* a una tarjeta de expansión que conecta con los pines hembra en Arduino Uno o Arduino Mega y sirve para añadirle una funcionalidad concreta (*Figura 50*). Se ha decidido seleccionar este factor de forma debido a su compatibilidad tanto con Arduino como con la tarjeta LRWAN1. Esto ha permitido el desarrollo del proyecto en ambos microcontroladores y plataformas. Es por ello por lo que, tanto en los diagramas y esquemas eléctricos, como en el montaje de la tarjeta de circuito impreso, se pueden observar diferencias con la versión montada en el prototipo final, debido a que hay componentes que finalmente no han sido montados por utilizarse para pruebas preliminares que se han decidido no implementar.

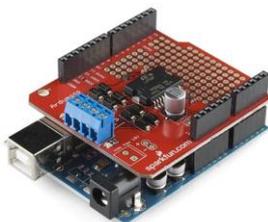


Figura 50 - Arduino Uno con Shield controlador de motores

El diseño de la tarjeta *LoRa Sensor Shield* se divide principalmente en 2 partes; en primer lugar, la integración con el transceptor RFM95 (*Sección 4.4.1*) y, en segundo lugar, la integración de los sensores de temperatura, humedad y luminosidad junto con los demás periféricos adicionales. Además, la tarjeta incluye funcionalidades adicionales como la monitorización de la batería a través del convertidor analógico a digital, los LEDs indicadores, también usados para depuración durante el desarrollo y un regulador de voltaje secundario y una pequeña pantalla OLED para depuración.

A continuación, se explican en detalle las distintas funcionalidades hardware y la integración en la tarjeta, comentando detalladamente cada bloque hardware funcional individualmente, centrándonos en los bloques más importantes, como son la integración del sensor de temperatura y humedad SHT31 de Sensirion y el de luminosidad.

El diseño electrónico se ha llevado a cabo usando la suite EDA de Cadence; Cadence Orcad para el diseño del esquema eléctrico y Cadence Allegro PCB Editor para el diseño del layout

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para
monitorización de patrimonio cultural

electrónico. Se trata de herramientas complejas ampliamente utilizadas en el sector profesional, que ofrecen una gran versatilidad en el diseño electrónico. Herramientas como la gestión de una *bill of materials* para los componentes electrónicos y la creación de huellas precisas para la PCB son de gran ayuda al ingeniero electrónico.

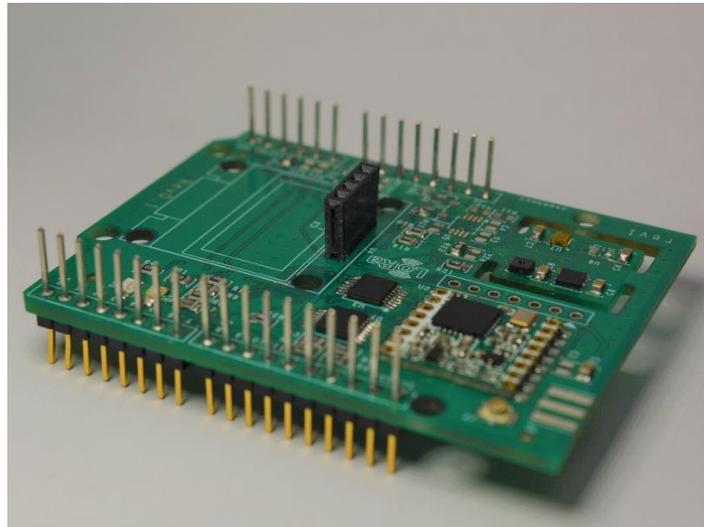


Figura 51 - LoRa Sensor Shield con RFM95



Figura 52 - Montaje final electrónica

Memoria

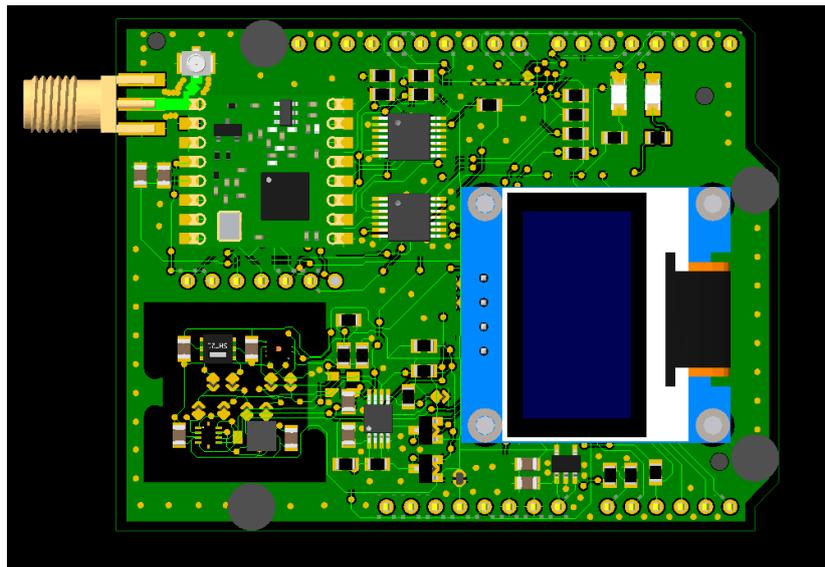


Figura 53 – Render 3D del layout extraído de Allegro PCB Editor

Se ha realizado un diagrama para identificar los distintos bloques funcionales del diseño de la electrónica con el objetivo de definir de una manera clara y visual los distintos módulos en los que se subdivide el diseño y las conexiones entre los mismos (Figura 54).

El diseño se divide en dos, la tarjeta de desarrollo LRWAN1 de *STMicroelectronics* y la tarjeta de los sensores diseñada desde cero para el propósito del proyecto, denominada *LoRa Sensor Shield*.

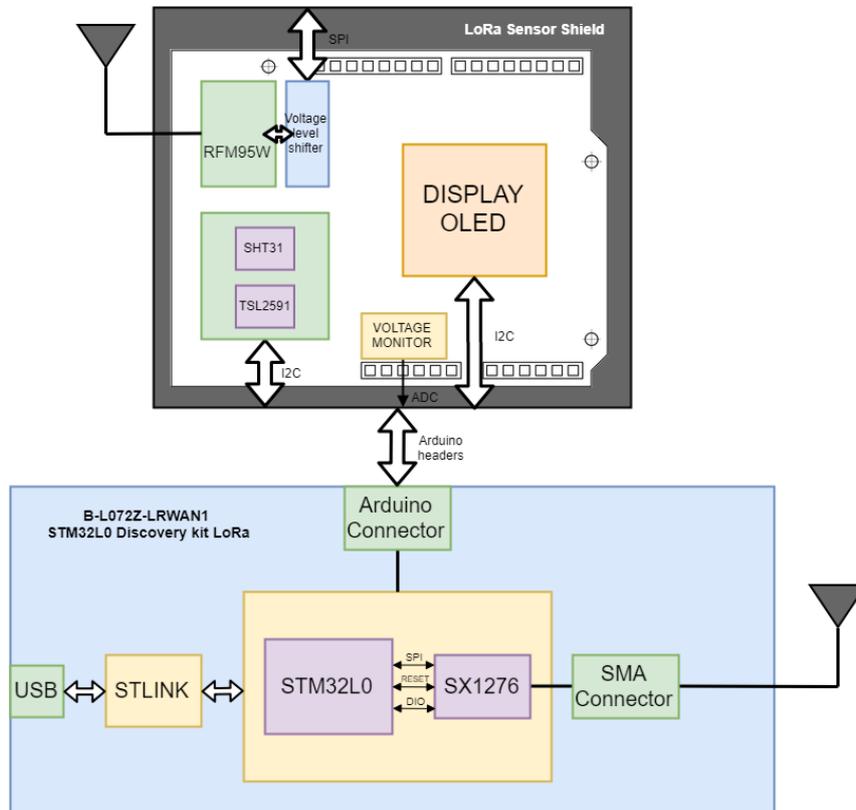


Figura 54 - Diagrama de bloques funcionales electrónica

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

5.1.1.1. Diseño del bus I2C

El bus I2C o I²C es un protocolo de comunicación serie síncrono desarrollado por *Phillips Semiconductors* (actualmente NXP) en 1982 y se utiliza principalmente como comunicación interna entre diferentes partes de un circuito (controlador y circuitos periféricos) ya que no funciona bien con grandes longitudes de cable. El protocolo I2C representa hoy día un estándar que resulta casi obligatorio conocer debido a la gran cantidad de periféricos, sensores, controladores y demás que hacen uso de este bus.

El I2C está diseñado como un bus maestro-esclavo, donde la comunicación, aunque bidireccional, es siempre inicializada por el dispositivo maestro que comienza enviando un byte compuesto con la dirección del dispositivo esclavo con el cual quiere comunicarse y 1 bit dedicado a indicar si se desea recibir datos del maestro o enviar. Al trabajar con direcciones de 7 bits permite hasta 112 nodos o dispositivos en el bus, ya que 16 direcciones están destinadas para fines especiales.

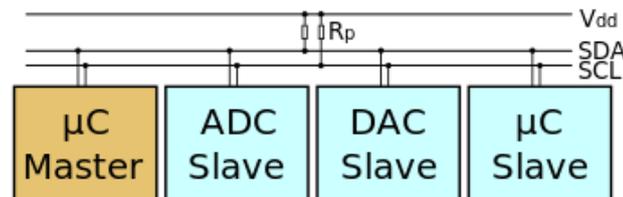


Figura 55 - Esquema eléctrico bus I2C (Wikipedia)

El bus I2C se compone de dos líneas, la línea SCL que es proporcionada por el dispositivo master y que provee de una señal de clock para sincronizar la comunicación con los dispositivos, y la línea SDA por donde viajan los datos.

Resulta de gran facilidad integrar una comunicación I2C ya que en la actualidad la mayoría de los dispositivos microcontroladores del mercado incluyen uno o varios periféricos I2C, por lo que el controlador hardware para procesar la comunicación ya viene dado y tan solo es necesario la programación de unos pocos parámetros para hacer uso del protocolo.

Respecto a la definición eléctrica del bus, es importante conocer que cualquier dispositivo conectado a un bus serie I2C ha de ser drenador o colector abierto por definición, esto significa que el dispositivo no debe ser capaz de imponer un nivel lógico en la línea y tan solo puede conducir esa línea a un nivel bajo. En caso de precisar un nivel alto, la línea será puesta en un estado de alta impedancia. Es por ello por lo que es necesario añadir unas resistencias *pull-up* en ambas líneas para proporcionar el estado alto al nivel de tensión correspondiente. El valor de estas resistencias dependerá de la velocidad de transmisión o frecuencia de reloj, que estará directamente relacionada con la carga capacitiva de las líneas y tensión a la que funcione el bus en particular y es una consideración importante el realizar un buen cálculo de estas resistencias. [35]

Una *strong pullup* (valor de resistencia pequeño) evita que el pin I2C sea llevado a nivel bajo, por lo que el V_{OL} definirá la menor tensión que podrá ser interpretada como un estado bajo. Con este valor y la corriente de drenador que es capaz de tragar el pin I2C se define el valor de resistencia *pullup* mínima como:

$$R_p(\min) = \frac{(V_{CC} - V_{OL}(\max))}{I_{OL}}$$

Para nuestro caso, y cogiendo los valores de la hoja de datos para el caso más desfavorable, que se trata del sensor TSL2591 tenemos que la resistencia de *pullup* mínima es:

Memoria

$$Rp(min) = \frac{3,3V - 0,6}{6mA} = 0,45k\Omega = 450\Omega$$

Ahora, el valor máximo vendrá limitado por la capacitancia del bus (C_b) definida por el *rise-time* que depende de la frecuencia de reloj a la que se vaya a utilizar el bus. Se define de esta manera:

$$Rp(max) = \frac{t_r}{0,8473 \times C_b}$$

Los parámetros del bus vienen definidos por la especificación del bus I2C [36] y para el caso de un bus I2C standard con una frecuencia de 100kHz, que proporciona unas velocidades de transmisión de hasta 100kbit/s tenemos que:

- $t_r(max) = 1000ns$
- $f_{SCL} = 100KHz$
- $C_b(max) = 400pF$

Con todo esto, tenemos que la resistencia de *pullup* máxima es:

$$Rp(max) = \frac{100ns}{0,8473 \times 400pF} = 2,950k\Omega$$

Idealmente la capacitancia del bus debe ser menor que las especificaciones máximas definidas, por lo que la formula debe dar una resistencia bastante conservadora para el peor de los casos, por lo que en la práctica esta resistencia puede ser incluso mayor sin afectar a la funcionalidad del bus.

No obstante, elegiremos una resistencia acorde con el cálculo realizado, y por tanto utilizaremos una resistencia entre el rango de 450 y 2950 Ohmios. Es importante recalcar que cuanto más baja sea esta resistencia de *pull-up* más corriente estaremos perdiendo, pero por el contrario podremos obtener velocidades de transmisión mayores. No obstante, en este proyecto se premia más la autonomía de la batería respecto a cualquier otra cosa, por lo que se decide seleccionar un valor lo más cercano al valor máximo para tener las menores perdidas en el bus. Seleccionamos el valor normalizado de **2,2k Ω** tanto para la línea de SCL como la de SDA (R13 y R14, *Figura 56*)

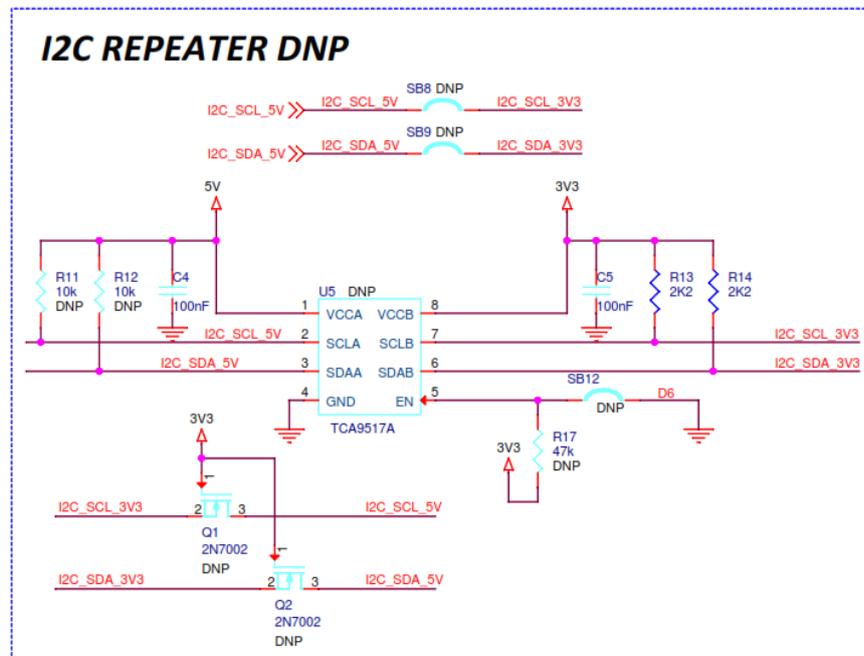


Figura 56 - Implementación resistencias pullup I2C

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

El esquema eléctrico de la implementación del bus se ve un poco más complicado. Esto es debido a que se han implementado dos funcionalidades para poder trabajar con dispositivos alimentados a 5V y dispositivos alimentados a 3,3V en el mismo bus. Para ello se consideran dos soluciones. La primera y más sencilla consiste en utilizar dos transistores MOSFET-N o BJT-NPN con su puerta o base conectada a nivel alto, su drenador conectado a 5V y su surtidor a 3.3V. No obstante, esta solución solo funciona si requerimos de una comunicación unidireccional y por tanto no nos sirve para el bus I2C. Se trata por tanto de una solución no válida y en la práctica no llegó a funcionar.

También se implementó en el mismo circuito la solución basada en un circuito integrado de Texas Instruments conocido como *Level-Shifting I2C Bus Repeater*, este tipo de integrados, además de proporcionar una translación de voltaje entre distintos niveles lógicos bidireccional y optimizada para buses I2C, sirve como repetidor para el caso en el que un mismo bus I2C se implemente en otra PCB conectada mediante un conector o aplicaciones donde del dispositivo master al esclavo haya una distancia superior a 5-10 centímetros. No obstante, no hizo falta montar este circuito tampoco debido a que para la aplicación final tan solo se utilizaron dispositivos funcionando a 3V3 en el mismo bus. Por ello solo se montan las resistencias de *pullup*.

Memoria

5.1.1.2. Sensor temperatura y humedad SHT31

Como ya se ha visto anteriormente, para el desarrollo de este proyecto se ha decidido usar un sensor calibrado en formato circuito integrado para ensamblaje en circuito impreso. Estos sensores funcionan con una interfaz o bus de comunicación digital, en este caso I2C, lo cual minimiza interferencias e imprecisiones generadas por el ruido eléctrico que afecta a una señal. Al usar un bus digital, los datos son codificados en binario, siendo más inmune al ruido.

La implementación de un sensor de este tipo en el diseño resulta extremadamente sencilla, y es que no requiere de apenas componentes externos para hacerlo funcionar. Además, el protocolo I2C es direccionable, lo que significa que es posible conectar varios sensores o periféricos al mismo bus y comunicarse con el que se requiera en cada momento.

En la (Figura 57) se muestra una numeración y descripción de cada uno de los pines del dispositivo. El sensor tiene un factor de forma muy reducido, en un *package* de montaje superficial de 8 pines. Dispone de un pin para resetear el dispositivo (nRESET, 6). La n al principio del nombre indica que utiliza una lógica negada, es decir, que el dispositivo se reseteará si llevamos ese pin a un estado lógico bajo, normalmente 0V o GND. Es por ello por lo que se recomienda usar una resistencia de *pull-up* (R21 en Figura 57) para seleccionar un estado lógico por defecto, que en el caso de una *pull-up* será un estado alto (3.3V). Este pin de reset se lleva a uno de los pines del conector de la shield con el propósito de poder resetear el sensor desde el microcontrolador en caso de fallo, aunque normalmente no será necesario.

Pin	Name	Comments
1	SDA	Serial data; input / output
2	ADDR	Address pin; input; connect to either logic high or low, do not leave floating
3	ALERT	Indicates alarm condition; output; must be left floating if unused
4	SCL	Serial clock; input / output
5	VDD	Supply voltage; input
6	nRESET	Reset pin active low; input; if not used it is recommend to be left floating
7	R	No electrical function; to be connected to VSS
8	VSS	Ground

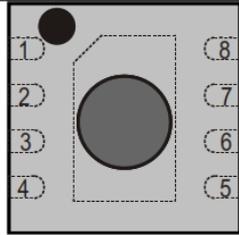


Figura 57 - Definición de pines SHT31

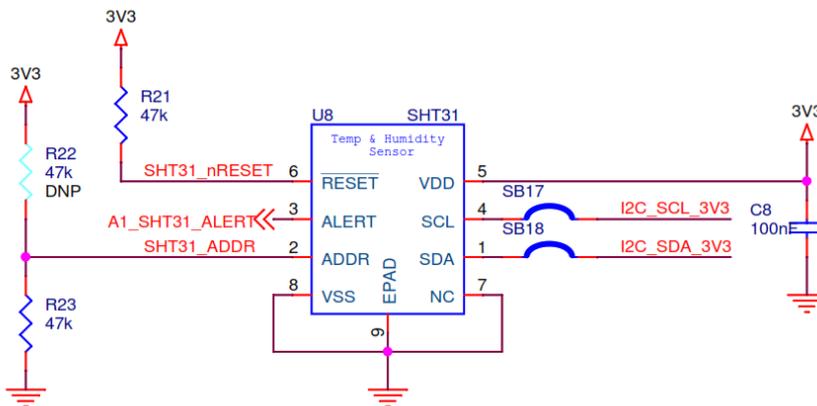


Figura 58 - Implementación SHT31 en el esquema eléctrico

El pin número 2 corresponde al pin de dirección, y es que este sensor permite seleccionar la dirección I2C del mismo seleccionando un estado alto o bajo en este pin (Figura 57). Esto permite que se puedan usar dos sensores independientes en el mismo bus. En este caso no es necesario, ya que solo se va a hacer uso de uno, no obstante, se decide dejar la opción de seleccionar la dirección posteriormente, haciendo uso de dos resistencias, una a nivel bajo y otra a nivel alto. Esto permite cambiar la dirección, si se monta la resistencia R22, que en cuyo caso será la 0x45 o la 0x44, si se monta la resistencia R23. Como se observa, se decide montar por defecto la resistencia

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

R23, dejando la otra como DNP²¹. Esto es una técnica que se utiliza ampliamente cuando se está haciendo un diseño de un prototipo, ya que permite al ingeniero, hacer cambios de configuración fácilmente, si es necesario.

SHT 3x-DIS	I2C Address in Hex. representation	Condition
I2C address A	0x44 (default)	ADDR (pin 2) connected to VSS
I2C address B	0x45	ADDR (pin 2) connected to VDD

Tabla 6 - Configuración dirección I2C

En tercer lugar, el pin número 3, nos generará una alerta configurada previamente según un umbral de temperatura o humedad deseado. Esta señal se pondrá a nivel alto cuando los valores estén fuera de un cierto rango previamente configurado y permite que solo se efectúen mediciones cuando los parámetros de temperatura o humedad se salgan de cierto límite, avisando al microcontrolador en ese momento mediante una interrupción externa. También puede ser usado para activar cierta circuitería como un indicador LED de aviso. Esta funcionalidad no será utilizada para nuestra aplicación, no obstante, este pin de alerta ha sido canalizado a un pin de interrupción externa tanto del STM32L0 como del ATMEGA328P, ya que con este diseño hardware, se pretende obtener la máxima flexibilidad posible.

Por último, se debe alimentar el circuito integrado por los pines 5 y 8, siendo esta alimentación (VCC) y masa (GND). Es importante no olvidar añadir un condensador de 100nF en paralelo con la alimentación del chip y masa. Este condensador recibe el nombre de *Condensador de desacoplo* y tiene la misión de aportar grandes picos de corriente en periodos de tiempo muy cortos, y es que, debido a la naturaleza de los condensadores, este permanece cargado de energía, y cuando los transistores CMOS internos del circuito integrado conmutan a grandes frecuencias, requieren de grandes picos de corriente, que son aportados por el condensador de desacoplo. Es por ello por lo que hay que situarlo en el circuito impreso lo más cerca posible del pin de alimentación del circuito integrado. Esto no solo es una recomendación del fabricante en su hoja de datos, si no que forma parte de las buenas prácticas de diseño de circuitos. El valor específico del condensador no suele ser importante, y 100nF suele ser lo habitual.

El sensor puede alimentarse en un rango de voltajes de entre 2,4V y 5,5V, pero debido a que toda la lógica de la tarjeta LRWAN1 funciona a 3.3V se ha decidido alimentar también los periféricos a 3V3.

Respecto a su colocación en el circuito impreso, de ahora en adelante llamado PCB²² para simplificar, el fabricante recomienda que se aisle lo máximo posible del resto de circuitería, ya que los componentes electrónicos se calientan (aunque sea poco), por el paso de corriente a través de ellos, es por ello por lo que, para obtener una medida lo más fiel posible y evitar que los demás chips del circuito puedan interferir, conviene generar un vacío o separación física. De no ser posible, también ayuda mucho, separar la zona de cobre para crear dos áreas aisladas. Debido a que la razón principal de creación de esta tarjeta es la de integrar los sensores, se ha decidido realizar una separación física mediante la creación de una apertura o corte de 1.5mm en el PCB, aislándolo así

²¹ DNP: Do Not Place

²² PCB: Printed Circuit Board

Memoria

(aunque no completamente) del resto de circuitería. (Figura 59). También se ha retirado la zona de cobre y se ha llevado la alimentación con pistas de 0,2mm de manera individual. Todo esto ha permitido crear una zona aislada térmicamente para la colocación de los 4 sensores. Además, como se verá posteriormente, esta apertura en el PCB se aprovecha posteriormente en el diseño de la carcasa de protección.

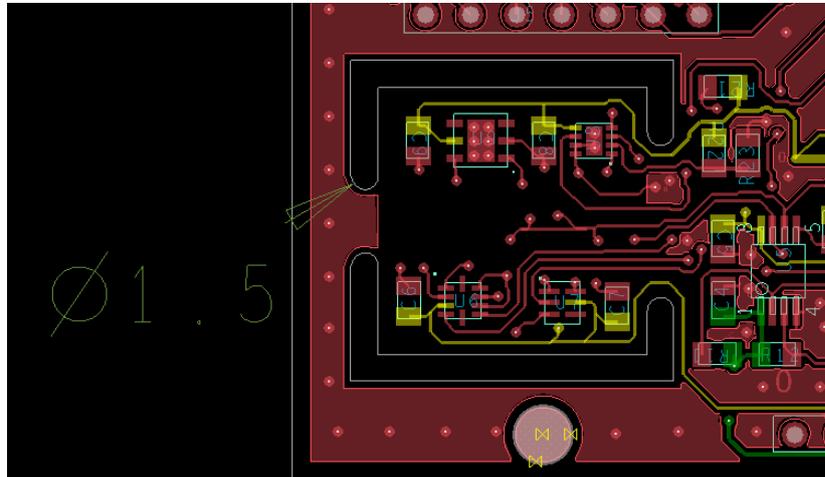


Figura 59 - Corte zona sensores PCB

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

5.1.1.3. Sensor luminosidad AMS TSL2591

El sensor de luminosidad también representa un reto a la hora de integrarlo en el diseño electrónico, ya que se ha de cuidar sobre todo el posicionamiento en la PCB para que este no se vea afectado por sombras o derivados que puedan afectar a su funcionamiento.

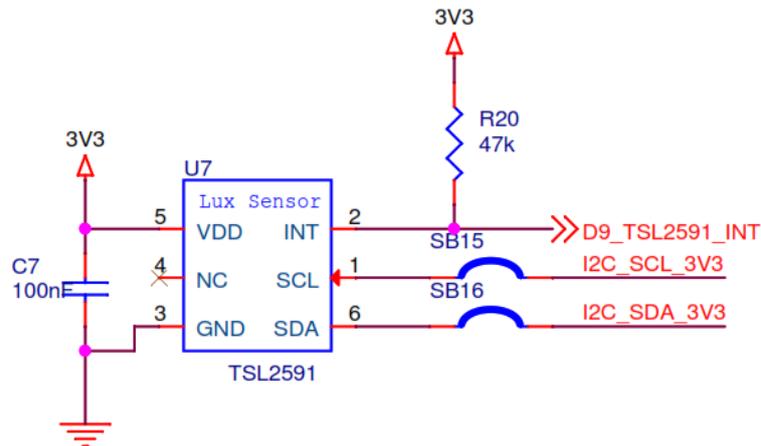


Figura 60 - Implementación TSL2591 en el esquema eléctrico

No obstante, su integración en el diseño también es muy sencilla, tan solo haciendo uso del bus I2C, pines 1 y 6 (Figura 60). Conviene explicar la función de los componentes SB15 y SB16. SB es la abreviatura de *Solder Bridge*, o puente de soldadura en castellano. Estos puentes están puestos en serie con la línea de I2C y se utilizan para activar o desactivar el dispositivo conectado al bus en caso de necesidad. Esto se repite en todos los dispositivos que están conectados al bus I2C y es muy común que se haga en prototipos o placas de evaluación para probar tecnologías. Cuando se desea conectar el dispositivo se suelda una pequeña bola de estaño, realizando un corto entre las dos líneas, si por el contrario se desea desconectar del circuito, se deja sin soldar, comportándose como un circuito abierto (Figura 61). En caso de un ensamble automático de la tarjeta, se suele montar una resistencia de 0 Ohmios.

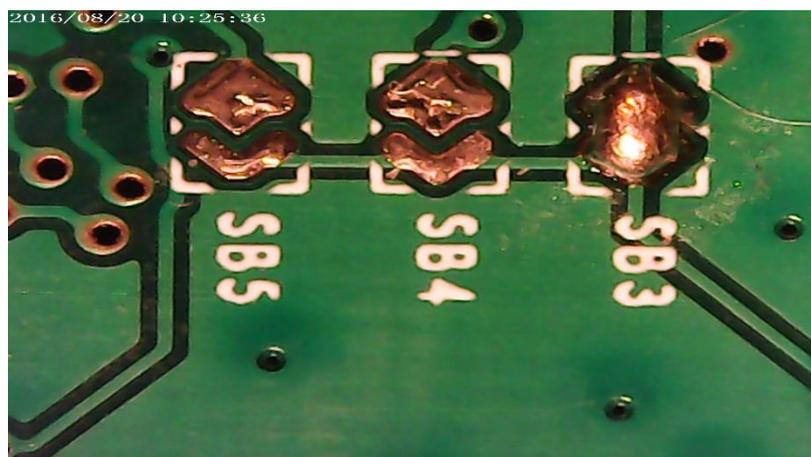


Figura 61 - Detalle Solder Bridge en prototipo real

El TSL2591 también dispone de una señal para un pin de interrupción, que permite configurar un umbral, como en el caso del sensor de temperatura.

Memoria

5.1.1.4. Transceptor RFM95

A pesar de que finalmente no se utiliza el módulo transceptor RFM95, se ha realizado la implementación de el mismo en la tarjeta shield de desarrollo, a modo de probar la tecnología. Este módulo hace uso de una interfaz SPI (*Serial Peripheral Interface*) para comunicarse con el microcontrolador. Este estándar de comunicación serie es usado principalmente para la transferencia de información entre circuitos integrados al igual que el I2C, pero, a diferencia de este, usa más pines y permite conseguir una velocidad de transmisión mayor. En la Figura 62 se observa su implementación. En el caso del SPI, a diferencia del bus I2C, se ha conectado un solo dispositivo, por lo que su implementación resulta tan sencilla como conectar cada señal del microcontrolador (master) a la señal con el mismo nombre del mismo nombre del transceptor (esclavo).

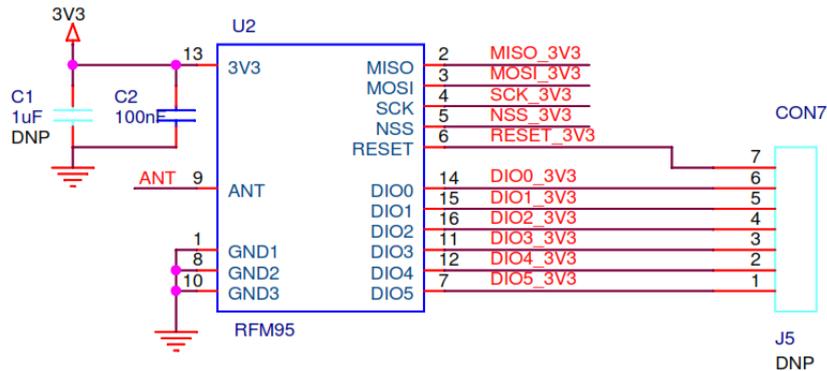


Figura 62 - Implementación modulo RF

Las señales DIO0 a DIO4 corresponden a señales de propósito general del módulo, que están directamente relacionadas con el estado de envío/recepción de datos y se usan como señales de interrupción en el microcontrolador para gestionar la máquina de estados del *stack* de LoRaWAN.

No obstante, todas estas señales no son conectadas directamente al microcontrolador, ya que en el caso del Arduino Uno, la lógica funciona a 5V y las señales del módulo transceptor a 3,3V, por lo que es necesario convertir estos niveles lógicos. Para ello se hace uso de un circuito integrado o puerta digital llamado *buffer triestado* y cuya función es la de permitir desactivar completamente una señal de un circuito. Para ello, el *buffer triestado* dispone de 3 estados lógicos en su salida; nivel alto, nivel bajo o nivel de alta impedancia (*Hi-Z*), que dependerán del estado lógico en sus entradas.

ENTRADAS		SALIDA
\overline{OE}	A	Y
LOW	HIGH	HIGH
LOW	LOW	LOW
HIGH	X	HI-Z

Tabla 7 - Tabla verdad Buffer Triestado

El chip usado es el **SN74LVC125A**, el cual es un buffer triestado cuádruple. De esta manera, y según su tabla de verdad (*Tabla 7*), la salida será nivel alto o bajo cuando el pin de control *Output Enable* esté a nivel bajo, y alta impedancia cuando este a nivel alto, independientemente de cuál sea el estado de la entrada o la salida. Esto permite desactivar circuitos que están conectados a un mismo bus, pero también convertir niveles lógicos digitales.

La forma de convertir los niveles es la siguiente: la entrada del buffer triestado se conecta a masa por lo que cuando el pin OE correspondiente se pone a nivel bajo, la salida es igual a la entrada, es decir, masa o nivel bajo. Por el contrario, cuando el pin OE se pone a nivel alto, la salida

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

es alta impedancia y, por tanto, el nivel lógico definido por la resistencia de *pull-up*, ya sea 3.3V o 5V en cada caso. Para las salidas que son salida del módulo y entradas para Arduino (MSIO) usaremos *pull-up* a 5V, y para las que representen una entrada al módulo usaremos resistencias *pull-up* a 3.3V.

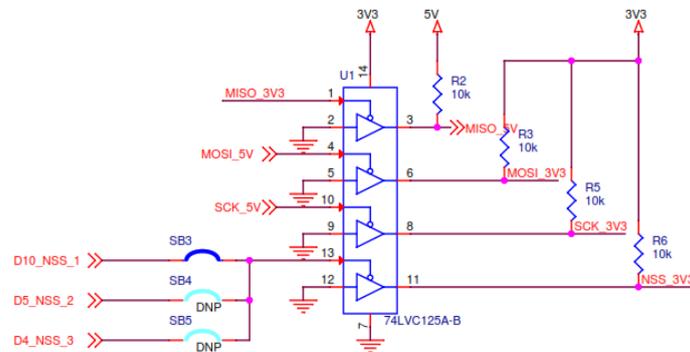


Figura 63 - Implementación conversor de niveles con SN74LVC125

Por último, resulta de vital importancia realizar un buen acoplamiento de la señal de la antena para garantizar que esta es lo más parecido a 50Ohm de impedancia de línea para que la antena funcione correctamente y tenga las menores reflexiones posibles. Para el desarrollo de esta tarjeta electrónica, se decide hacer uso de los dos conectores para antena externa más habituales, un conector SMA y un conector UFL, más pequeño. Finalmente, para las pruebas se hizo uso del conector UFL.

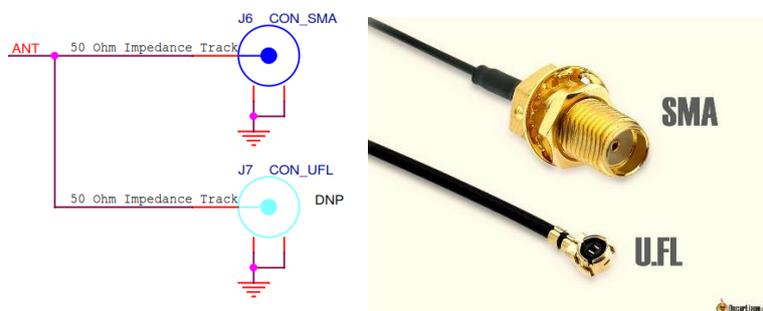


Figura 64 – Conector SMA y UFL y su implementación en el esquema

La impedancia de una línea en un PCB depende básicamente de 4 factores: el **ancho** de la pista en sí, la **altura de la pista** o altura del cobre (generalmente 35um) y del **ancho del dieléctrico** (normalmente sobre 1,5) y su **constante dieléctrica**. Hay otros parámetros que influyen como la distancia de la pista al plano de cobre o la frecuencia de la señal, no obstante, para hacer unos cálculos estimativos y para simplificar el proceso de diseño es suficiente, aunque cabe destacar que, en un futuro, para una aplicación real, es de vital importancia realizar un buen acoplamiento de impedancias en la antena para pasar los ensayos de certificación del producto según la normativa radioeléctrica (Directiva 2014/53/UE (Radio Equipment Directive, RE-D) [37]

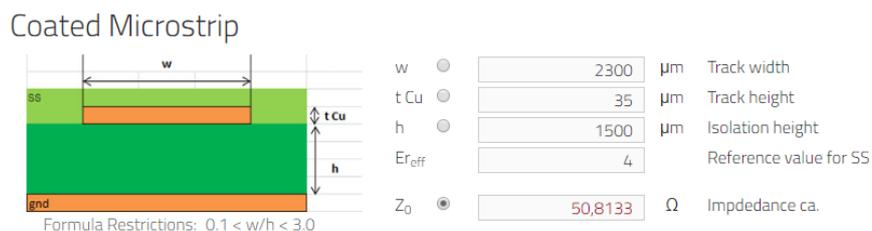


Figura 65 - Calculo de la impedancia característica de la línea

Memoria

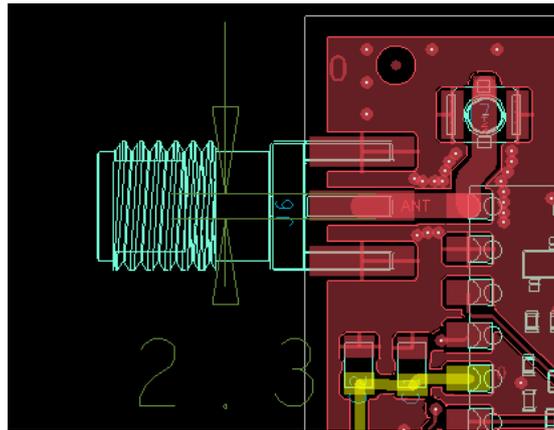


Figura 66 - Implementación en PCB de la antena

5.1.1.5. Circuito monitorización de la batería

Es de vital importancia conocer el estado de la batería para poder hacer una estimación de su autonomía y saber cuándo se debe sustituir con anterioridad. Esto permitirá que el dispositivo no deje de funcionar y tener el servicio siempre disponible. El dispositivo mandará el estado de carga de la batería junto con la información de los sensores.

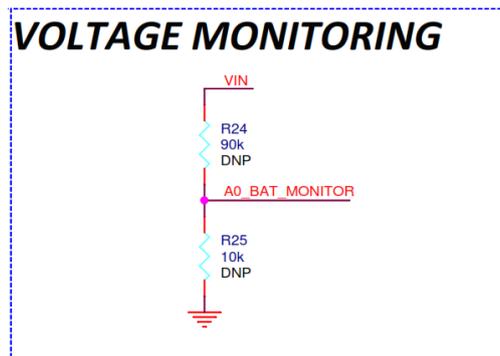


Figura 67 - Implementación monitorización batería

Un circuito para monitorizar el estado y carga de la batería puede ser tan complejo como el caso de un *Battery Fuel Gauge*²³ o tan simple como un divisor resistivo conectado al ADC del microcontrolador. En este caso, optamos por el método simple del divisor resistivo. Para ello, se conecta un divisor resistivo entre el voltaje de entrada conectado directamente a la batería (VIN) y masa, cuyo centro irá a un pin analógico del Arduino o LRWAN1, en su defecto.

Teniendo en cuenta el nivel de tensión máxima que tenemos en las baterías completamente cargadas si usamos 3 baterías en serie de 1,5V ($1,5V \times 3 = 4,5V$), y que las entradas de la LRWAN1 permiten un voltaje analógico máximo de 3,0V, se utiliza un divisor resistivo para reducir la tensión a un 10%.

Es decir, si tenemos un nivel de potencial en VIN de 4,5V, el voltaje que medirá el ADC será:

$$V_{ADC} = R_{R25} \times \frac{V_{IN}}{R_{R24} + R_{R25}} = 10K \times \frac{4,5V}{10K + 90K} = 4,5V \times 0,1 = 0,45V$$

²³ Un **Battery Fuel Gauge** es un circuito integrado capaz de estimar la batería restante y el porcentaje de carga midiendo el voltaje y corriente proporcionada al circuito basándose en complejos algoritmos de aprendizaje. Estos integrados son los responsables de indicarnos el porcentaje de carga en nuestros teléfonos inteligentes y resultan mucho más precisos y certeros que el mecanismo tradicional de medición de voltaje.

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

5.1.1.6. Regulador lineal 3,3V

Tanto el Arduino como la tarjeta de evaluación LRWAN1 integran reguladores de voltaje que proporcionan una tensión estable de 3,3V independientemente de la tensión de las baterías. No obstante, la corriente máxima que pueden ofrecer estos reguladores es un poco justa si se le conectan varios periféricos que demanden de corriente en este bus. Para ello se opta por implementar un pequeño regulador lineal que cogerá la tensión de la batería y la regulará a un voltaje de 3.3V. Este módulo del circuito se ha llamado de *backup* ya que, sobre el papel, no será necesario hacer uso de este regulador, pero se implementa por si en un futuro fuese necesario. Cabe destacar que este regulador no será montado, por ello su atributo de DNP (*Figura 68*).

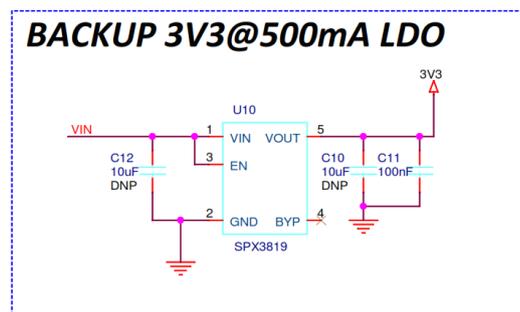


Figura 68 - Implementación LDO 3.3V en el esquema eléctrico

El circuito integrado que se implementa es el SPX3819 de *Exar Corporation* (ahora *Max Linear*) [38]. Se trata de un LDO o regulador lineal de baja caída de tensión, para aplicaciones portables, con una tensión de caída de 340mV y una corriente máxima regulada de 500mA. Existe en varias versiones según su voltaje de salida regulado y huella en la PCB, el que se implementa es el **SPX3819M5-L-3-3/TR** (paquete SOT23-5).

5.1.2. Murata LoRaWAN CMWX1ZZABZ

CMWX1ZZABZ es el módulo que viene integrado en la placa de desarrollo LRWAN1 de STMicroelectronics. Este SoM (*System on Module*) integra un microcontrolador STM32L0 de bajo consumo con un transceptor LoRa. El módulo viene prácticamente preparado para trabajar con él en la tarjeta de desarrollo, la cual se puede programar por USB gracias al STLINK integrado.

Tan solo se han tenido que soldar dos *solder brige* o puentes en la placa de desarrollo para conectar los pines I2C del chip de Murata con los conectores hembra compatibles con Arduino para que el bus I2C quede accesible a la placa de los sensores. En concreto se han soldado el SB11 y SB12.

Para más información sobre la tarjeta de evaluación, referirse a los anexos correspondientes sobre la hoja de características y el esquemático de la tarjeta.

Memoria

5.1.3. Programación

La programación del *firmware* ha sido el segundo reto del desarrollo del proyecto, ya que ha implicado juntar distintos submódulos en una aplicación capaz de interactuar entre ellos. Además, el código desarrollado incluye el *stack* de LoRaWAN que es una máquina de estados encargada de codificar los mensajes y generar el *payload* junto con la cabecera de mensaje correspondiente y es la responsable de comunicarse con el transceptor de *Semtech* mediante SPI para modular y transmitir correctamente el mensaje. Esta parte del código viene implementada en una serie de bibliotecas desarrolladas por Semtech junto con STMicroelectronics, con el objeto de crear una capa de abstracción que permita al usuario centrarse en su aplicación.

Al igual que con el diseño electrónico, con el diseño software también conviene definir un diagrama o arquitectura software con el objetivo de separar el código en distintos submódulos que permitan simplificar la aplicación final por medio de funciones de abstracción. A este tipo de diseño software se le denomina programación modular y facilita enormemente la depuración del código.

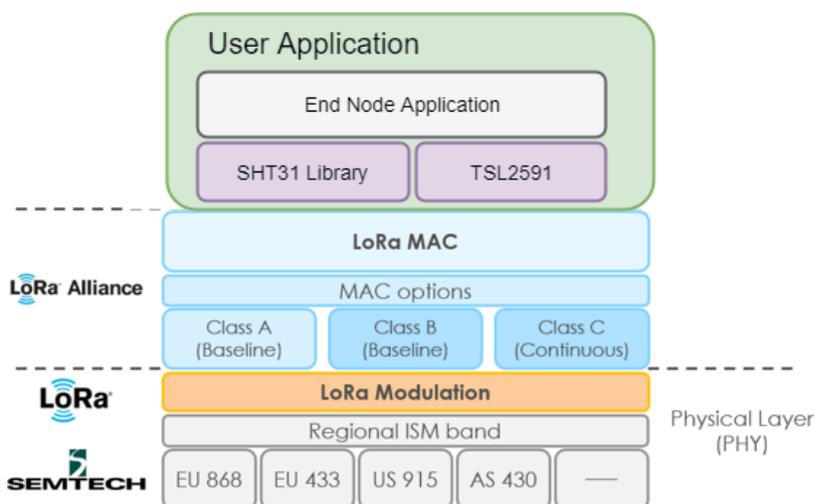


Figura 69 - Arquitectura del firmware

La programación a desarrollar será el bloque verde llamado *User Application*. En este bloque realizaremos llamadas a funciones de más bajo nivel que nos permitan hacer uso del stack LoRaWAN. También se desarrollarán los bloques morados de las bibliotecas de los sensores. Esto permitirá depurar la comunicación de los sensores por separado y una vez obtenida una versión estable, incluir dichas funciones a la aplicación de usuario mediante llamadas a la biblioteca.

A continuación, se describen en profundidad las funciones creadas en los distintos bloques o módulos. Para consultar el código fuente completo de las distintas aplicaciones conviene referirse al **Anexo 1** del presente proyecto.

5.1.3.1. Biblioteca sensor SHT31

La biblioteca del sensor SHT31 incluye las funciones necesarias para la lectura de los valores de temperatura y humedad del integrado de Sensirion, así como funciones para realizar la configuración de este. Este integrado hace uso del protocolo I2C, pero tan solo hará falta pasarle un puntero de memoria con la configuración del bus y la dirección, para realizar llamadas a las funciones de configuración y lectura de manera transparente.

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

La biblioteca ha sido diseñada en C++ y orientada a objetos, esto permite definir varios sensores cada uno con una dirección diferente si fuera necesario y llamarlos en la aplicación por separado.

Se han creado dos archivos, el .h o cabecera donde se encuentran las definiciones de las funciones públicas de alto nivel accesibles al usuario, y el .cpp, que incluye la declaración de dichas funciones.

Se deberá crear un objeto de la clase SHT31 e inicializar con las funciones SHT31. `begin(&i2cHandler, i2caddr = SHT31_DEFAULT_ADDR)`, si no se especifica la dirección I2C se utilizará la que viene por defecto (0x44). Las funciones que se estarán disponibles posteriormente son las siguientes:

- `float readTemperature(void)` que devuelve un valor en coma flotante con la temperatura en grados centígrados
- `float readHumidity(void)` que devuelve un valor en coma flotante con la humedad
- `uint16_t readStatus(void)` que devuelve un entero codificando el estado del dispositivo por si hubiera algún error
- `void reset(void)` para resetear el sensor
- `void heater(bool)` para activar o desactivar el calentador

Normalmente solo se utilizarán las de lectura de temperatura y humedad.

5.1.3.2. Biblioteca sensor TSL2591

De manera análoga al sensor de temperatura y humedad, se construye una biblioteca con las funciones necesarias para el control del sensor de luminosidad. Cabe destacar que este sensor es más complejo de utilizar, debido a la multitud de registros y configuraciones que incluye. No obstante, se facilita en gran medida su utilización gracias a la biblioteca desarrollada.

De igual manera, se ha de instanciar un objeto de la clase TSL2581 e iniciar mediante el método `bool begin(&i2cHandler, i2caddr = TSL2591_DEFAULT_ADDR)` que devolverá un valor verdadero si se ha iniciado correctamente. Esta función también configurará el sensor con los parámetros por defecto. Posteriormente estarán disponibles los siguientes métodos o funciones:

- `uint32_t calculateLux(uint16_t ch0, uint16_t ch1)`
- `void setGain(tsl2591Gain_t gain);`
- `void setTiming(tsl2591IntegrationTime_t integration);`
- `uint16_t getLuminosity(uint8_t channel);`
- `uint32_t getFullLuminosity();`
- `uint32_t getLux ();`

De todas estas, en un funcionamiento normal por defecto, tan solo haremos uso de la última (`TSL2591.getLux()`) para obtener un entero sin signo de 32 bits con la cantidad de lux. No obstante, se dispone de funciones para establecer la ganancia del fototransistor, así como del tiempo de integración, que puede que sean utilizadas en un futuro.

Como se ve, el concepto de desarrollo modular ayuda a una escalabilidad futura del producto ya que se implementan funciones que por el momento no son de utilidad pero que están disponibles en caso de necesidad futura. Esto permitirá mejorar el funcionamiento en un futuro de forma sencilla debido a que estas funciones ya están implementadas.

Memoria

5.1.3.3. Prueba sensores

Se ha desarrollado una aplicación para la prueba de los sensores. Esta aplicación solo incluye las bibliotecas y un bucle con la llamada a las funciones y la asignación de los valores a unas variables. Posteriormente estas variables se pueden imprimir en una consola serie o graficar usando un entorno de depuración como el que nos ofrece STMStudio (*Figura 70*).

Para la configuración del proyecto de Keil uVision, se usa la herramienta STM32CubeMX, la cual permite configurar parámetros como la velocidad de reloj, los periféricos o la asignación de pines de una manera gráfica y sencilla. El programa nos genera un proyecto completo con todas las configuraciones del microcontrolador para empezar a desarrollar la aplicación de usuario utilizando la biblioteca HAL. Esto permite al desarrollador ahorrar mucho tiempo.



Figura 70 - Captura gráficas STMStudio

Una vez se tienen los sensores funcionando satisfactoriamente con las bibliotecas, se pasa a implementar la aplicación final.

5.1.3.4. Aplicación End-Node final

Por último, una vez se ha probado el correcto funcionamiento de las bibliotecas de los sensores, se programa la aplicación final que integrará la arquitectura descrita en la *Figura 69*.

Se parte del ejemplo proporcionado por STMicroelectronics para el manejo de la capa de abstracción que integra el *stack* de LoRaWAN añadiendo las lecturas de los sensores.

En primer lugar, se inicia y configura el *stack* y se configura el modo de baja energía a utilizar (modo *standby*), posteriormente se inicializan los sensores haciendo una llamada a las funciones de *begin* y se invoca la función *LORA_Join()* para realizar la activación del nodo en el servidor LoRaWAN. En el modo depuración se encenderá un LED verde cuando el dispositivo esté correctamente identificado en la red.

Una vez identificado comienza el modo de transmisión normal en el bucle principal, en el cual se realiza una medida de los sensores, se mide el voltaje de la batería, se compone el paquete o mensaje *payload* y posteriormente se invoca la función *LORA_Send()* para enviar el paquete

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

inalámbicamente. Todo esto se realiza a través de funciones de *callback* o eventos de interrupción, de tal manera que en el bucle *while* no se escribe nada más que la llamada al modo low power.

El paquete se codifica según la definición de **Cayenne LPP** (*Low power payload*). Se trata de un estándar de codificación para dispositivos IoT que trata de optimizar al máximo los paquetes para que ocupen la menor cantidad de *bytes* posible. Define varios tipos de sensores, cada uno con un ID y una cantidad de bytes asociada según el tipo de sensor [39].

La temperatura se multiplica por 10 para obtener una resolución de 0,1°C y la humedad se multiplica por 2 para obtener una resolución de 0,5% de humedad relativa.

Adicionalmente, es necesario configurar varios parámetros que afectan a la configuración LoRaWAN y la identificación de la red, estos parámetros se configuran en un archivo de cabecera denominado *Comissioning.h*. La configuración introducida es la siguiente:

- **Método de activación:** *Over The Air Activation* (OTAA)
- **Identificación del dispositivo** (Device EUI): dirección física del dispositivo.
- **Identificación de la aplicación** (App EUI): proporcionada por la red
- **Llave de aplicación encriptada:** permite acceder a la aplicación de red y encriptar los paquetes
- **Clase del dispositivo:** Clase A
- **Tasa de datos:** DR5, que corresponde a un factor de propagación de 7 y un ancho de banda de 125kHz.

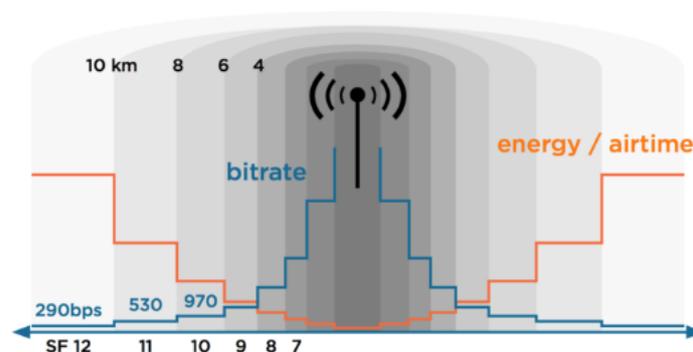


Figura 71 - Relación tasa de propagación con consumo, velocidad y alcance

Se intenta buscar un compromiso entre consumo energético y alcance, por eso se selecciona un factor de propagación de 7.

Memoria

5.2. Pasarela LoRaWAN

Para poder hacer uso de la tecnología LoRaWAN se ha tenido que instalar una pasarela LoRaWAN o *Gateway* como punto de acceso para la red de malla de LoRaWAN. Esta pasarela cuenta con un transceptor LoRa y una conexión a Internet mediante cable LAN o WiFi y se encarga de recoger los paquetes LoRa enviados por los nodos, demodularlos, decodificarlos y enviarlos a la plataforma web a través de Internet.

Un Gateway, pasarela, concentrador o puerta de enlace es, por tanto, un dispositivo que actúa de interfaz o intermediario de conexión entre aparatos o dispositivos. Puede ser entendido como un *router*, ya que éste hace de interfaz entre una red local en el hogar u oficina y la red global, proporcionándonos conexión a Internet. En el ámbito de las LPWAN y en concreto de LoRaWAN, un Gateway es el dispositivo que dota de conexión a Internet a los end-nodes, haciendo de interfaz entre la malla física de LoRa y el servidor conocido como LoRaWAN Network, que será el responsable de decodificar los mensajes.

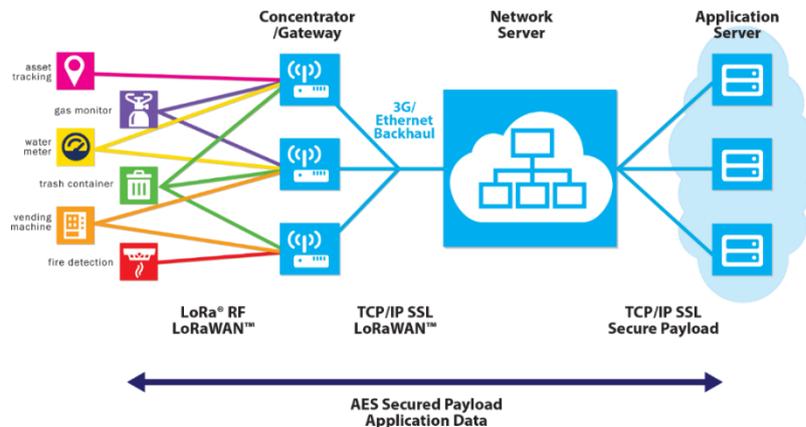


Figura 72 - Diagrama básico LoRaWAN (LoRa Alliance)

En LoRa existen básicamente dos tipos de puntos de acceso, los multibanda o multicanal y los que solamente escuchan en un canal. La diferencia reside en el número de canales que pueden escuchar simultáneamente, los dispositivos multicanal disponen de transceptores mucho más potentes que les permiten escuchar en hasta 7 canales a la vez, y, por tanto, capturar mensajes que sean enviados en cualquiera de estos 7 canales. Esto representa la situación ideal y es el caso idóneo ya que se pueden tener muchos más nodos conectados a ese mismo Gateway sin que estos se interfieran. En el caso de usar *gateways* mono-canal, la cantidad de nodos que se pueden instalar por cada *gateway* se reduce debido a que el concentrador solo es capaz de escuchar en un canal simultáneamente y si dos nodos transmiten en el mismo canal al mismo tiempo, ambos mensajes se pierden. El stack de LoRaWAN está implementado de tal manera que los canales en los cuales transmite el nodo se van rotando y estadísticamente, con 7 canales, es muy difícil que haya interferencias, pero con tan solo un canal esto se reduce drásticamente y la posibilidad de que haya interferencias aumenta enormemente. Por el contrario, los concentradores multi-canal se mueven en un rango de precios mucho más alto que las pasarelas mono-canal.

Lo ideal es instalar el punto de acceso en una ubicación alta en el exterior como por ejemplo un tejado, y con una antena omnidireccional grande, esto permite la máxima cobertura y alcance. No obstante, para la realización de este proyecto prototipo, el punto de acceso se instala en un ambiente *indoor*, es decir, dentro de un edificio (Figura 76). Esta no es la situación ideal y, como se verá posteriormente, afecta notablemente al rango de alcance de la red, pero permite realizar pruebas de desarrollo de manera más que correcta.

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

Actualmente están empezando a surgir muchas soluciones de pasarelas o puntos de acceso para suplir la gran demanda que está despertando la tecnología LoRaWAN. Prácticamente todas están basadas en los mismos transceptores de Semtech, donde principalmente existen dos; el SX1276 y el SX1308. Y precisamente, en la diferencia entre estos dos reside la diferencia entre los dos tipos de pasarela que podemos clasificar.

A continuación, analizamos brevemente dos opciones que representan cada una un tipo de pasarela distinto.

5.2.1. Dragino LG01-S

Por un lado, los puntos de acceso mono-canal o *single-channel gateway* ofrecen una sola banda de escucha simultánea. Están basados normalmente en el integrado SX1276 que es el mismo que se utiliza para los nodos y, por tanto, su precio resulta muy asequible. No obstante, no cumplen 100% con las especificaciones de LoRaWAN al no permitir la escucha simultánea en los 7 canales que define.



Figura 73 - Gateway Dragino LG01-S

El Dragino LG01-S es un punto de acceso LoRaWAN mono-canal basado en el módulo RFM95, que a su vez integra el SX1276 de Semtech. Corre bajo un sistema operativo Linux OpenWRT y un microcontrolador ATMEGA328P con el stack de LoRaWAN. Tiene un precio que oscila entre los **30 y 50 euros**.

Memoria

5.2.2. Multitech Conduit

El MultiTech MultiConnect Conduit representa una solución profesional para el despliegue de una red LoRaWAN. Se trata de un punto de acceso multicanal basado en el transceptor SX1308 que permite hasta 7 canales de escucha simultáneos. Esto significa que se pueden añadir muchos más nodos a la red sin que los mensajes de estos sufran interferencias y colisiones. Corre un sistema operativo Linux, pero tiene una potencia de procesamiento mucho mayor. Además, dispone de WiFi, una versión con soporte para 3G/LTE y una bahía adicional de expansión.

Tiene un precio que supera los **600 euros**.



Figura 74 - Multitech Conduit

5.2.3. Instalación del Multitech Conduit

El MultiConnect® Conduit™ de Multitech es la pasarela industrial más configurable y escalable para aplicaciones IoT industriales. Permite conectarse a Internet haciendo uso de tecnología inalámbrica celular como 2G, 3G o 4G-LTE, WiFi o Ethernet por cable. Además, hay disponibles una gran cantidad de accesorios de expansión denominados mCard™. Se trata de pequeñas tarjetas electrónicas que proveen de conexión LoRa, Ethernet, GPIO o Serie, lo cual permite una gran flexibilidad de utilización. Se pueden instalar hasta dos tarjetas de expansión en cada concentrador.

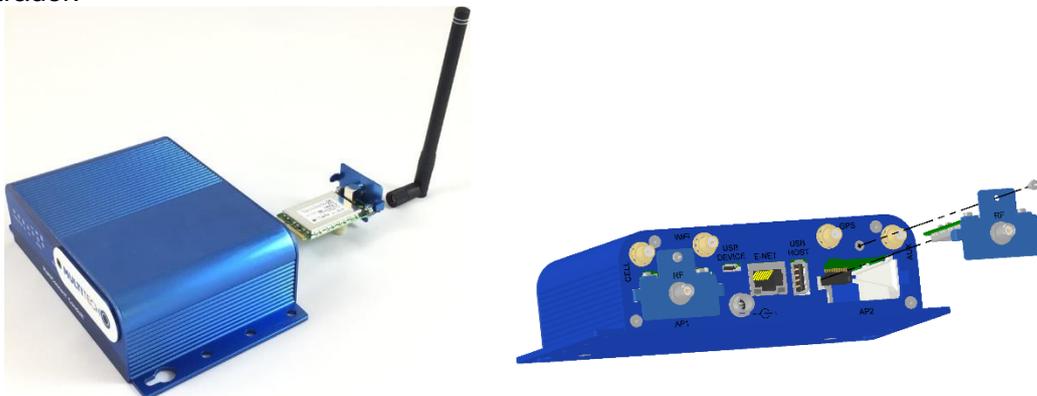


Figura 75 - Multitech Conduit con transceptor LoRa

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

En nuestro caso, tan solo ha hecho falta instalar la tarjeta transceptora *MTAC-LORA-H-868* preparada para una frecuencia de 868MHz. Esta tarjeta tiene una interfaz estándar MiniPCI-express y hace conexión directamente con uno de los puertos del *router Multitech* tal y como se muestra en la *Figura 75*. Adicionalmente, se provee de una conexión a Internet mediante un cable Ethernet LAN convencional conectado a una de las bocas del *router* proveedor de acceso a Internet. Este router le asignará una IP dinámica de red. (*Figura 76*)



Figura 76 - MultiConnect Conduit instalado y conectado al router

El MultiConnect® Conduit™ dispone de un sistema operativo basado en Linux, llamado mLinux, en concreto, este proyecto se ha realizado con su versión 3.2.0. Para acceder a él se usa el protocolo SSH para conectarse a una consola de comandos con la cual interactuar con el sistema operativo y poder instalar las utilidades necesarias para conectarlo con la plataforma de red en la nube LoRaWAN. Cabe destacar que Conduit también puede alojar una plataforma de red LoRaWAN privada, pero para simplificar el desarrollo, utilizaremos una plataforma pública gratuita ya existente como se explicará posteriormente.

Para acceder al Conduit tenemos que conocer la IP que le ha asignado el router para poder conectarnos a él mediante SSH. Una IP es una dirección de red única para identificar a cada uno de los dispositivos conectados. Consta de 4 números separados por puntos simples. En el caso de una red local los dispositivos tendrán una dirección de la forma 192.168.1.xx. Para descubrir la dirección que le ha asignado dinámicamente el router utilizaremos un programa conocido como *sniffer* que hace un escaneo de todos los dispositivos existentes en la red local. (*Figura 77*) Cabe destacar que el ordenador que utilizemos debe estar conectado a la misma red local que el dispositivo Conduit.

Memoria

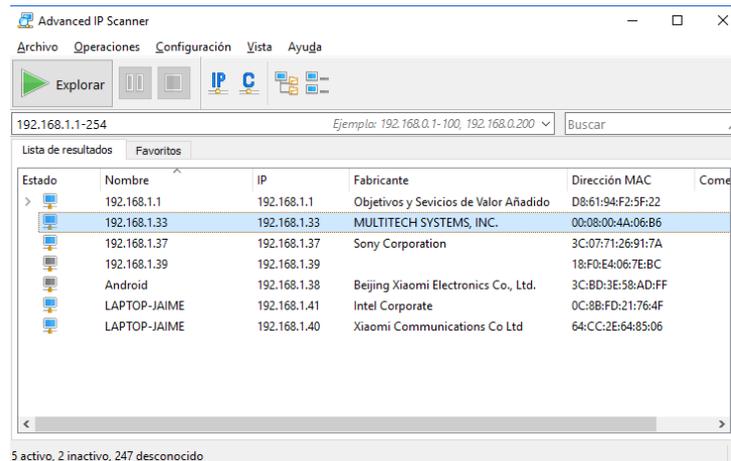


Figura 77 - Programa sniffer escaneo de red

Una vez obtenida la dirección IP del Conduit utilizaremos el programa PuTTY sobre Windows para establecer una conexión SSH con él, nos logharemos como usuarios con privilegios de administrador y haremos la descarga e instalación de un programa llamado *packet-forwarder*, el cual se encargará de recoger los paquetes LoRa que lleguen por cualquiera de los 7 canales de escucha a la plataforma de red LoRaWAN.

El script instalador que lanzaremos nos pregunta varios parámetros de red como la dirección del servidor LoRaWAN, la frecuencia utilizada (868MHz en este caso), la localización o la hora. El script ha sido desarrollado por *kersing* y puede ser descargado desde el repositorio original:

<https://github.com/kersing/multitech-installer/>

También pedirá que se introduzcan los parámetros de configuración del Gateway en el servidor como el nombre, y la clave única identificativa. Estos parámetros son proporcionados por la plataforma LoRaWAN proveedora donde previamente tendremos que haber dado de alta el dispositivo Gateway.

Para más detalle sobre la instalación y configuración del Conduit visitar:

<https://www.thethingsnetwork.org/docs/gateways/multitech/mlinux.html>

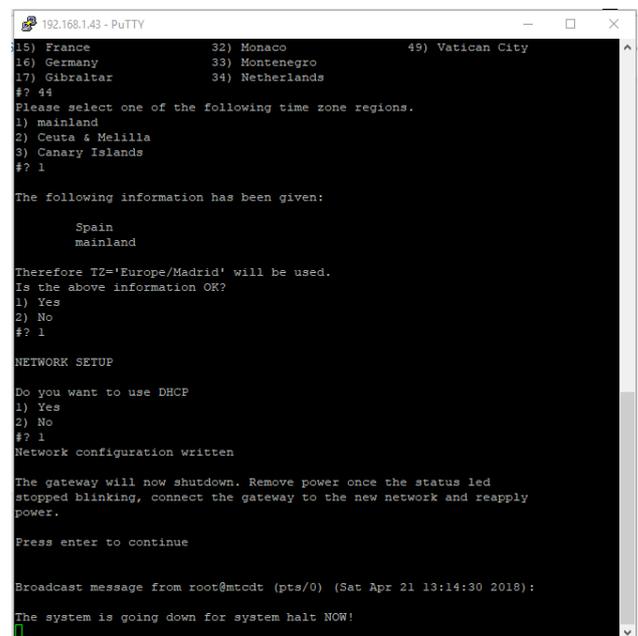


Figura 78 - Configuración packet forwarder

5.3. Plataforma cloud recolectora de datos (TTN)

La plataforma recolectora de datos en la nube es un servidor web donde se implementa la capa de red de LoRaWAN. Esta capa de red está definida en las especificaciones de LoRaWAN y permite crearse una red privada basada en las especificaciones. No obstante, debido a la naturaleza del proyecto, a que simplifica bastante el desarrollo y al interés del graduado por el *open source* se decide conectar el punto de acceso a una red libre llamada The Things Network (TTN).

The Things Network es una plataforma de red global, comunitaria, libre y gratuita para el Internet of Things haciendo uso de la tecnología LoRaWAN para la conexión de los dispositivos. The Things

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

Network tiene como misión crear una red LoRaWAN global gracias a las aportaciones individuales de la comunidad y, por tanto, permite ofrecer un punto de conexión LoRaWAN gratuitamente a la comunidad que podrá hacer uso de ella libremente y sin suscripción de pago. Esto permite a la red expandirse y crear una cobertura mundial.

Por tanto, se conecta el punto de acceso a esta red, haciéndola disponible a cualquier nodo que emita en la frecuencia 868MHz con una modulación LoRa que este en el rango de alcance del punto de acceso.



Figura 79 – Comunidad The Things Network Valencia

Para la conexión con la plataforma recolectora de datos se precisa de tres claves de acceso a la red. Estas claves son únicas e identifican al dispositivo y son utilizadas tanto para la identificación en la red como para la encriptación de los datos.

- **Device EUI:** Se trata de una clave única hexadecimal de 8 bytes que identifica a cada dispositivo. Se trata de la dirección física o MAC como se conoce en los computadores. Este código irá impreso en una etiqueta externa a la vez de grabado en la memoria del chip.
- **Application EUI:** Identifica al dispositivo a nivel de aplicación y también consta de 8 bytes.
- **App Key:** Es la clave de acceso a la red LoRaWAN con una encriptación AES-128 y también es utilizada para realizar la encriptación de la información.

Tanto la Application EUI como la App Key dependen de la red y deben ser proporcionados por ella. Para ello en TTN se precisa de un registro del dispositivo dentro de la aplicación. En este registro es necesario disponer de la clave física o **Device EUI** ya que el servidor la utilizará para identificar que los mensajes que lleguen con esa identificación pertenecen a nuestra aplicación y no a otra.

El método de activación es OTTA. La *Figura 80* muestra como los datos van llegando a la consola de TTN cada 25 segundos aproximadamente (en modo depuración). Los mensajes son recibidos y decodificados correctamente. Para ello previamente se ha tenido que seleccionar la opción de que el *payload* o mensaje viene codificado según el estándar Cayenne LPP, ya que los datos en crudos vienen en bytes hexadecimales. También es posible definir formatos de payload customizados.

Memoria

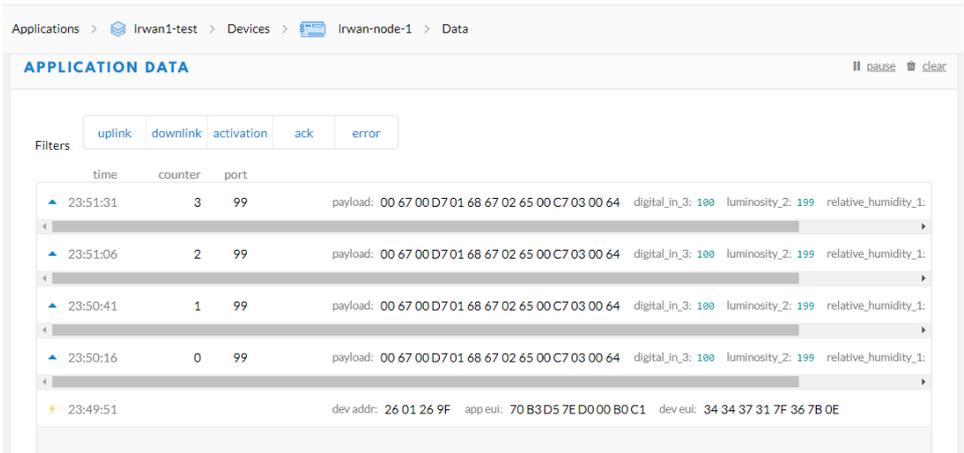


Figura 80 - Consola aplicación TTN

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

5.4. Aplicación final de usuario

Se ha desarrollado una aplicación web SPA donde el usuario final accede a consultar todos los datos recibidos por los nodos tanto en tiempo real como los datos históricos guardados en la base de datos. Esta aplicación se desarrolla con la intención de ofrecer una plataforma donde gestionar y visualizar los datos de manera fácil y muy visual pensada en la experiencia de usuario.

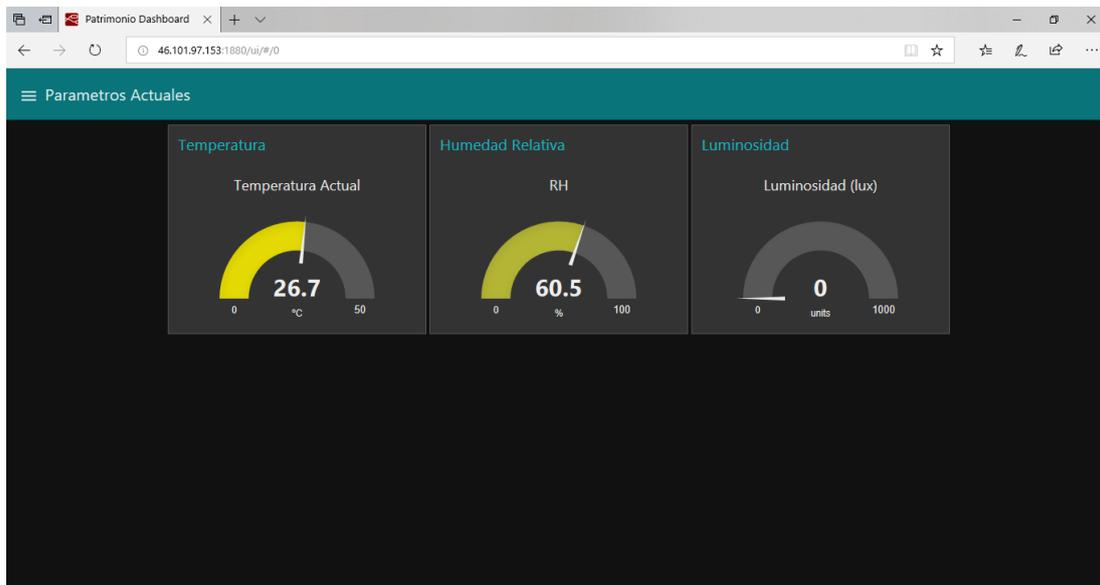


Figura 81 - Vista principal plataforma web

La aplicación ha sido desarrollada utilizando la herramienta Node-Red. Esta herramienta, programada en lenguaje JavaScript permite generar aplicaciones web de procesamiento para entornos IoT de manera sencilla mediante el concepto de bloques interconectados, de tal manera que permite conectar plataformas y servicios de manera muy fácil para programar nuestras propias aplicaciones. También permite disponer de una interfaz gráfica de usuario donde visualizar datos mediante gráficas o etiquetas, además de una interacción con el usuario mediante el uso de botones, sliders y derivados. A todo esto, se le suma una integración total con la plataforma de The Things Network a través de MQTT, para la cual existe un añadido muy fácil de configurar para recibir el *payload* del nodo y poder procesarlo y actuar en consecuencia en Node-Red.

Se trata de una aplicación de usuario muy sencilla que tan solo muestra los valores de temperatura, humedad y luminosidad que reportan los sensores tanto en unos indicadores visuales junto con el último dato reportado en tiempo real (*Figura 81*) como en unas gráficas temporales (*Figura 83*). Además, también almacena la información en crudo recibida por TTN (*Figura 85*) en una base de datos de tipo no relacional usando MongoDB. No se realiza ninguna acción con estos datos almacenados salvo pintarlos en las gráficas, pero en caso de ser necesario, están disponibles accediendo a la base de datos para poder consultarlos y analizarlos por expertos conservadores del patrimonio.

La *Figura 82* muestra la programación visual realizada mediante la conexión de diversos bloques. Esta programación se realiza en un entorno web que sirve la aplicación de Node-Red. Se puede observar como el dato en crudo recibido de la plataforma de TTN se envía a un bloque de tipo *mongodb out* que almacena el valor en la tabla de la base de datos configurada.

Memoria

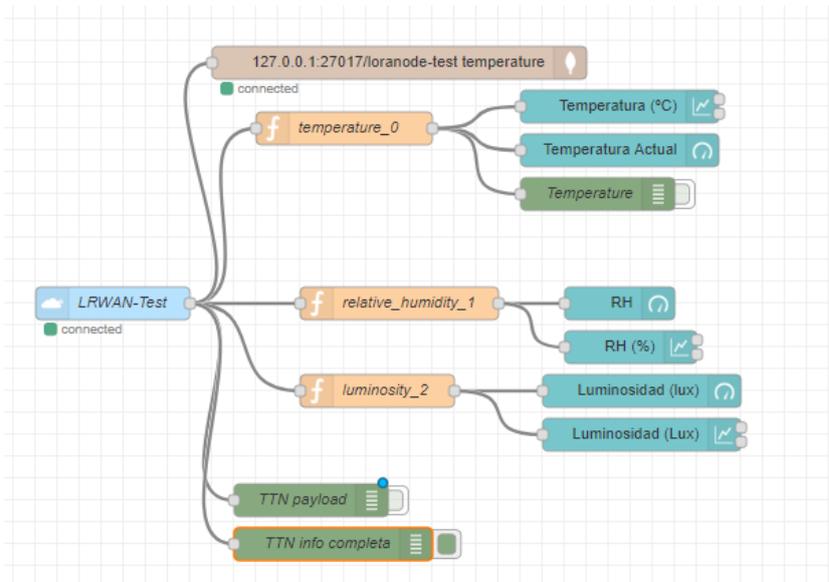


Figura 82 - Programación Node-Red aplicación usuario

Paralelamente el *payload* pasa por varios filtros que analizan y separan la temperatura, la humedad relativa y la luminosidad independientemente y la muestran tanto en un gráfico temporal de tipo *chart* como en un visualizador de tipo *gauge*. Los bloques verdes son bloques de depuración.

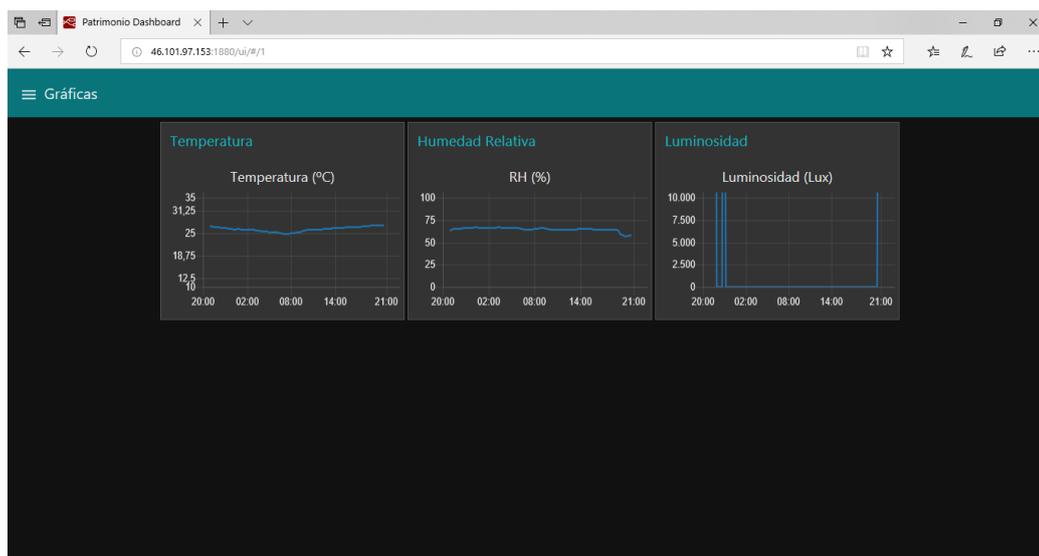


Figura 83 - Gráficas temporales aplicación

El *payload* viene codificado en un formato estándar llamado JSON (*JavaScript Object Notation*), que es un formato de texto ligero para el intercambio de datos. Tiene su origen en el lenguaje JavaScript, pero debido a su amplia adopción ya se considera un formato de lenguaje independiente que puede ser usado en técnicamente todos los lenguajes y ya supera en uso al XML. JSON define varios tipos de datos como números, cadenas, booleanos, arrays y objetos, que se codifican en distintos tipos de formato según el lenguaje. Existen bibliotecas nativas para cada uno de los diferentes lenguajes para trabajar con este tipo de notación con lo que resulta muy sencillo el intercambio de archivos entre plataformas.

En este caso, la API de The Things Network nos envía un JSON con toda la información del mensaje LoRa recibido en crudo (*Figura 85*) junto con información como el *timestamp* que codifica

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

la fecha y hora, la frecuencia y canal, y lo más interesante, los datos de información de los sensores decodificados según el formato definido por *Cayenne LPP* (Figura 84).

```
1. {  
2.   "digital_in_3": 100,  
3.   "luminosity_2": 199,  
4.   "relative_humidity_1": 51.5,  
5.   "temperature_0": 21.5  
6. }
```

Figura 84 - Información de los sensores

La aplicación es por el momento muy sencilla, pero puede escalarse según las necesidades específicas de los conservadores de patrimonio cultural.

Actualmente la aplicación web se encuentra hospedada en un servidor web VPS de la empresa DigitalOcean (<https://www.digitalocean.com/>). Esta empresa ofrece servidores privados virtuales en la nube para instalar prácticamente cualquier aplicación o servicio. Se trata de máquinas virtuales corriendo un sistema operativo Linux Ubuntu conectadas a Internet y permite gestionar los servicios como si de una máquina local se tratara. Resulta muy fácil instalar Node-Red junto con sus complementos para hacerlo funcionar. Además, los servidores privados son configurables y escalables, pudiendo empezar con un plan básico para poder ir escalándolo a una máquina más potente según las necesidades de nuestro proyecto.

En el momento de la escritura de esta memoria, la aplicación web de usuario es accesible en este enlace: <http://bit.ly/patrimoniolora>. No obstante, debido a que es un entorno de pruebas, no se garantiza la accesibilidad en un futuro.

Memoria

```
1. {
2.   "app_id": "lrwan1-test",
3.   "dev_id": "lrwan-node-1",
4.   "hardware_serial": "343437317F367B0E",
5.   "port": 99,
6.   "counter": 29,
7.   "payload_raw": [
8.     0,
9.     103,
10.    0,
11.    215,
12.    1,
13.    104,
14.    103,
15.    2,
16.    101,
17.    0,
18.    199,
19.    3,
20.    0,
21.    100
22.  ],
23.   "payload_fields": {
24.     "digital_in_3": 100,
25.     "luminosity_2": 199,
26.     "relative_humidity_1": 51.5,
27.     "temperature_0": 21.5
28.   },
29.   "metadata": {
30.     "time": "2018-06-10T19:05:15.94184252Z",
31.     "frequency": 867.3,
32.     "modulation": "LORA",
33.     "data_rate": "SF7BW125",
34.     "airtime": 66816000,
35.     "coding_rate": "4/5",
36.     "gateways": [
37.       {
38.         "gtw_id": "loragtw-multitech",
39.         "gtw_trusted": true,
40.         "timestamp": 743087659,
41.         "time": "2018-06-10T19:05:15Z",
42.         "channel": 4,
43.         "rssi": -71,
44.         "snr": 9.25,
45.         "rf_chain": 0,
46.         "latitude": 39.47683,
47.         "longitude": -0.614045,
48.         "location_source": "registry"
49.       }
50.     ]
51.   },
52.   "payload": {
53.     "digital_in_3": 100,
54.     "luminosity_2": 199,
55.     "relative_humidity_1": 51.5,
56.     "temperature_0": 21.5
57.   },
58.   "_msgid": "eb16dc87.7ae58"
59. }
```

Figura 85 - Payload TTN en crudo

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

5.5. Carcasa de protección

La electrónica es muy vulnerable a polvo y factores climáticos como lluvia o viento que pueden hacer que se estropee, tanto en exterior como en interior. Es por ello por lo que resulta de vital importancia encerrarla en una carcasa más o menos estanca que la proteja de estos daños.

No obstante, debido a que se trata de un sensor que mide parámetros del ambiente como temperatura, humedad y luz incidida, la electrónica no puede estar totalmente cubierta, y ha de diseñarse de tal manera que se deje una apertura en la zona de los sensores. El diseño de la electrónica ya se diseñó teniendo esta idea en mente y, por tanto, se habilitó una apertura en la PCB que además de servir para aislar térmicamente del resto de circuitería, es utilizada para proteger el resto de la electrónica exponiendo la zona de los sensores.



Figura 86 - Renderizado 3D del nodo sensor montado en la carcasa

La carcasa consta de dos partes, la propia carcasa o caja donde van ensambladas mediante tornillos las dos placas electrónicas junto con las baterías y una tapa que encaja en la carcasa y que cierra el conjunto. (Figura 88). Se trata de un diseño muy simple pero que resulta suficiente para este primer prototipo. La carcasa también incluye aberturas para la pantalla OLED de 0.96 pulgadas, la antena, un agujero para facilitar el montaje en una pared y un acceso al conector USB para reprogramar la tarjeta.

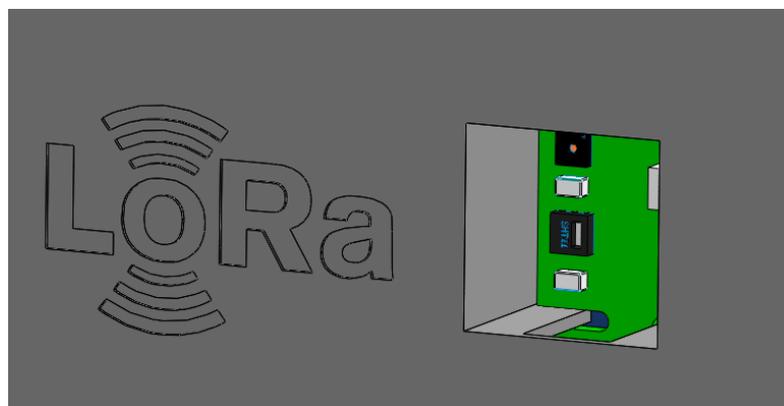


Figura 87 - Detalle apertura sensores

Memoria

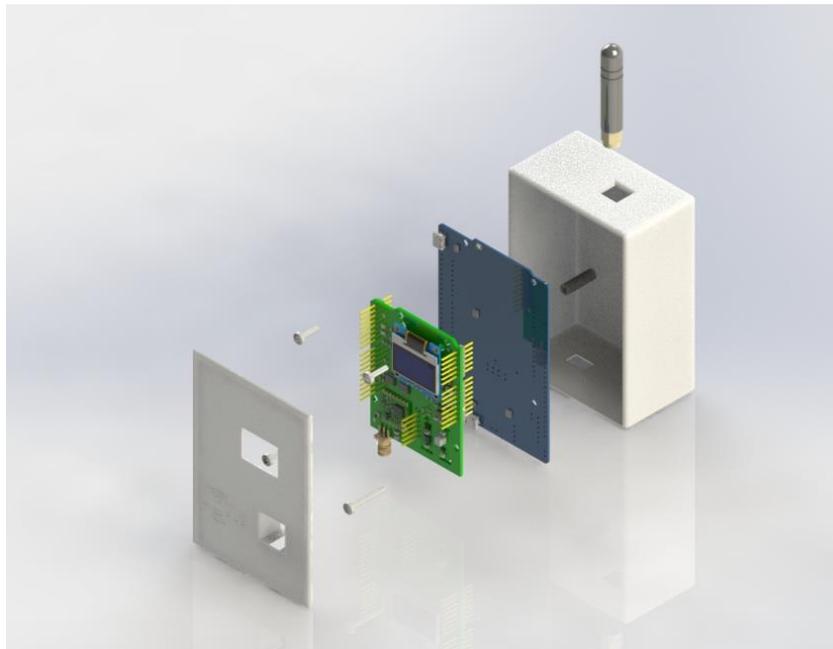


Figura 88 - Renderizado 3D vista explotada

La carcasa se ha impreso en 3D con el objetivo de verificar el diseño mecánico de la misma y que las tolerancias son las correctas.



Figura 89 - Prototipo completo instalado junto a un cuadro

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

5.6. Pruebas de cobertura

Se han realizado pruebas de cobertura con el objetivo de poner a prueba el punto de acceso y la tecnología para comprobar si efectivamente cumple con lo prometido.

LoRaWAN y la modulación LoRa en particular, promete alcances teóricos de hasta 15km en zonas suburbanas y 5km en zonas urbanas. No obstante, esto se consigue en un ambiente ideal y con la antena o punto de acceso posicionada en una zona alta en el exterior y siempre manteniendo la línea de visión (*line of sight*) entre el punto de acceso o pasarela y el nodo emisor.

Las pruebas realizadas no son las más favorables para la tecnología ya que se han realizado con la pasarela instalada en el interior de una vivienda (configuración *indoor*) y sin tener línea de visión directa, como se puede apreciar en el perfil de altura del terreno (*Figura 92*).

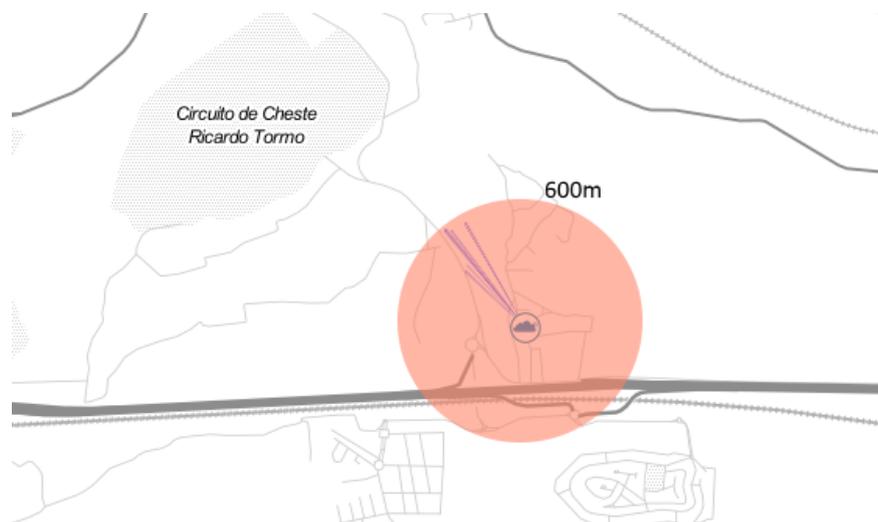


Figura 90 - Radio de alcance conseguido

Para la prueba de alcance se ha utilizado un servicio web asociado a la plataforma The Things Network conocido como TTNMapper (<https://ttnmapper.org>). Se trata de una aplicación para teléfonos inteligentes que hace uso del GPS para localizar con exactitud los paquetes recibidos. Se trata de conectar la aplicación de la plataforma TTN con la aplicación de tal manera que cuando se recibe un paquete de datos, la aplicación lo registra con las coordenadas del GPS obtenidas del teléfono.

La prueba ha consistido en recorrer varios puntos cercanos a la localización de la pasarela con el nodo emitiendo paquetes cada 25 segundos y el teléfono móvil con la aplicación de TTNMapper reportando los datos GPS. Posteriormente estos datos son mostrados en la página web del servicio, pudiendo ver que límite se ha alcanzado.

Se han conseguido una cobertura radial de 600m, no obstante, se trata de una situación muy desfavorable debido sobre todo a que, como se puede apreciar en el perfil de elevación del terreno, no había una línea de visión directa entre pasarela y nodo y la señal se atenúa en gran medida con obstáculos fijos como montañas o elevaciones. Además, el nodo se encuentra instalado en una vivienda a una altura de un metro sobre el suelo.

Memoria

Para una futura mejor evaluación del alcance, se pueden realizar pruebas más exactas y controladas instalando la antena en el exterior a una altura superior a 10 metros y recoger la información de varios puntos radiales. Con todos estos puntos se podrá dibujar un mapa de calor para estimar la cobertura en situaciones más favorables y reales según se especifica en el documento de Carrión Gonzalez, Marianela. [40]



Figura 91 - Paquete recibido a 600 metros de la pasarela



Figura 92 - Perfil de elevación del terreno

En el perfil de elevación, el 0 en el eje de abscisas representa la situación de la pasarela, y los 0,6km la situación desde donde se envió el paquete.

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

5.7. Dimensionado de la batería y autonomía

Para la selección de la batería se ha realizado un cálculo estimativo teniendo en cuenta el consumo global del nodo en funcionamiento y la autonomía que se requiere que tenga (en principio superior a un año). En función de esto, también se selecciona el intervalo de transmisión.

Se trata de un cálculo teórico estimativo con la información que provee el fabricante en sus hojas de datos de cada sensor e integrado utilizado y por tanto en la práctica puede variar en gran medida debido a las pérdidas de energía por corrientes parásitas que puedan existir o a rangos de temperatura y condiciones no ideales. Los cálculos que se van a realizar son sin tener en cuenta los LED de depuración ni las ineficiencias generadas por los reguladores internos de la placa de desarrollo y en situaciones ideales.

En primer lugar, se han definido tres estados de funcionamiento para el nodo inalámbrico con el objetivo de caracterizar cada modo de funcionamiento, y realizar un cálculo estimativo de la corriente consumida en cada uno de los modos. El nodo pasará por estos tres estados de manera cíclica en el orden que los define. Posteriormente se definirá un cronograma para identificar cuanto tiempo pasará el nodo en cada estado.

- 1. Medida de los sensores:** En este modo, el microcontrolador está activo y genera una llamada a los sensores para obtener los parámetros. En este modo están activos el microcontrolador y los periféricos de comunicación como el I2C, así como los dos circuitos integrados de los sensores.
- 2. Transmisión LoRaWAN:** Los datos son transmitidos inalámbricamente por el nodo haciendo uso de la modulación LoRa. Se encuentran activos el microcontrolador y el transceptor LoRa transmitiendo a máxima potencia.
- 3. Modo dormido:** Este será el estado principal del nodo en el cual pasará la mayoría del tiempo. En este estado todos los módulos se encuentran inactivos haciendo uso de sus modos *sleep*.

A continuación, se identifica el estado de cada dispositivo o submódulo en cada uno de los estados definidos anteriormente, con el objetivo de estimar su consumo (*Tabla 8*)

Estado	MCU	SHT31	TSL2591	LoRa Tx
Medida de los sensores	ON	ON	ON	OFF
Transmisión LoRa	ON	OFF	OFF	ON
Modo dormido	OFF	OFF	OFF	OFF

Tabla 8 - Estado de los módulos en cada estado

Posteriormente se realiza una estimación de la corriente consumida por cada submódulo principal en cada estado, según la información recogida en las hojas de características (*Tabla 9*)

Dispositivo	Consumo ON	Consumo OFF
LRWAN1 EVB	8 mA	1,3 uA
SHT31	1.5 mA	2 uA
TSL2591	0.325 mA	4 uA
SX1276	128 mA	1 uA

Tabla 9 - Consumos de los integrados

Memoria

Con estos datos, se realiza la suma de consumos en cada uno de los modos de acuerdo con los estados indicados anteriormente.

Estado	MCU	SHT31	TSL2591	LoRa Tx	TOTAL
Medida de los sensores	8 mA	1,5 mA	0,325 mA	1 uA	11,325 mA
Transmisión LoRa	8 mA	2uA	4 uA	128 mA	136,006 mA
Modo dormido	1,3 uA	2uA	4uA	1uA	0,0083 mA

Tabla 10 - Suma de los consumos

Como vemos, el modo que más consumo instantáneo representa es, como es lógico, el modo en el cual se está transmitiendo el mensaje con la información de los sensores inalámbricamente.

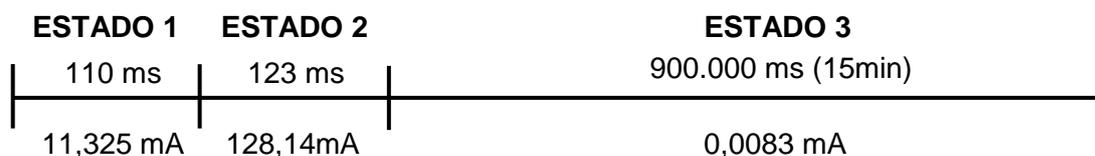


Figura 93 - Cronograma de los estados del ciclo de envío

En tercer lugar, se crea un cronograma para estimar el tiempo que pasa el dispositivo en cada estado (Figura 93), cabe destacar que el cronograma no está hecho a escala, pues el tiempo que pasa en el estado 1 y 2 representa una minucia comparado con el modo dormido o estado 3. En este momento se ha estimado que el nodo envía datos a razón de una vez cada 15 minutos y, por tanto, permanece 900.000 milisegundos en estado inactivo. El tiempo que permanece en el estado dos, es decir transmitiendo, corresponde a la estimación del *air-time* correspondiente a un *payload* de 29 bytes con SF7 y 125BW.

Por último, se realiza el cálculo para estimar cuantos mAh de energía consume en un clico de envío sabiendo cuanto tiempo está consumiendo esa corriente. Resulta un cálculo simple. Veremos un ejemplo en el modo 1, los demás serán calculados de la misma manera. Se puede ver un resumen en la Tabla 11.

$$11,325 \text{ mA} \times 110 \text{ ms} = 1245,75 \text{ mAms} \times \frac{1 \text{ h}}{3600000 \text{ ms}} = 0,34604 \mu\text{Ah}$$

Estado	Consumo ciclo
Medida de los sensores	0,34604 μ Ah
Transmisión LoRa	4,37812 μ Ah
Modo dormido	2,075 μ Ah
TOTAL TRANSMISIÓN	6,8 μAh

Tabla 11 - Gasto energético que representa realizar un envío

Obtenemos que para realizar un ciclo completo se emplean **6,8 μ Ah** de energía.

Si la batería tiene disponibles **1200mAh** de capacidad sabemos que podremos realizar **176470 envíos**, a razón de **96 envíos por día** (intervalos de 15 minutos) son **1838 días**, o lo que es lo mismo, poco más de **5 años**.

Concluimos que con una batería de 1200mAh como las **Energizer L92-FR03** es suficiente como para dar servicio durante 5 años.

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

6. Conclusiones

6.1. Sobre el trabajo realizado

En el presente proyecto se ha intentado abarcar el máximo de disciplinas distintas aprendidas durante la carrera con el objetivo de realizar un trabajo a modo de resumen del grado. Se ha realizado un diseño electrónico completo con sistemas digitales y analógicos y se ha utilizado buses de comunicación propios de sistemas embebidos como el I2C o el SPI. Además, se ha tratado de enfocar el proyecto de tal manera que ha permitido formarse en nuevas tecnologías y materias que no se han tratado con profundidad en el Grado de Electrónica como la radiofrecuencia inalámbrica o las telecomunicaciones, que me han hecho aprender nuevos conceptos como potencia de transmisión, cobertura, *line of sight* y modulaciones, así como conceptos sobre la colocación, tipos de antenas y su diseño, conectores de señal como los SMA, UFL y demás.

Asimismo, se ha hecho una gran labor de investigación sobre el estado del arte en tecnologías de transmisión inalámbricas muy recientes como las redes inalámbricas LPWAN, o el BLE y en concreto sobre LoRaWAN, el Internet de las Cosas y la Industria 4.0.

Por el contrario, puesto que se han tratado de abarcar muchos temas y construir un sistema completo en el mismo proyecto, no se ha llegado a profundizar en ninguno de estos como me hubiera gustado. No obstante, me siento satisfecho y considero que he hecho un buen trabajo que me ha permitido aprender de esta gran oleada de tecnologías que surgen con el IoT y que, sin duda, tienen un gran futuro por delante.

El proyecto ha estado enfocado como un primer prototipo como prueba de concepto con la intención de crear un producto real y completo con una salida directa al mercado. De este modo se ha realizado desde el diseño electrónico del nodo sensor hasta la aplicación web de usuario pasando por infraestructura de red necesaria y la instalación de los puntos de acceso, la programación completa del firmware y software necesarios e incluso el diseño mecánico 3D de la carcasa. El prototipo diseñado se muestra como un producto cerrado completamente preparado para pruebas de campo.

Además, el proyecto me ha permitido reforzar competencias relacionadas con la comunicación con diversas empresas para el intercambio de documentación, costes de componentes electrónicos y muestras y fabricación de piezas customizadas como la placa de circuito impreso, así como la generación de documentación relacionada con procesos de fabricación como la *Bill of Materials*, presupuestos, planos de fabricación o generación de fotolitos para la fabricación de PCB.

6.2. Mejoras

El proyecto presenta un gran margen de mejora ya que se trata de un diseño que ha partido prácticamente desde cero y ha sido desarrollado como un prototipo prueba de concepto, por lo que es muy mejorable y escalable.

El siguiente paso respecto al desarrollo electrónico según la naturaleza del proyecto pasa por una miniaturización del mismo, ya que el primer prototipo tiene un tamaño considerable y poco óptimo. Esto se puede llevar a cabo realizando un diseño que integre el chip de Murata y la electrónica necesaria para hacerlo funcionar, permitiendo eliminar la placa de evaluación de STMicroelectronics. Dicho diseño también debería incluir un soporte para la batería que facilite el ensamblaje en futuras series pensando en una producción en masa y debe prescindir de todos los

Memoria

integrados y sensores que no se vayan a utilizar, así como de la pantalla OLED que solo es usada durante el desarrollo. Todo esto con la intención de reducir tamaño y costes.

En segundo lugar, al reducir el tamaño de la placa electrónica y embeberla toda en una sola placa, se puede reducir en gran medida la carcasa que la recubre y aprovechando el rediseño, hacerla resistente al agua y al polvo según una certificación IP (Ingress Protection) adecuada. Esto permitirá garantizar su uso en exteriores.

Pese a haber estimado el consumo de energía en más de 5 años, el consumo sigue siendo relativamente alto, y se puede mejorar en gran medida optimizando la programación del nodo, ya que, por el momento, el firmware no está optimizado para ello. También conviene hacer ensayos prácticos y medidas que permitan garantizar una estimación de autonomía en condiciones reales.

En futuras revisiones, se puede estudiar el uso de tecnologías de *Energy Harvesting* mediante el uso de pequeñas células fotovoltaicas del estilo a las que llevan las calculadoras solares o capturando la energía de las ondas electromagnéticas de radio en el aire. Actualmente existen circuitos integrados en el mercado que te permiten gestionar la carga de condensadores y capturar y optimizar el consumo de energía al máximo con reguladores lineales muy eficientes que no deberían dificultar demasiado el uso de tecnologías como el *Energy Harvesting*.

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

7. Bibliografía

- [1] Telefónica, «Telefónica presenta “Smart Patrimonio”, una solución que monitoriza los bienes de Patrimonio Cultural para su conservación preventiva,» *Telefónica*, 25 Junio 2015.
- [2] T. Padfield, «How to keep for a while what you want to keep for ever,» *Conservation Physics*, 2005.
- [3] IEEE, 802.11aj-2018 - Ieee standard for information technology--telecommunications and information exchange between systems local and metropolitan area networks: MAC and physical layer specifications, 2018.
- [4] K. -. A. Townsend, *Introduction to Bluetooth Low Energy*, 2017.
- [5] IEEE, 802.15.4™-2015, IEEE Standard for Low-Rate Wireless Networks, 2015.
- [6] J. M. Moreno y D. Ruiz Fernández, Informe Técnico: Protocolo ZigBee (IEEE 802.15.4), 2017.
- [7] Z. Becvar, P. Mach y I. Pravda, *Redes móviles*, Czech Republic.
- [8] Motorola Solutions, «1973 - 1983: Making History: Developing the Portable Cellular System,» *Motorola Solutions*.
- [9] P. Pardal, *Redes de Área Extensa para aplicaciones de IoT: modelado de comunicaciones Sigfox*, Valencia: UPV, 2017.
- [10] M. Rouse, «Internet Of Things Agenda,» TechTarget, [En línea]. Disponible: <https://internetofthingsagenda.techtarget.com/definition/LPWAN-low-power-wide-area-network>.
- [11] LoRa Alliance, *LoRaWAN™ Specification*, 2016.
- [12] IoTNet, «IoTNet,» [En línea]. Disponible: <http://www.iotnet.mx/index.php/2017/11/06/lora-vs-nb-iot-vs-sigfox-cual-es-el-mejor/>.
- [13] «Sigfox,» [En línea]. Disponible: <https://www.sigfox.com>.
- [14] Vodafone, «Narrowband - IoT,» 2018. [En línea]. Disponible: <http://www.vodafone.com/business/iot/managed-iot-connectivity>.
- [15] M. Sierra, «Vodafone se adelanta a Movistar y Orange y lanza el Internet de las Cosas de las Smart Cities,» *VozPopuli*, 24 Enero 2017.
- [16] A. Mancheño, «Gestión inteligente de residuos urbanos con tecnología Narrow Band IoT (NB-IOT),» *Esmartcity.es*, 4 Octubre 2017.
- [17] Ingenium, «RPMA vs Competition,» [En línea]. Disponible: <http://www.ingenium.com/technology/rpma/competition/>.
- [18] Wikipedia, «Arduino,» [En línea]. Disponible: <https://es.wikipedia.org/wiki/Arduino>. [Último acceso: Mayo 2018].
- [19] Atmel, *ATmega328/P Datasheet*.
- [20] Arduino.cc, «Arduino Uno Rev3,» [En línea]. Disponible: <https://store.arduino.cc/arduino-uno-rev3>.
- [21] Creative Commons, «Reconocimiento-CompartirIgual 3.0 España (CC BY-SA 3.0 ES),» [En línea]. Disponible: <https://creativecommons.org/licenses/by-sa/3.0/es/>. [Último acceso: Mayo 2018].

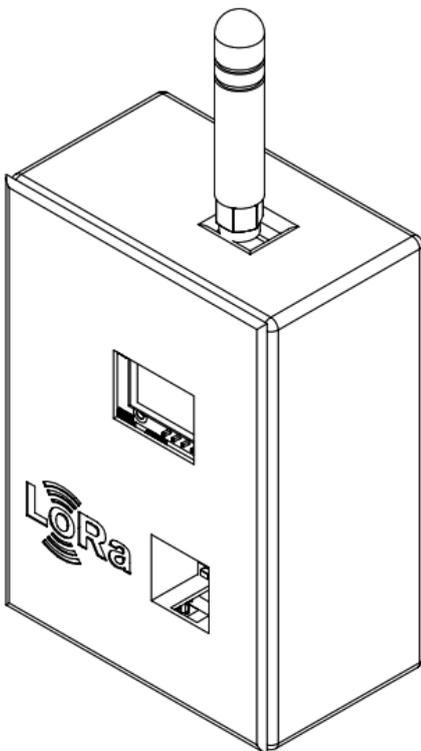
Memoria

- [22] Arm Developer, «Cortex-M0+,» [En línea]. Disponible: <https://developer.arm.com/products/processors/cortex-m/cortex-m0-plus>. [Último acceso: Mayo 2018].
- [23] STMicroelectronics, «STM32L0 series of ultra-low-power MCUs,» [En línea]. Disponible: <http://www.st.com/en/microcontrollers/stm32l0-series>. [Último acceso: Mayo 2018].
- [24] STMicroelectronics, Datasheet STM32L072CZ - en.DM00141133, 2018.
- [25] IoTBits, «Ultra Low Power Design using ESP32,» [En línea]. Disponible: <http://iot-bits.com/ultra-low-power-design-esp32-10-tips/>. [Último acceso: Mayo 2018].
- [26] Semtech, SX127x Reference Design, 2018.
- [27] HopeRF Electronic, RFM95/96/97/98 Datasheet, 2017.
- [28] Murata, CMWX1ZZABZ Datasheet, 2017.
- [29] Arm, «Arm Mbed Platform,» [En línea]. Disponible: <https://www.mbed.com/en/platform/>. [Último acceso: Mayo 2018].
- [30] Battery University, «Types of Lithium-ion,» [En línea]. Disponible: http://batteryuniversity.com/learn/article/types_of_lithium_ion. [Último acceso: Mayo 2018].
- [31] Energizer, Cylindrical Primary Lithium, 2016.
- [32] Energizer, L92-FR03 Product Datasheet, 2016.
- [33] PowerTech Systems, «Primary cells Lithium Thionyl Chloride (Li-SOCl₂),» 2015. [En línea]. Disponible: <https://www.powertechsystems.eu/home/products/primary-cells-lithium-thionyl-chloride-li-socl2/>. [Último acceso: Mayo 2018].
- [34] Saft Batteries, LS 14500Ex Primary Lithium battery Datasheet, 2018.
- [35] Texas Instruments, I2C Bus Pullup Resistor Calculation, 2015.
- [36] NXP, UM10204 I2C-bus specification and user manual, 2002.
- [37] Diario Oficial de la Unión Europea, Directiva 2014/53/UE (Radio Equipment Directive, RED), 2014.
- [38] Exar Corporation, SPX3819 Datasheet, 2012.
- [39] MyDevices.com, «Cayenne Low Power Payload,» 2017. [En línea]. Disponible: <https://mydevices.com/cayenne/docs/lora/#lora-cayenne-low-power-payload>. [Último acceso: Mayo 2018].
- [40] M. D. C. C. González, Desarrollo De Una Pasarela LORA y Evaluación De Prestaciones, Valencia: UPV, 2017.
- [41] «Wikipedia - Sensor,» 08 Mayo 2017. [En línea]. Disponible: <https://es.wikipedia.org/wiki/Sensor>.
- [42] Apuntes Instrumentación Electrónica, Introducción a los Sistemas de Medida, 2016.
- [43] Ministerio de energía, turismo y agenda digital, Cuadro Nacional de Atribución de Frecuencias (CNAF), 2017.
- [44] K. Ashton, «That 'Internet of Things' Thing,» *RFID Journal*, 2009.
- [45] B. Ray, «What is LoRaWAN,» Link Labs, 2015. [En línea]. Disponible: <https://www.link-labs.com/blog/what-is-lorawan>.
- [46] Panasonic, NCR18650 Lithium Ion Datasheet, 2012.



DESARROLLO DE UN NODO INALÁMBRICO DE BAJO CONSUMO BASADO EN TÉCNICAS DE IOT PARA MONITORIZACIÓN DE PATRIMONIO CULTURAL

2. PLANOS



Autor:
D. Jaime Laborda Macario

Tutor:
D. Ángel Perles Ivars

Valencia, mayo de 2018

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para
monitorización de patrimonio cultural

Índice de planos

1.	Plano electrónico	1
1.1.	Esquemático electrónico de conexiones.....	1
1.1.1.	Cover & Revisions.....	1
1.1.2.	Top.....	2
1.1.3.	RFM95	3
1.1.4.	Sensors.....	4
1.2.	Capas de fabricación circuito impreso	5
1.2.1.	Plano ensamblaje Top	5
1.2.2.	Plano ensamblaje Bottom	6
1.2.3.	Plano contorno	7
1.2.4.	Plano cobre Top.....	8
1.2.5.	Plano cobre Bottom.....	9
1.2.6.	Plano mascara de Soldadura Top.....	10
1.2.7.	Plano mascara de Soldadura Bottom.....	11
1.2.8.	Plano serigrafía Top.....	12
1.2.9.	Plano serigrafía Bottom.....	13
1.2.10.	Plano pasta de soldadura Top.....	14
1.2.11.	Plano pasta de soldadura Bottom.....	15
1.2.12.	Plano de fabricación y taladros.....	16
2.	Plano mecánico	17
2.1.	Vista general ensamble	17
2.2.	Plano explosión ensamblaje	18
2.3.	Plano de vistas caja principal	19
2.4.	Plano de vistas tapa exterior	20

Arduino LoRa Sensor shield

Revision History

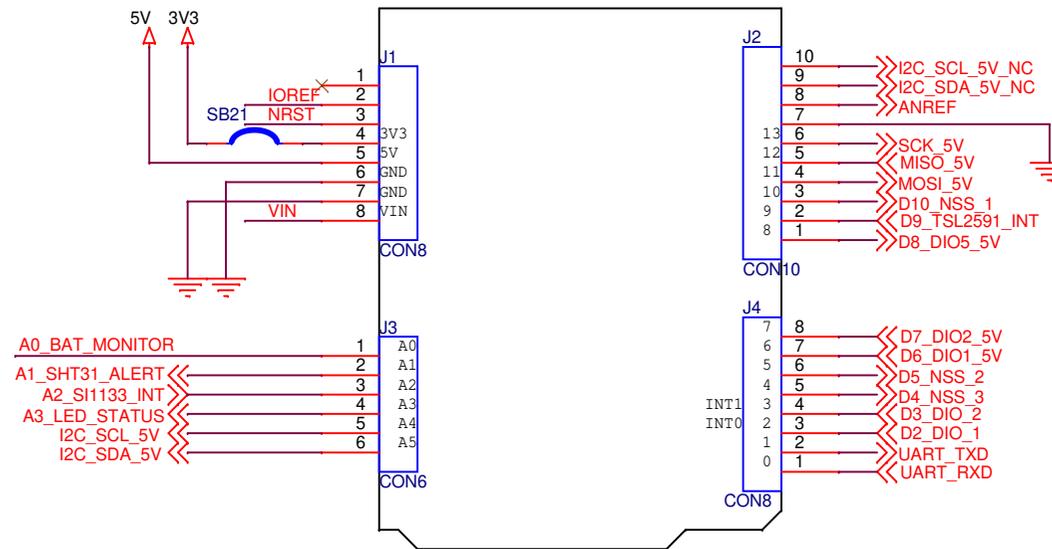
0.1	Feb 11th, 2018	J. Laborda	First version for proof of concept prototype
0.2	Jun 09th, 2018	J. Laborda	Changed I2C pullup resistor value (R13, R14). Q1 and Q2 as DNP

Table of Content

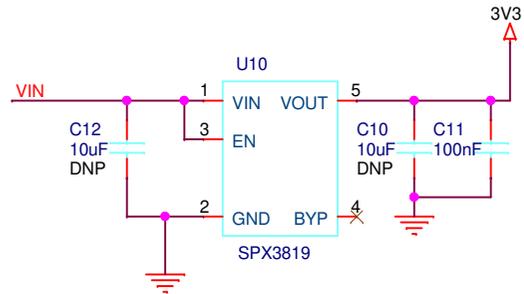
Page 0	Cover, Revisions
Page 1	Top: Arduino Connector, Backup LDO
Page 2	RFM95 LoRa Transceiver
Page 3	Sensors

J. Laborda		
Title 0. Cover & Revision		
Size A4	Project Name Arduino LoRa Sensor shield	Rev 0.1
Date: Saturday, June 09, 2018		Sheet 1 of 4

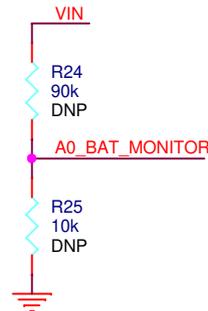
ARDUINO CONNECTOR



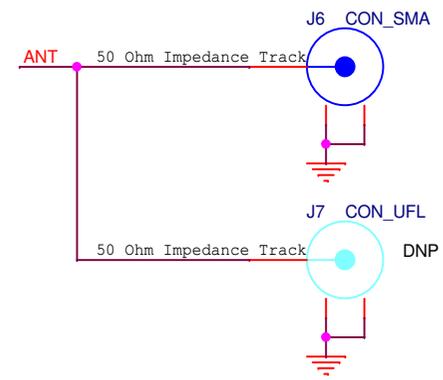
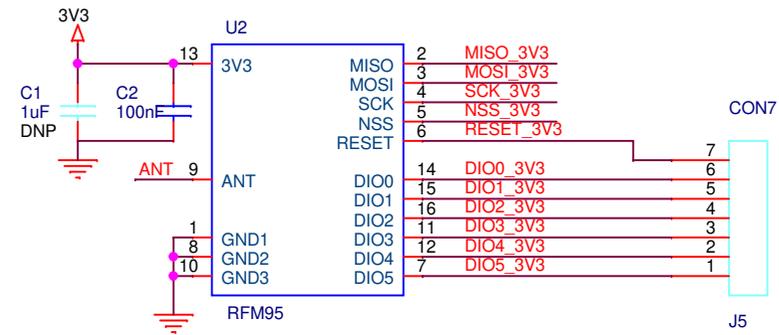
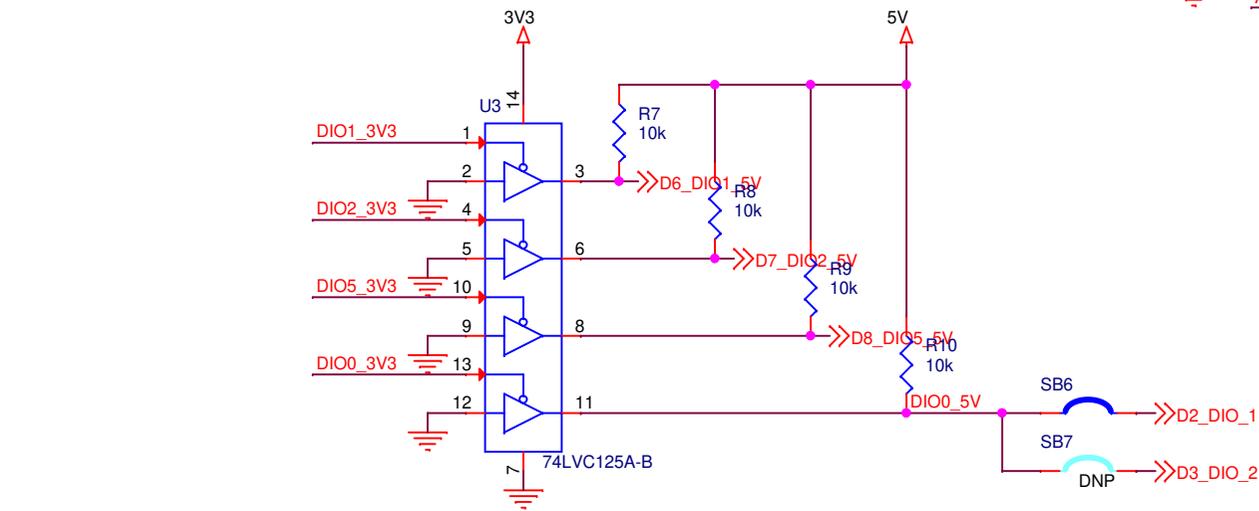
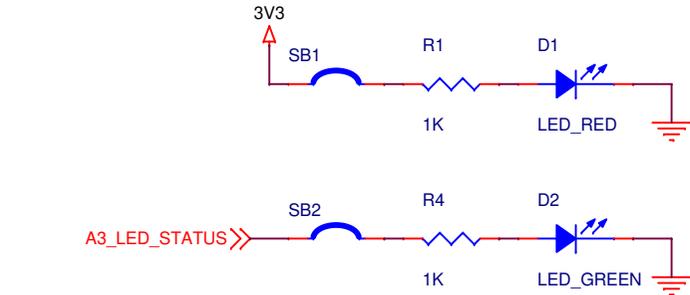
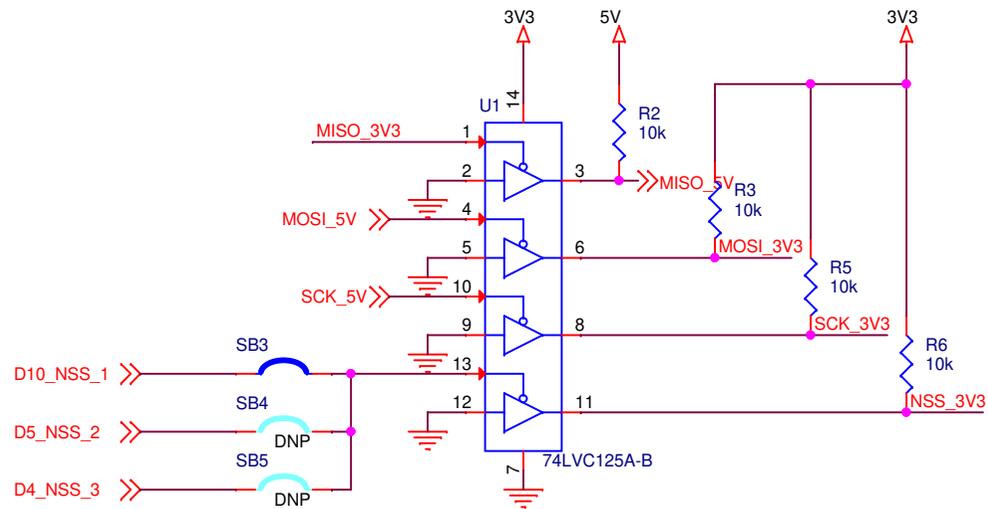
BACKUP 3V3@500mA LDO



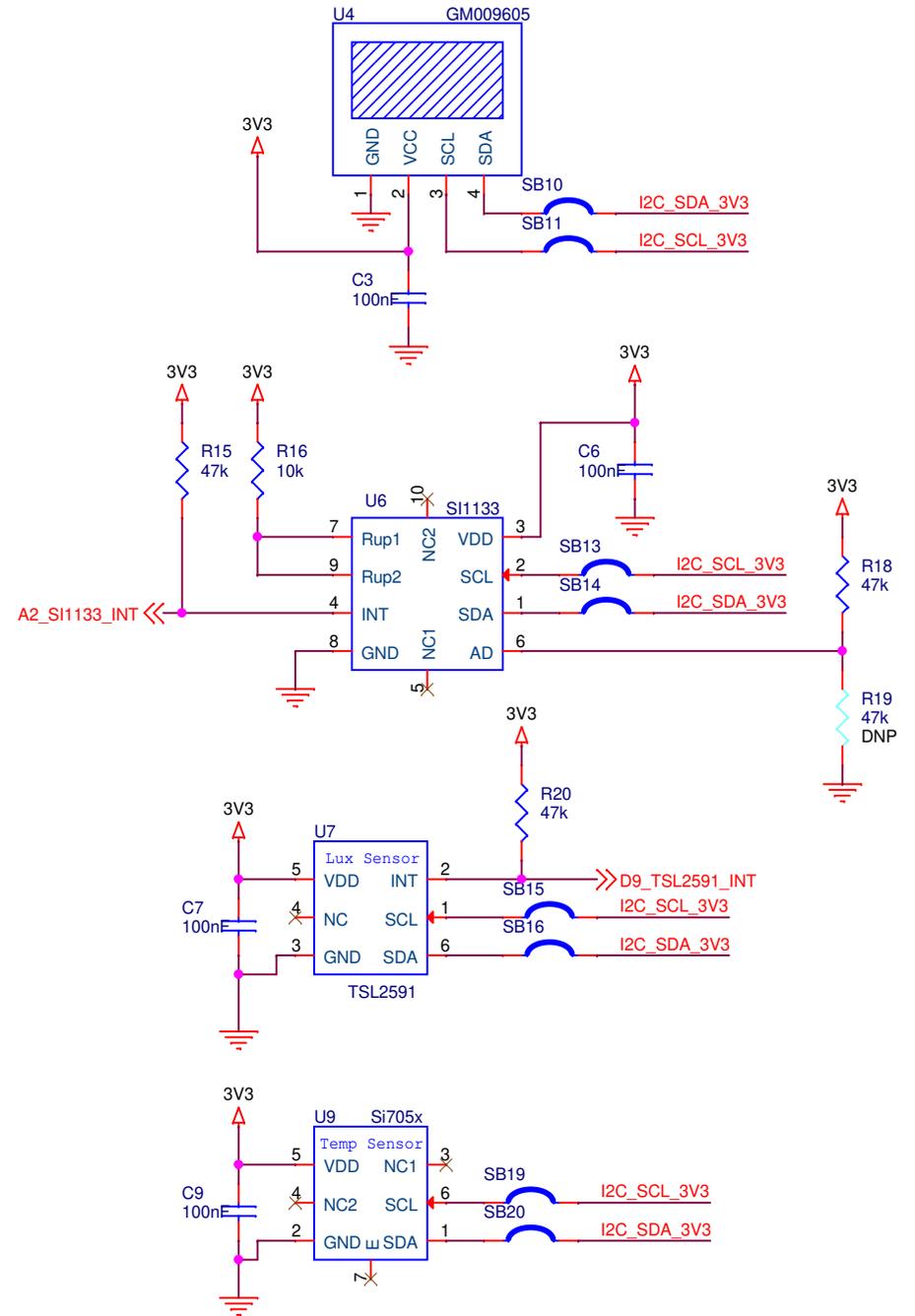
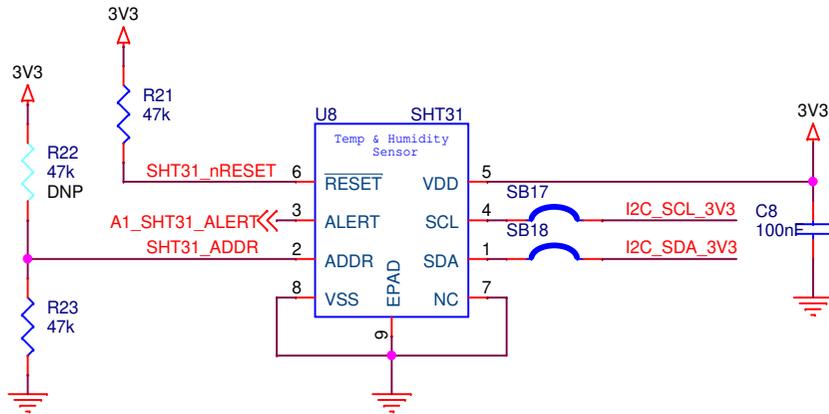
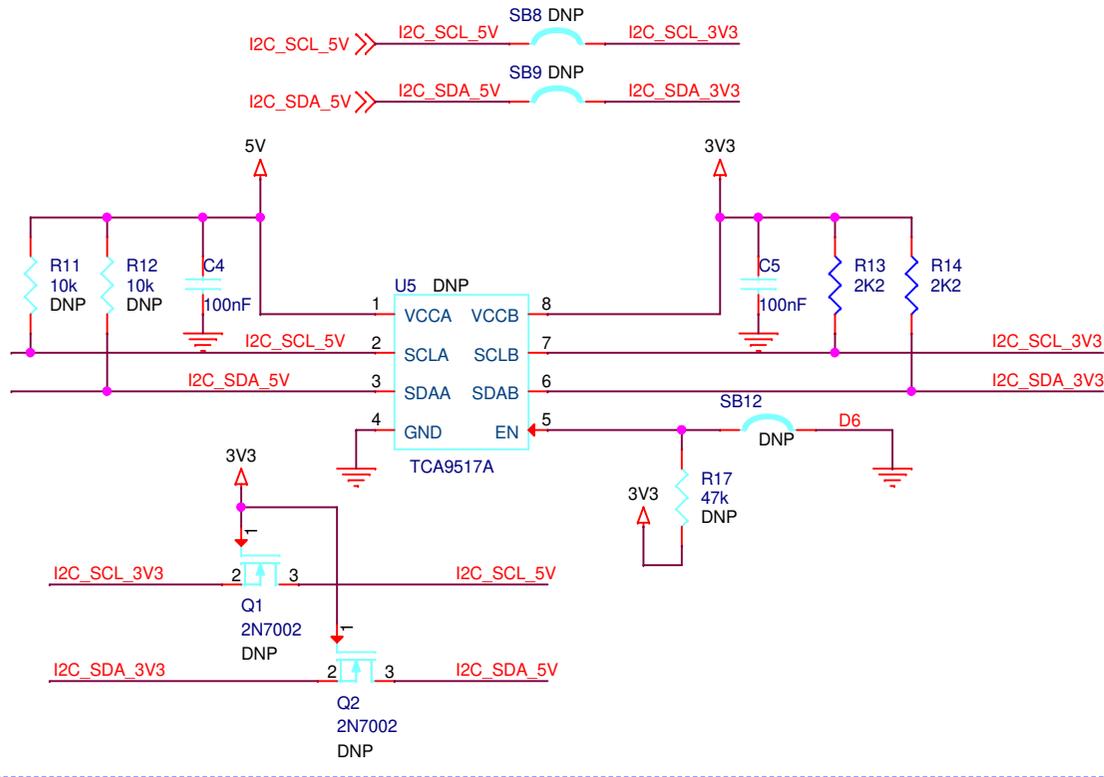
VOLTAGE MONITORING



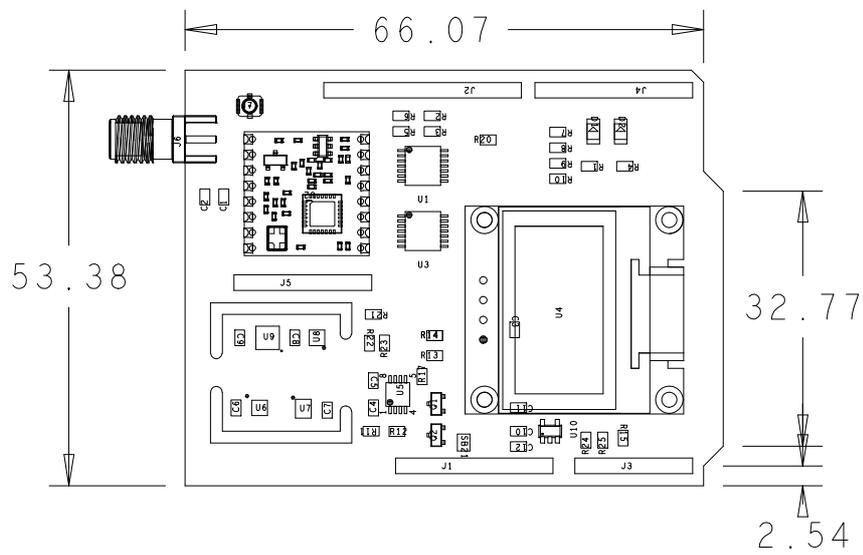
J. Laborda		
Title 1. Top		
Size A4	Project Name Arduino LoRa Sensor shield	Rev 0.1
Date: Sunday, February 11, 2018		Sheet 2 of 4



I2C REPEATER DNP



J. Laborda		
Title 3. Sensors		
Size A4	Project Name Arduino LoRa Sensor shield	Rev 0.1
Date: Saturday, June 09, 2018	Sheet 4	of 4



SI NO SE INDICA LO CONTRARIO:
 LAS COTAS SE EXPRESAN EN MM
 TOLERANCIAS
 LINEAL
 ANGULAR:

ACABADO:

NO CAMBIE LA ESCALA

REVISION

Plano PCB 1

TITULO:

ASSEMBLY TOP

N. DE DIBUJO

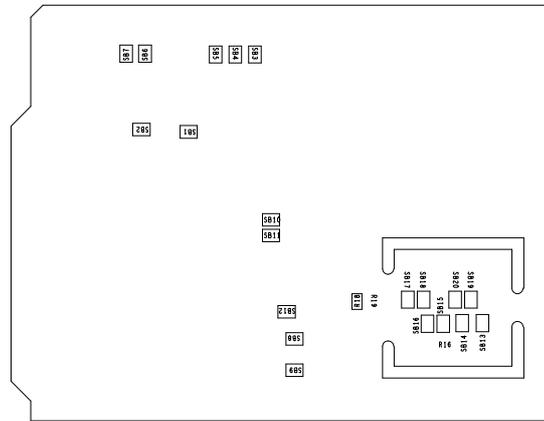
ARDUINO_LORA_SENSOR_SHIELD.ASMT

A4

PESO:

ESCALA 1:1

HOJA 1 DE 1



SI NO SE INDICA LO CONTRARIO:
 LAS COTAS SE EXPRESAN EN MM
 TOLERANCIAS
 LINEAL
 ANGULAR:

ACABADO:

NO CAMBIE LA ESCALA

REVISION

Plano PCB 2

TITULO:

ASSEMBLY BOTTOM

N. DE DIBUJO

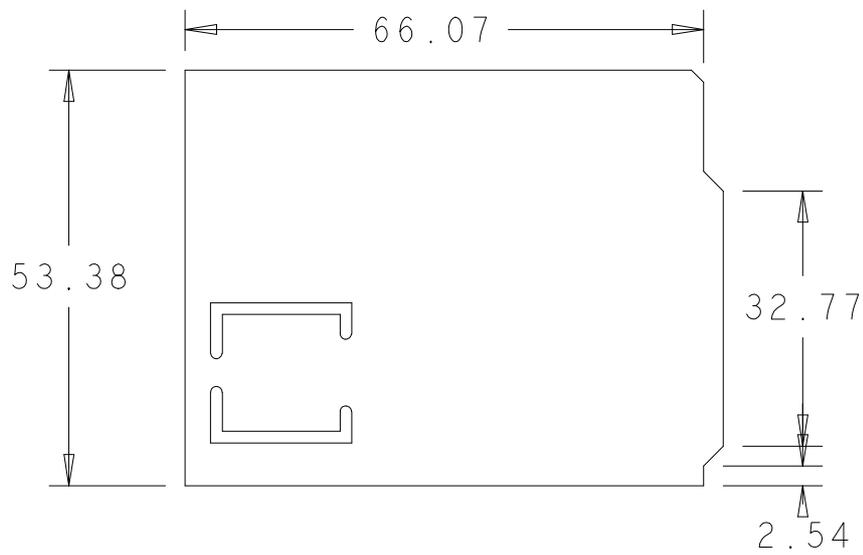
ARDUINO_LORA_SENSOR_SHIELD.ASMB

A4

PESO:

ESCALA 1:1

HOJA 1 DE 1



SI NO SE INDICA LO CONTRARIO:
LAS COTAS SE EXPRESAN EN MM
TOLERANCIAS
LINEAL
ANGULAR:

ACABADO:

NO CAMBIE LA ESCALA

REVISION

Plano PCB 3

TITULO:

OUTLINE

N. DE DIBUJO

ARDUINO_LORA_SENSOR_SHIELD_OUTLINE

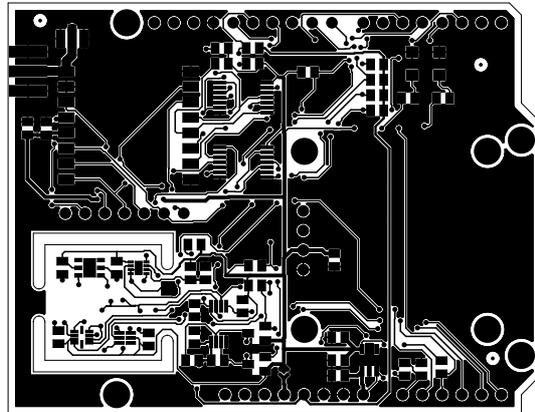
A4

PESO:

ESCALA 1:1

HOJA 1 DE 1

	NOMBRE	FIRMA	FECHA		
DIBUJ.	J .Laborda		27/01/18		
VERIF	A. Perles		05/06/18		
APROB	J .Laborda		05/06/18		
FABR.					
CALID.				MATERIAL	



SI NO SE INDICA LO CONTRARIO:
LAS COTAS SE EXPRESAN EN MM
TOLERANCIAS
LINEAL
ANGULAR:

ACABADO:

NO CAMBIE LA ESCALA

REVISION

Plano PCB 4

TITULO:

COPPER TOP

	NOMBRE	FIRMA	FECHA			
DIBUJ.	J .Laborda		27/01/18			
VERIF	A. Perles		05/06/18			
APROB	J .Laborda		05/06/18			
FABR.						

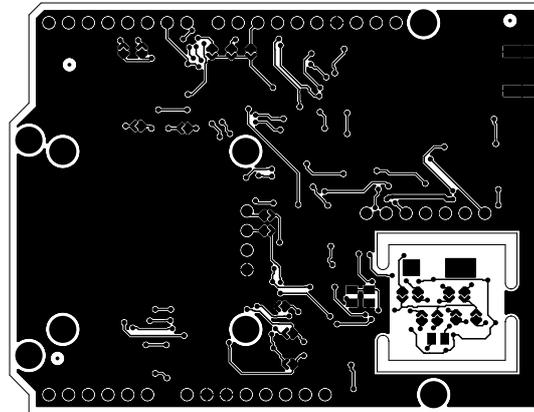
CALID.				MATERIAL

N. DE DIBUJO
ARDUINO_LORA_SENSOR_SHIELD_TOP

ESCALA 1:1

A4

HOJA 1 DE 1



SI NO SE INDICA LO CONTRARIO:
LAS COTAS SE EXPRESAN EN MM
TOLERANCIAS
LINEAL
ANGULAR:

ACABADO:

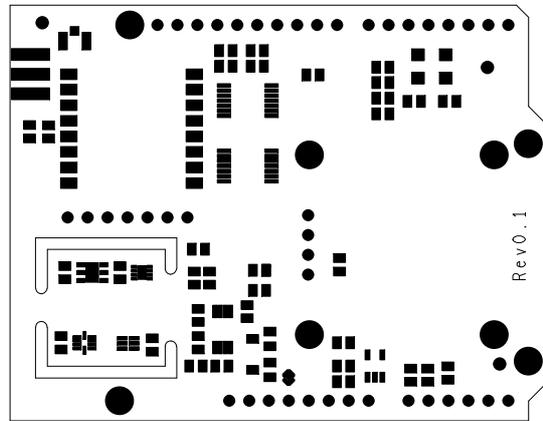
NO CAMBIE LA ESCALA

REVISION

Plano PCB 5

	NOMBRE	FIRMA	FECHA			
DIBUJ.	J. Laborda		27/01/18			
VERIF	A. Perles		05/06/18			
APROB	J. Laborda		05/06/18			
FABR.						
CALID.				MATERIAL		
				PESO:		

TITULO:		A4
COPPER BOTTOM		
N. DE DIBUJO		
ARDUINO_LORA_SENSOR_SHIELD_BOTTOM		
ESCALA 1:1	HOJA 1 DE 1	



SI NO SE INDICA LO CONTRARIO:
LAS COTAS SE EXPRESAN EN MM
TOLERANCIAS
LINEAL
ANGULAR:

ACABADO:

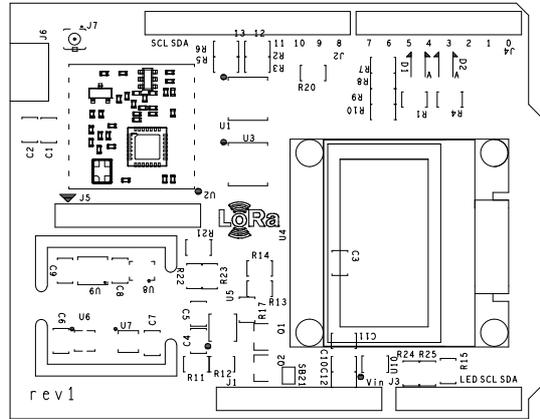
NO CAMBIE LA ESCALA

REVISION

Plano PCB 6

	NOMBRE	FIRMA	FECHA		
DIBUJ.	J .Laborda		27/01/18		
VERIF	A. Perles		05/06/18		
APROB	J .Laborda		05/06/18		
FABR.					
CALID.				MATERIAL	
				PESO:	

TITULO:	SOLDERMASK TOP
N. DE DIBUJO	
ARDUINO_LORA_SENSOR_SHIELD_TMK	A4
ESCALA 1:1	HOJA 1 DE 1



SI NO SE INDICA LO CONTRARIO:
 LAS COTAS SE EXPRESAN EN MM
 TOLERANCIAS
 LINEAL
 ANGULAR:

ACABADO:

NO CAMBIE LA ESCALA

REVISION

Plano PCB 8

TITULO:

SILKSCREEN TOP

N. DE DIBUJO

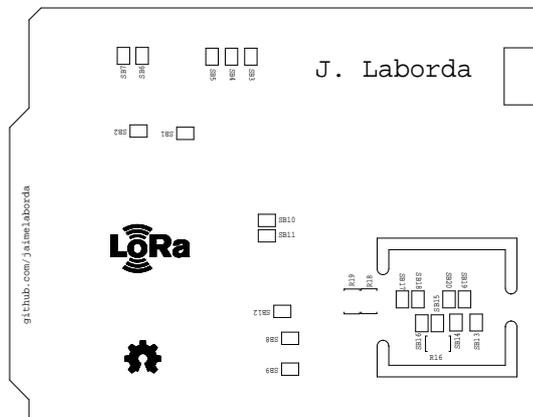
ARDUINO_LORA_SENSOR_SHIELD_TSK

A4

PESO:

ESCALA 1:1

HOJA 1 DE 1



SI NO SE INDICA LO CONTRARIO:
LAS COTAS SE EXPRESAN EN MM
TOLERANCIAS
LINEAL
ANGULAR:

ACABADO:

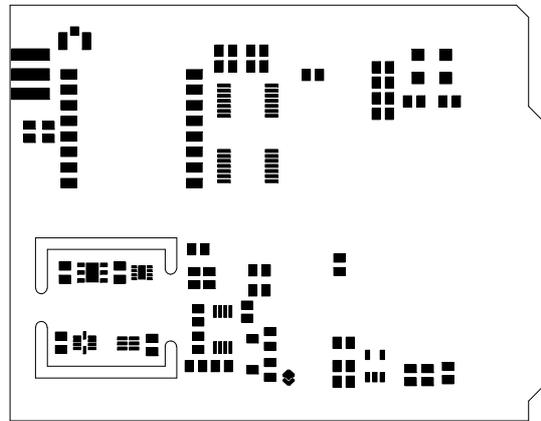
NO CAMBIE LA ESCALA

REVISION

Plano PCB 9

	NOMBRE	FIRMA	FECHA			
DIBUJ.	J. Laborda		27/01/18			
VERIF	A. Perles		05/06/18			
APROB	J. Laborda		05/06/18			
FABR.						
CALID.				MATERIAL		
				PESO:		

TITULO:		A4
SILKSCREEN BOTTOM		
N. DE DIBUJO		
ARDUINO_LORA_SENSOR_SHIELD_BSK		
ESCALA 1:1	HOJA 1 DE 1	



SI NO SE INDICA LO CONTRARIO:
 LAS COTAS SE EXPRESAN EN MM
 TOLERANCIAS
 LINEAL
 ANGULAR:

ACABADO:

NO CAMBIE LA ESCALA

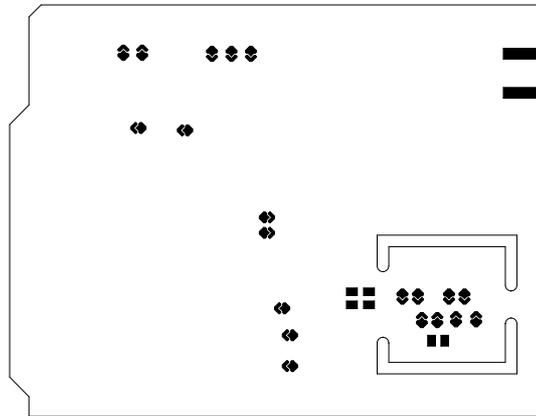
REVISION

Plano PCB 10

	NOMBRE	FIRMA	FECHA		
DIBUJ.	J .Laborda		27/01/18		
VERIF	A. Perles		05/06/18		
APROB	J .Laborda		05/06/18		
FABR.					
CALID.				MATERIAL	
				PESO:	

TITULO:	SOLDERPASTE TOP
N. DE DIBUJO	
ARDUINO_LORA_SENSOR_SHIELD_SPT	
ESCALA 1:1	HOJA 1 DE 1

A4



SI NO SE INDICA LO CONTRARIO:
LAS COTAS SE EXPRESAN EN MM
TOLERANCIAS
LINEAL
ANGULAR:

ACABADO:

NO CAMBIE LA ESCALA

REVISION

Plano PCB 11

Plano PCB

TITULO:

SOLDERPASTE BOTTOM

N. DE DIBUJO

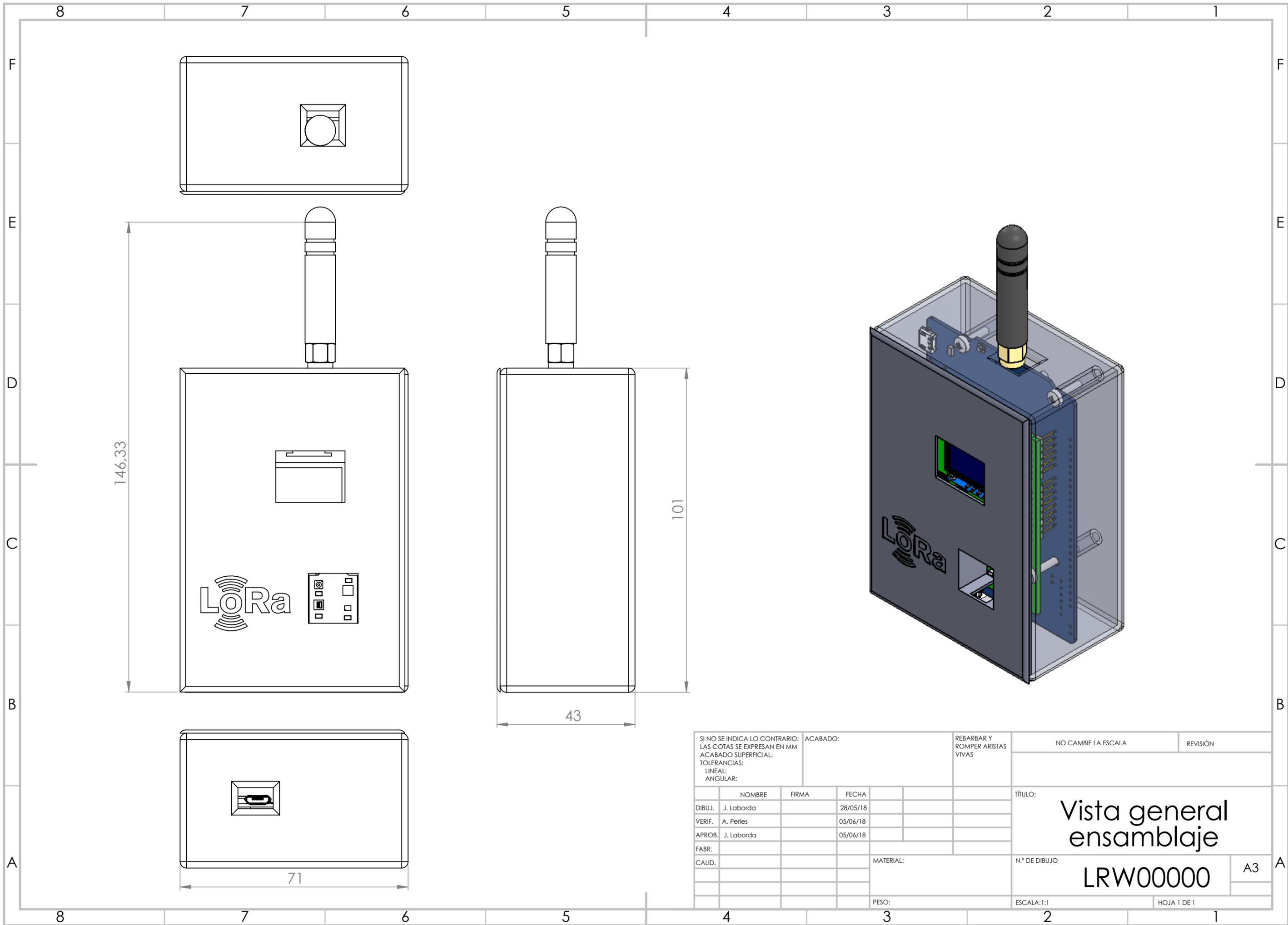
ARDUINO_LORA_SENSOR_SHIELD_SPB

A4

PESO:

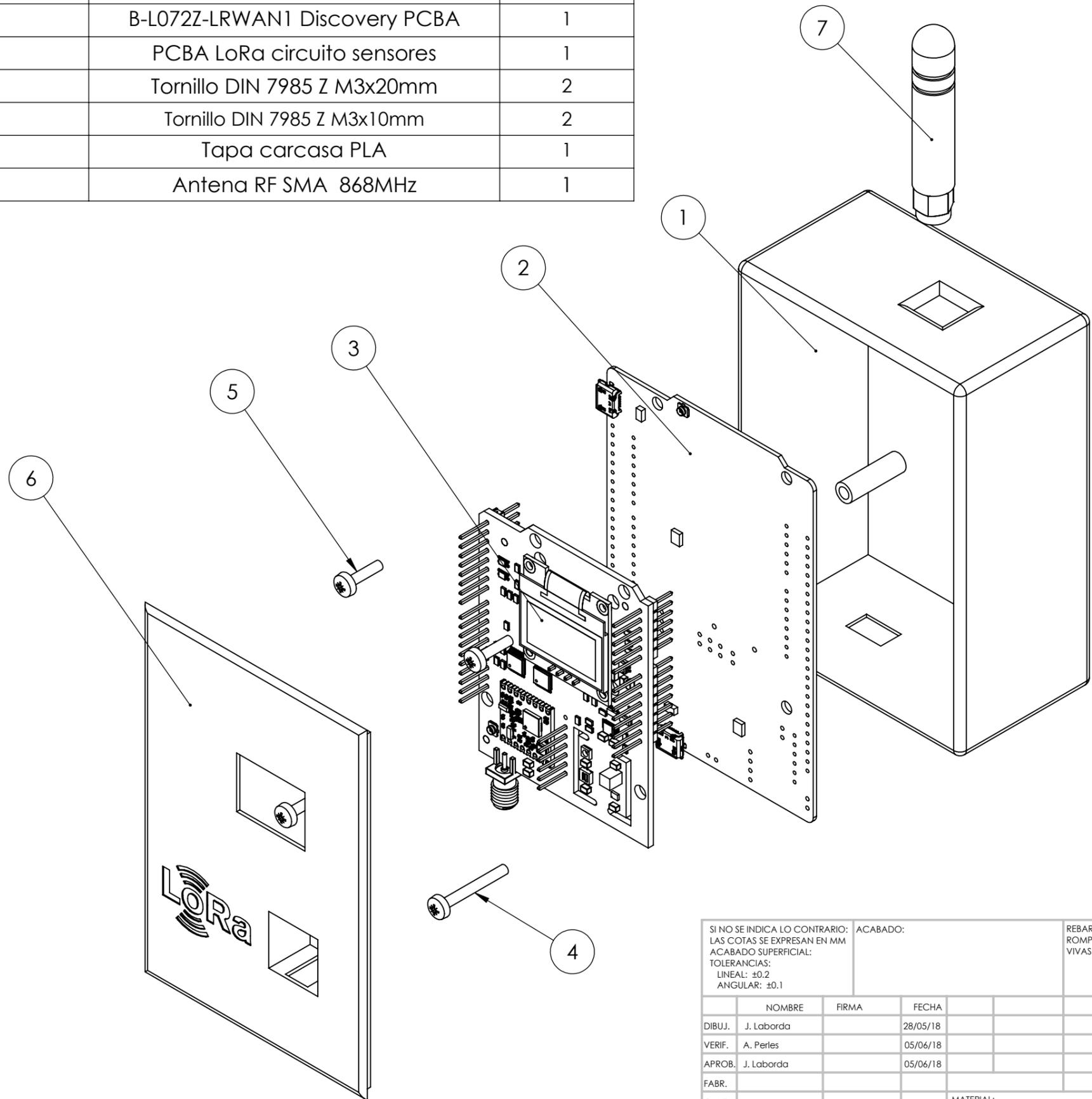
ESCALA 1:1

HOJA 1 DE 1

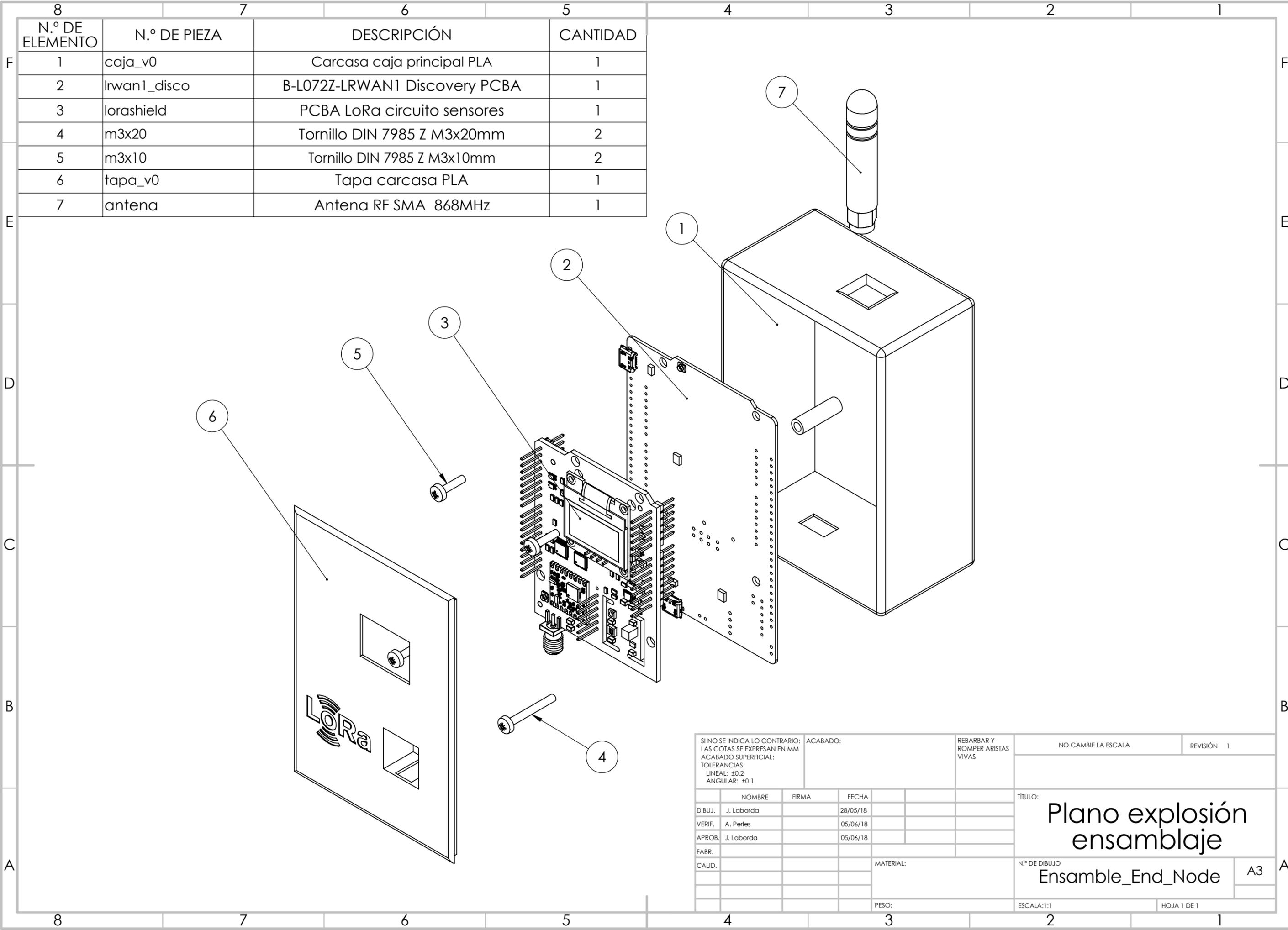


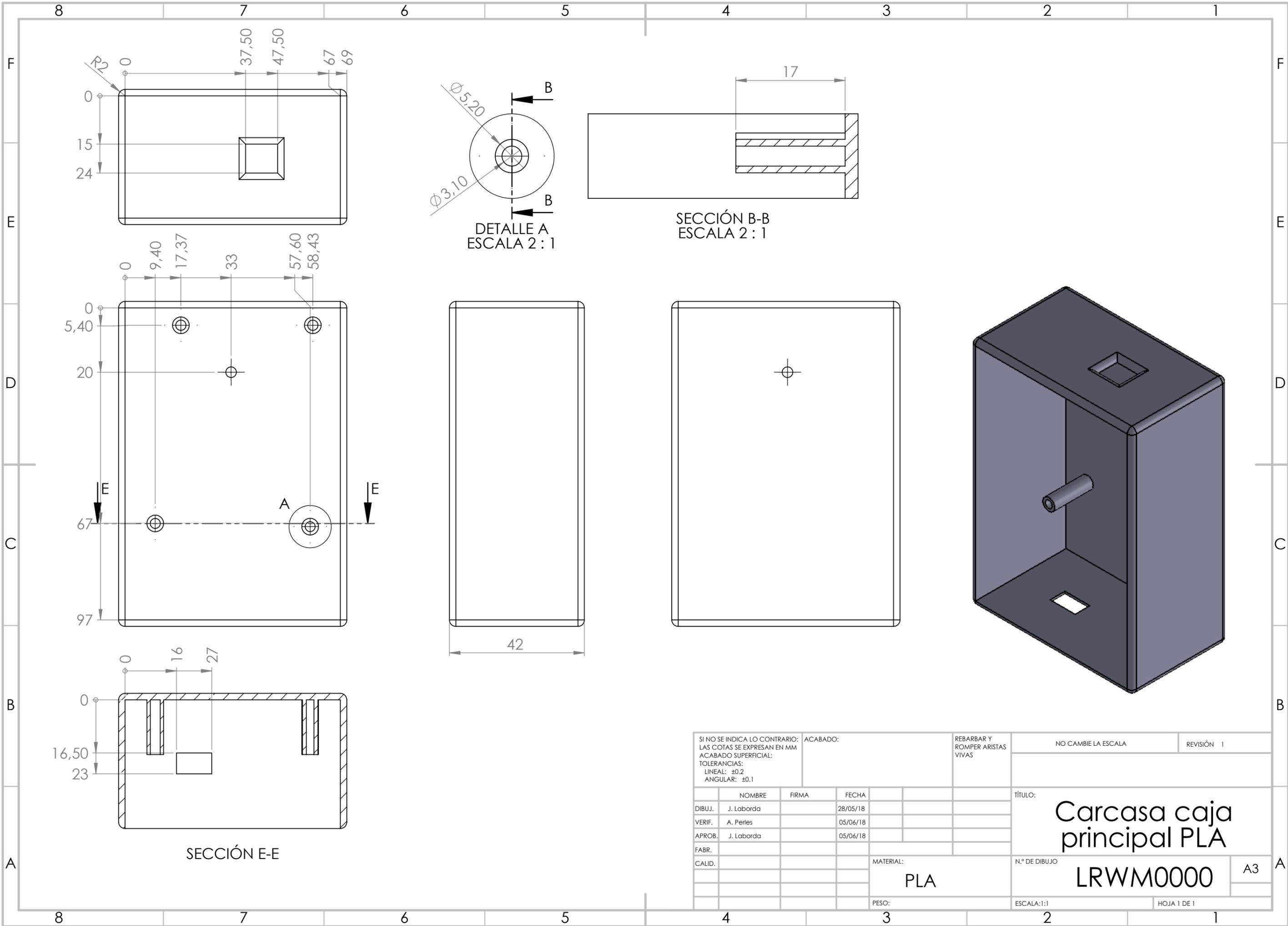
SI NO SE INDICA LO CONTRARIO: LAS COTAS SE EXPRESAN EN MM ACABADO SUPERFICIAL: TOLERANCIAS: LINEAL: ANGULAR:			ACABADO:	REBARBAR Y ROMPER ARISTAS VIVAS	NO CAMBIE LA ESCALA	REVISIÓN
DIBUJ. J. Laborda				FECHA	TÍTULO:	
VERIF. A. Perles				05/06/18	Vista general ensamblaje	
APROB. J. Laborda				05/06/18	N.º DE DIBUJO	
FABR.				LRW00000		
CALID.				MATERIAL:	A3	
PESO:				ESCALA:1:1		
HOJA 1 DE 1						

N.º DE ELEMENTO	N.º DE PIEZA	DESCRIPCIÓN	CANTIDAD
1	caja_v0	Carcasa caja principal PLA	1
2	lrwan1_disco	B-L072Z-LRWAN1 Discovery PCBA	1
3	lorashield	PCBA LoRa circuito sensores	1
4	m3x20	Tornillo DIN 7985 Z M3x20mm	2
5	m3x10	Tornillo DIN 7985 Z M3x10mm	2
6	tapa_v0	Tapa carcasa PLA	1
7	antena	Antena RF SMA 868MHz	1

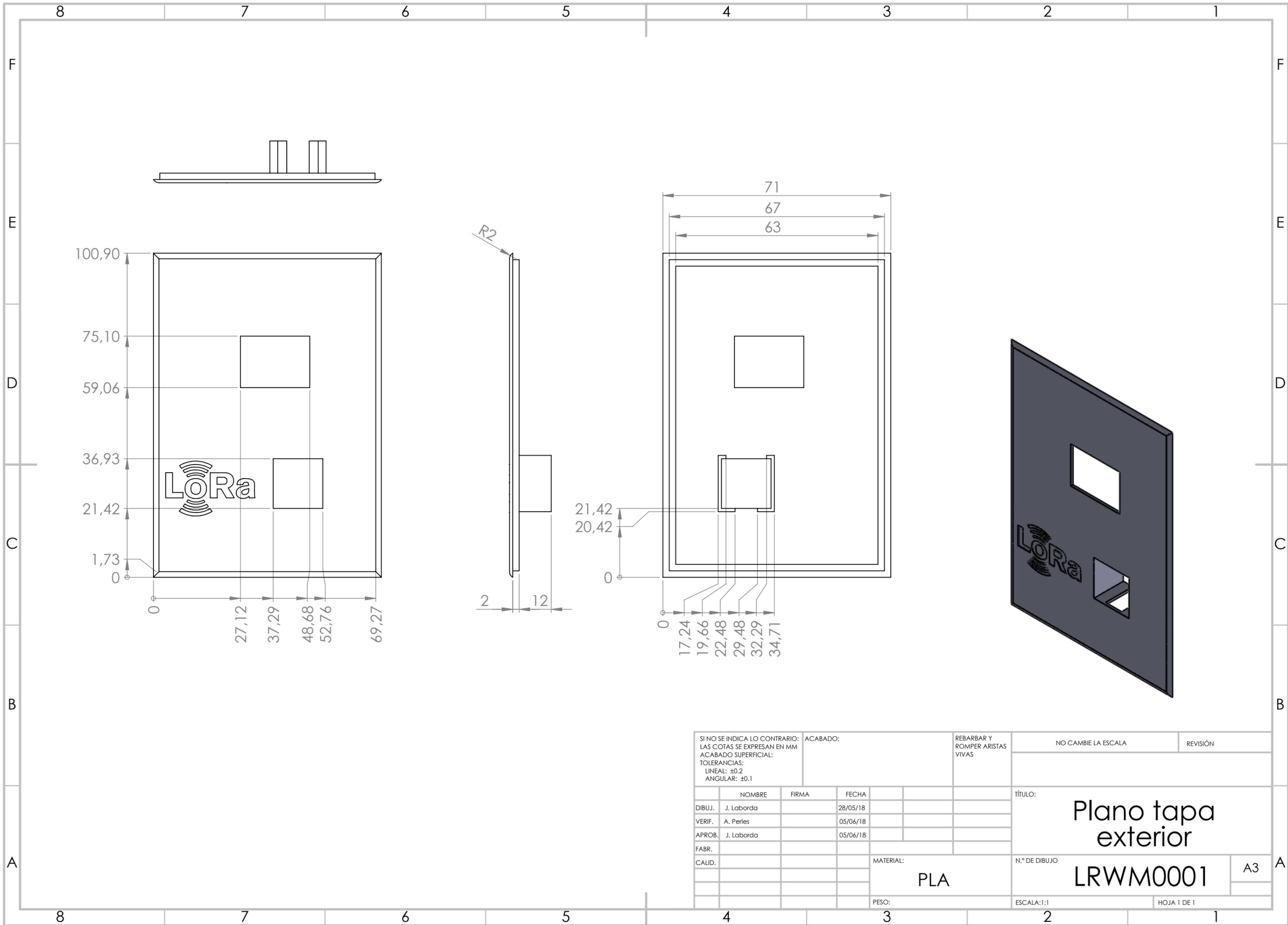


SI NO SE INDICA LO CONTRARIO: ACABADO: LAS COTAS SE EXPRESAN EN MM ACABADO SUPERFICIAL: TOLERANCIAS: LINEAL: ±0.2 ANGULAR: ±0.1			REBARBAR Y ROMPER ARISTAS VIVAS		NO CAMBIE LA ESCALA	REVISIÓN 1
					TÍTULO: Plano explosión ensamblaje	
					N.º DE DIBUJO Ensamble_End_Node	
					A3	
					ESCALA:1:1	
					HOJA 1 DE 1	





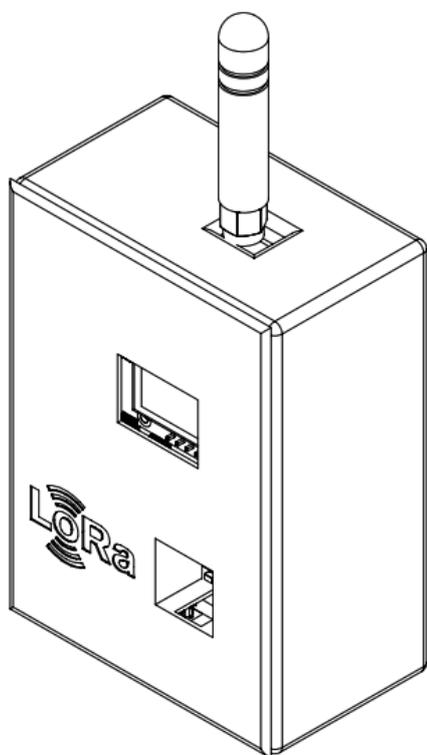
SI NO SE INDICA LO CONTRARIO: LAS COTAS SE EXPRESAN EN MM ACABADO SUPERFICIAL: TOLERANCIAS: LINEAL: $\pm 0,2$ ANGULAR: $\pm 0,1$			ACABADO:	REBARBAR Y ROMPER ARISTAS VIVAS	NO CAMBIE LA ESCALA	REVISIÓN 1
DIBUJ. J. Laborda			FIRMA	FECHA 28/05/18	TÍTULO: Carcasa caja principal PLA	
VERIF. A. Perles				05/06/18	N.º DE DIBUJO LRWM0000	
APROB. J. Laborda				05/06/18	A3	
FABR.					ESCALA:1:1	
CALID.				MATERIAL: PLA	HOJA 1 DE 1	
				PESO:		



SI NO SE INDICA LO CONTRARIO: LAS COTAS SE EXPRESAN EN MM ACABADO SUPERFICIAL: TOLERANCIAS: LINEAL: ±0.2 ANGULAR: ±0.1			ACABADO:		REBARBAR Y ROMPER ARISTAS VIVAS		NO CAMBIE LA ESCALA		REVISIÓN		
						TÍTULO:					
						Plano tapa exterior					
						N.º DE DIBUJO					
						LRWM0001					
						ESCALA:1:1					
						HOJA 1 DE 1					
NOMBRE		FIRMA		FECHA							
DIBUJ. J. Laborda				28/05/18							
VERIF. A. Perles				05/06/18							
APROB. J. Laborda				05/06/18							
FABR.											
CALID.											
						MATERIAL:					
						PLA					
						PESO:					

DESARROLLO DE UN NODO INALÁMBRICO DE BAJO CONSUMO BASADO EN TÉCNICAS DE IOT PARA MONITORIZACIÓN DE PATRIMONIO CULTURAL

3. PLIEGO DE CONDICIONES



Autor:
D. Jaime Laborda Macario
Tutor:
D. Ángel Perles Ivars

Valencia, junio de 2018

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para
monitorización de patrimonio cultural

Contenido del pliego de condiciones

1. Definición y alcance del pliego	1
2. Condiciones y normas de carácter general	1
2.1. Instalación	1
2.2. Seguridad	2
2.3. Utilización	2
2.4. Mantenimiento	3
3. Condiciones particulares y fabricación	3
3.1. Diseño PCB	3
Fabricación PCB	4
3.2. Ensamblaje PCBA	4
3.3. Componentes electrónicos	5

Pliego de condiciones

Glosario de términos

CAD	Diseño asistido por computadora
CE	Conformidad Europea
Gerber	Formato de archivo para fabricación de PCB
IOT	Internet de las cosas
IPC	Asociación de estándares para la industria electrónica
JEDEC	Asociación independiente de estandarización de circuitos integrados
LoRaWAN	Long Range Wide Area Network
PCB	Tarjeta de circuito impreso
ROHS	Restricción de Sustancias Peligrosas
UE	Unión Europea
UL	Underwriters Laboratories

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

1. Definición y alcance del pliego

El presente proyecto supone el diseño e implementación de un nodo sensor inalámbrico de bajo consumo, que mediante técnicas de IoT y haciendo uso de una red LoRaWAN, transmite sin cables las medidas de temperatura, humedad y luminosidad ambiental.

El presente pliego de condiciones pretende describir las condiciones bajo las cuales se debe fabricar, reproducir y utilizar el dispositivo cuyo diseño se describe en la memoria, así como en los planos y demás documentación técnica.

Es importante recalcar que para la completa documentación del presente trabajo es necesario disponer de los archivos CAD y *Gerbers* para la fabricación de la electrónica, así como el código fuente y los binarios para programar el hardware.

2. Condiciones y normas de carácter general

El sensor inalámbrico ha sido diseñado bajo unas condiciones de funcionamiento y debe utilizarse y fabricarse de acuerdo con estas.

2.1. Instalación

Al tratarse de un prototipo en fase de pruebas, la instalación ha de efectuarse por un personal cualificado que conozca el producto de primera mano. Esto es debido a que el sensor ha de instalarse bajo unas condiciones específicas para su correcto funcionamiento.

Para evitar la interferencia y que las medidas del sensor sean lo más fiables y certeras posibles, se deberá evitar su colocación junto a fuentes de calor y/o frío que puedan afectar a su funcionamiento. De esta manera, evitaremos su instalación junto a aparatos que desprendan calor propio como lamparas, bombillas o salidas de ventilación, ya que estas pueden inferir perjudicialmente en las medidas ambientales del sensor. Tampoco deberá instalarse en carcasas metálicas o jaulas que impidan la correcta dispersión de las ondas de radio.

Se deberá realizar un estudio previo a su colocación en la obra de arte con la intención de buscar el lugar ideal que represente un compromiso entre fiabilidad de los parámetros leídos e impacto visual en la obra. En caso de duda se deberá consultar con el autor del proyecto o a un experto conservador.

La instalación de la antena deberá ser efectuada en una situación que no interfiera con el correcto funcionamiento de otros dispositivos de radio como WiFi o TDT, y previa autorización de los organismos componentes. No obstante, al trabajar en el espectro ISM libre de licencia de la banda 868MHz no se deberá licenciar la instalación. La instalación del punto de acceso deberá ser efectuada por personal cualificado y haciendo uso de una antena unidireccional certificada respetando los niveles de potencia. No se garantiza el correcto funcionamiento de la red con un punto de acceso, router y/o diferente al descrito en la memoria del proyecto.

Pliego de condiciones

2.2. Seguridad

El dispositivo está catalogado como seguro, pues se alimenta a través de baterías de bajo voltaje no peligrosas. No hay riesgo de explosión ni fuego. Las baterías cumplen con la normativa UL94 de inflamabilidad y la MH29980 de transporte. El dispositivo debe usarse con las baterías para las que ha sido diseñado especificadas en la documentación del mismo, y no se garantiza tanto el funcionamiento como la seguridad con un uso fuera de las especificaciones descritas.

A este dispositivo se le aplica la directiva europea del mercado CE y en específico las siguientes normativas de obligado cumplimiento:

- 206/95/CE de baja tensión y seguridad general
- 2004/108/CE y 2014/30/UE de compatibilidad electromagnética donde se miden y ensayan las emisiones o perturbaciones electromagnéticas e inmunidad conducidas o radiadas según los materiales y el entorno de trabajo de la aplicación.
- 1995/5/CE de Radio y Telecomunicaciones

Y en específico las normativas propias de aparatos radioeléctricos al tratarse de un sensor inalámbrico como la Directiva 2014/53/UE (NB nº. 0370) y la 2008/07/11/BOE sobre dispositivos de radiofrecuencia operando en la banda de 865-868MHz.

El chip de Murata que integra el transceptor de Semtech SX1276 está certificado por el mercado CE así como por la LoRa Alliance. Cualquier modificación o sustitución de este chip no garantiza la correcta compatibilidad del dispositivo.

2.3. Utilización

El dispositivo se conecta a la red automáticamente una vez recibe alimentación. Puede tardar hasta 5-10 minutos en identificarse correctamente en el servidor y en empezar a tener datos disponibles en la plataforma. Para la correcta conexión con el servidor, el dispositivo será suministrado con las claves de identificación de red precargadas. Si los parámetros de aplicación varían, se ha de volver a configurar el sensor, obligando a un acceso físico al mismo para su reflasheo.

El dispositivo hace uso de una red LoRaWAN de uso libre y comunitario que no es propiedad de la empresa o autor del proyecto y, por tanto, no se garantiza su disponibilidad y funcionamiento las 24 horas del día, pudiendo producirse bajas por mantenimiento de servidores o similares.

Al tratarse de un prototipo experimental, no se garantiza su correcto funcionamiento, pudiendo dejar de funcionar en cualquier momento sin previo aviso interrumpiendo el servicio. Se recomienda, por tanto, que no se utilice como sustituto a un método de monitorización tradicional.

Para más detalles sobre el funcionamiento y utilización del equipo, póngase en contacto con el autor del proyecto.

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

2.4. Mantenimiento

El único mantenimiento del dispositivo es el cambio de baterías. Estas deberán ser sustituidas a los 5 años o antes independientemente de que el dispositivo no avise de baja batería. Las baterías usadas serán desechadas correctamente según la normativa pertinente **2006/66/CE** sobre pilas y acumuladores y la gestión ambiental de sus residuos o sustituta de la misma.

Es posible que en ambientes de polvo o derivados sea necesario una limpieza del sensor, especialmente de luminosidad, cada cierto tiempo para seguir garantizando el correcto funcionamiento.

Al tratarse de un dispositivo prototipo experimental en pruebas es posible que se tengan que realizar cambios en el firmware para corrección de anomalías o introducción de mejoras y funcionalidades. El flasheo de un nuevo firmware ha de realizarse in-situ y obliga a una conexión física USB con el dispositivo. No se recomienda la instalación en lugares de difícil acceso hasta pasado el periodo de pruebas.

3. Condiciones particulares y fabricación

3.1. Diseño PCB

El diseño de la tarjeta de circuito impreso ha sido realizado de acuerdo con la normativa IPC vigente para el diseño de circuitos impresos:

- **IPC-2221A** de diseño genérico de tarjetas de circuito impreso
- **IPC-2222** específica para tarjetas rígidas
- **IPC-7351B** que especifica los requerimientos para componentes de montaje superficial y define el tamaño de huella en la placa de circuito impreso
- **IPC-2252**, guía de diseño para circuitos impresos de radio frecuencia o microondas.

Esta conformidad garantiza la correcta fabricación y ensamblaje de la placa electrónica.

Pliego de condiciones

Fabricación PCB

Están son las especificaciones bajo las que se debe fabricar la placa de circuito impreso o PCB

- El dieléctrico debe ser FR4 con un grosor total de 1.6mm de acuerdo a las especificaciones de *stack-up* definidas en el plano de fabricación correspondiente. Los parámetros de ancho son recomendados y podrán ser ligeramente ajustados por el fabricante del PCB con la intención de acercarse a valores comerciales.
- La tarjeta debe ser fabricada de acuerdo a la normativa **IPC-6012B – Qualification and performance specifications for rigid printed circuit boards** y **IPC-600** o revisión más actualizada. Cualquier desviación de esta norma debe ser aprobada por el autor.
- La tarjeta será fabricada a 2 capas de acuerdo con la información de fabricación aportada
- El acabado final del PCB debe ser ENIG de acuerdo con la especificación IPC-4552 o revisión más actual con un acabado dorado de 0.050um.
- La máscara de soldadura debe ser verde y de tipo LPISM
- La serigrafía debe ser de color blanco a dos capas. Se acepta la distorsión entre pistas. No se acepta serigrafía sobre *pads*. El fabricante debe aplicar una macro para eliminar la máscara de soldadura si lo estima adecuado.
- La tarjeta debe entregarse totalmente testeada bajo un test eléctrico de continuidad correspondiente que asegure la correcta continuidad y aislamiento entre pistas de acuerdo con la *netlist* proporcionada.
- Cualquier anomalía detectada en el diseño durante la fase de inspección en oficina técnica debe ser reportada formalmente al autor y el trabajo será pausado hasta que se resuelvan las anomalías. No se fabricará sin aprobación expresa y por escrito del ingeniero.
- Las tarjetas serán entregadas sueltas, sin panel.
- La PCB debe llevar la señalización UL y fecha marcada en cobre o máscara de soldadura, nunca en la serigrafía. No se permiten otras marcas. Nunca eliminar cobre de los planos externos.
- La tarjeta debe cumplir con la normativa ROHS correspondiente y la directiva europea 2011/65/EU
- Las tarjetas deben ser entregadas junto con un Certificado de Conformidad

Cualquier desviación de los parámetros especificados debe ser aprobada por el autor o ingeniero responsable. La no conformidad con estas especificaciones puede ser razón de rechazo de las PCBs por parte del cliente.

3.2. Ensamblaje PCBA

La tarjeta electrónica será ensamblada de acuerdo con la normativa IPC-A-610E de aceptabilidad de ensamblajes electrónicos por una empresa ensambladora o técnico certificado.

Las soldaduras se realizarán de acuerdo con la norma J-STD-0001F que define los requerimientos para soldadura eléctrica y ensamblajes electrónicos

Los procesos de soldadura serán libres de alójenos tales como plomo, mercurio o cadmio, de acuerdo con la normativa ROHS (2002/95/CE) y la empresa montadora estará obligada a proporcionar un certificado de conformidad que así lo acredite.

La tarjeta será soldada por horno de acuerdo con el perfil de temperatura más conservador o, en su defecto, de acuerdo con el perfil especificado en la *Figura 1*. Nunca se superarán los 250 grados centígrados en el proceso de horneado de la tarjeta electrónica.

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

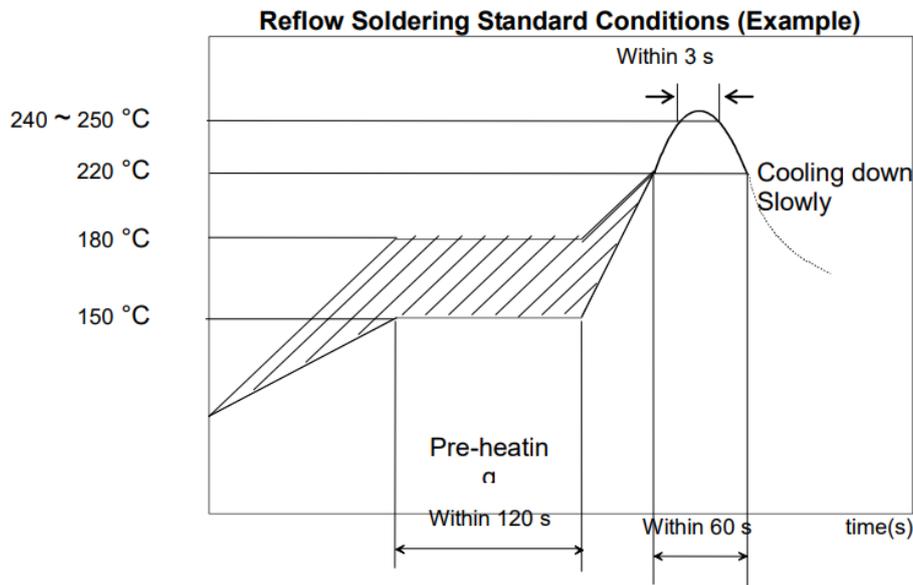


Figura 1 - Perfil de temperatura recomendado

3.3. Componentes electrónicos

Los componentes electrónicos usados en el montaje serán las referencias comerciales descritas en la lista de materiales electrónicos proporcionada en la documentación del proyecto o similares.

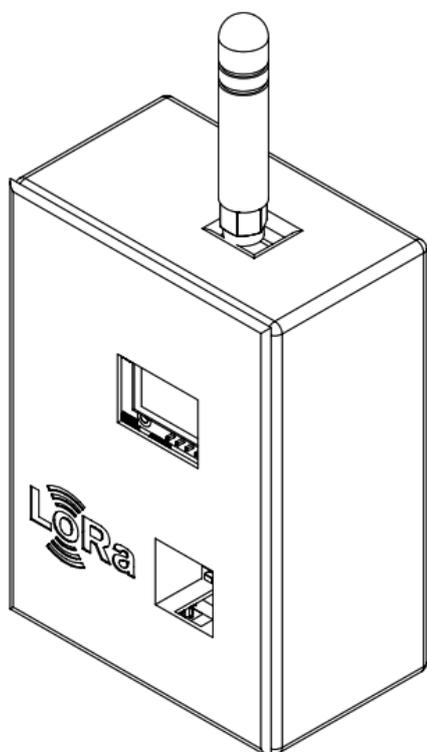
Cualquier modificación o alternativo propuesto debe ser aprobada por el autor del proyecto o ingeniero responsable pertinente.

Los componentes usados deberán cumplir con la normativa de Restricción de Sustancias Peligrosas en aparatos eléctricos y electrónicos (ROHS) correspondiente y venir acompañados de un certificado de conformidad que así lo acredite.

Los componentes sensibles a humedad tales como circuitos integrados, microcontroladores o diodos luminosos deberán ser distribuidos en bolsas cerradas herméticamente al vacío que garanticen la correcta permeabilidad del componente de acuerdo con la normativa **IPC/JEDEC J-STD-020**. Deberán ser secados en horno durante 48 horas a 125 ± 5 °C (**J-STD-033**) o lo especificado por el fabricante si su humedad durante el transporte es superior al 60%.

DESARROLLO DE UN NODO INALÁMBRICO DE BAJO CONSUMO BASADO EN TÉCNICAS DE IOT PARA MONITORIZACIÓN DE PATRIMONIO CULTURAL

4. PRESUPUESTO



Autor:
D. Jaime Laborda Macario
Tutor:
D. Ángel Perles Ivars

Valencia, junio de 2018

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

Contenido del presupuesto

1. Sobre el presupuesto.....	1
2. Materiales	2
2.1. Componentes electrónicos	2
2.2. Fabricación circuito impreso	3
2.3. Carcasa impresión 3D PLA	4
2.4. Punto acceso y servidor	4
3. Mano de obra	5
4. Amortización de los equipos	6
5. Resumen	7

Glosario de términos

EA	Each (por unidad)
IVA	Impuesto sobre el Valor Añadido
PVP	Precio Venta Público
UNS	Unidades
UOM	<i>Unit of measure</i> (unidad de medida)
VPS	<i>Virtual Private Server</i> (Servidor virtual privado)

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

1. Sobre el presupuesto

El presupuesto del proyecto se ha realizado principalmente en dos fases; la primera representa el desarrollo de la electrónica y construcción del prototipo funcional con todos los costes asociados relativos a los materiales y componentes electrónicos y las horas de trabajo de desarrollo, diseño y montaje. Para la segunda fase se ha considerado el despliegue de la red LoRaWAN y la aplicación web de usuario teniendo en cuenta los gastos de material que representa y los costes relacionados con el servidor web. No se ha considerado en esta fase los gastos de instalación y configuración del punto de acceso LoRaWAN.

Los precios de los componentes electrónicos y piezas comerciales se han cotizado en diferentes distribuidores autorizados de equipos de electrónica como Mouser.es, Digikey y Arrow Electronics a fecha de junio de 2018. Se trata de un mercado muy cambiante por lo que los costes pueden variar con el tiempo e incluso las piezas pueden haber quedado obsoletas y será necesario un rediseño de las mismas. El presente presupuesto tiene una validez y garantía de un año.

Los gastos relativos a la mano de obra se han estimado en cada caso según las horas que han sido invertidas.

El presupuesto deberá ir acompañado de la memoria del proyecto y tan solo contempla lo descrito en ella, cualquier requisito que se salga de ella puede tener costes adicionales no reflejados en el presente presupuesto.

Presupuesto

2. Materiales

2.1. Componentes electrónicos

Precios cotizados en Mouser.es a fecha de junio de 2018

Referencia	Cantidad	Descripción	Un	Fabricante	Ref Fabricante	Precio	Total
LRWEC000	6	Condensador ceramico 100nF 10V X7R 20% 0805	EA	Murata	GRM219R71C104MA01J	0,090 €	0,54 €
LRWED000	1	Diodo LED Rojo SMD 1206	EA	Kingbright	APT3216EC	0,131 €	0,13 €
LRWED001	1	Diodo LED Verde SMD 1207	EA	Kingbright	APT3216SGC	0,139 €	0,14 €
LRWEJ000	2	Con 8-pin paso 2.54mm Fila única PTH Header	EA	Samtec	TSW-110-08-L-S	1,100 €	2,20 €
LRWEJ001	1	Con 10-pin paso 2.54mm Fila única PTH Header	EA	Samtec	TSW-110-10-L-S	1,100 €	1,10 €
LRWEJ002	1	Con 6-pin paso 2.54mm Fila única PTH Header	EA	Samtec	TSW-110-06-L-S	1,100 €	1,10 €
LRWEJ003	1	Con RF UFL 50Ohm Mini SMD	EA	Hirose	U.FL-R-SMT-1(10)	0,650 €	0,65 €
LRWEQ000	2	Mosfet N Chan 60V 300mA SOT23	EA	DiodesZetex	2N7002E-7-F	0,221 €	0,44 €
LRWER000	2	Resistencia 1K 1% 0805	EA	TE Connectivity	CRG0805F1K0	0,012 €	0,02 €
LRWER001	2	Resistencia 2K2 1% 0805	EA	TE Connectivity	CRG0805F2K2	0,012 €	0,02 €
LRWER002	9	Resistencia 10K 1% 0805	EA	TE Connectivity	CRG0805F10K	0,012 €	0,11 €
LRWER003	5	Resistencia 47K 1% 0805	EA	Yageo	RC0805FR-0747KL	0,082 €	0,41 €
LRWEU000	2	IC Quad Buffer 3-state driver TSSOP14	EA	Nexperia	74LVC125APW,118	0,328 €	0,66 €
LRWEU001	1	Módulo RF LoRa Transceptor 868MHz SMD16	EA	RFM95	HopeRF	5,000 €	5,00 €
LRWEU002	1	Pantalla Monocromo OLED 0.96 128x64px blanca i2c	EA	Heltec Automation	HTDS-WI96	3,000 €	3,00 €
LRWEU004	1	IC Ambient Lux Sensor i2c DFN6	EA	ams	TSL25911FN	1,800 €	1,80 €
LRWEU005	1	IC Temperature&RH Sensor i2c 0.3C DFN8	EA	Sensirion	SHT31-DIS-B	4,760 €	4,76 €
LRWO0000	1	B-L072Z-LRWAN1 Discovery PCBA	EA	STMicroelectronics	B-L072Z-LRWAN1	39,650 €	39,65 €
LRWO0001	2	Tornillo DIN 7985 Z M3x20mm	EA	Bossard	3060721	0,020 €	0,04 €
LRWO0002	2	Tornillo DIN 7985 Z M3x10mm	EA	Bossard	3060718	0,020 €	0,04 €
Total							61,81 €

Tabla 1 - Presupuesto componentes electrónicos

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

2.2. Fabricación circuito impreso

El circuito impreso forma parte de una de las piezas customizadas del proyecto y, por tanto, se ha de encargar a fabricar a uno de los fabricantes que cumpla con la normativa de fabricación de circuitos impresos pertinente (IPC-600). Al tratarse de un prototipo, se busca un fabricante económico que permita reducir el coste de desarrollo del producto. Se ha seleccionado el fabricante chino AllPCB.

El precio se calcula solicitando una cotización al fabricante AllPCB según las medidas y el acabado que se indican en la *Tabla 2*.

Característica	Opción
Dimensiones	54x67mm
Cantidad	10 unidades (el mínimo)
Tipo de dieléctrico	FR-4 (ShengYiTg140)
Número de capas	2
Ancho final del PCB	1.6mm
Ancho de cobre	1oz (35µm)
Agujero pasante mínimo	0.3mm
Mascara de soldadura	Si, Verde
Serigrafía	Si, Blanca
Test eléctrico	Si

Tabla 2 - Parámetros de fabricación PCB

El coste para fabricar diez placas de circuito impreso con estas características asciende a 26\$ (envío incluido), que según la conversión a Euros del día 9 de junio de 2018 son 22,07€. Es decir **2,2 € por unidad**.

Charge Details

Project	Panel	Board	Film	HASL	Testing	Color	Others
12.75	0.00	5.26	7.65	0.00	0.00	0.00	0.34

PCB price and delivery Current Time (GMT+8): 2018-06-10 01:42:58

lead time	Quantity	Price
TopSpeed 24 hours	10	26.00

Shipping by

SPAIN

DHL  3-7 days wt : 0.62 kg



Estimated Shipping Time
2018-06-12



Estimated Arriving Time
2018-6-19

PCB Cost (Product Cost):	\$26.00
Shipping Cost:	Free
Subtotal:	\$26.00

Figura 1 - Cotización AllPCB

Presupuesto

2.3. Carcasa impresión 3D PLA

La carcasa ha sido impresa en 3D para este primer prototipo ya que permite abaratar costes y reducir el “time-to-market” al no depender de un tercero que lo tenga que producir.

Se ha utilizado una impresora 3D FDM de deposición de material fundido tipo Prusa Mendel propiedad del diseñador. Para realizar la estimación de coste de impresión de las piezas se tiene en cuenta la cantidad de material PLA en Kg necesaria para la fabricación de cada una de las piezas y el gasto energético que representa tener la máquina encendida las horas de impresión necesarias para la impresión de cada una de las piezas.

Un	Descripción	Parcial	Cantidad	Precio	Total
mm	PLA filamento blanco 1.75mm 1.25gcm3 330m (24€/kg)		17329	0,0001 €	1,6770 €
	LRWM0000 - Carcasa caja principal PLA	14340			
	LRWM0001 - Tapa carcasa PLA	2989			
h	3D printing cost for manufacturing (printer hours)		5,5	0,2500 €	1,3750 €
	LRWM0000 - Carcasa caja principal PLA	3,75			
	LRWM0001 - Tapa carcasa PLA	1,75			
EA	Coste de preparación y calibración (por pieza)		2	3,0000 €	6,0000 €
Total					9,0520 €

Tabla 3 - Estimación impresión 3D FDM

El gasto de impresión 3D de la carcasa de protección para la electrónica asciende a **9,05 €**.

2.4. Punto acceso y servidor

Uno de los aspectos diferenciadores del proyecto ha sido el despliegue de la red LoRaWAN no existente hasta el momento. Para ello es necesario la compra de equipos como el router o concentrador, el accesorio mCard para la recepción LoRa y la antena. Para más información al respecto referirse al capítulo correspondiente de la memoria.

Para el desarrollo de la aplicación final es necesario el alquiler de un servidor o *hosting* conectado a Internet para proveer los servicios de red al usuario. Este ha de ser compatible con las aplicaciones de red necesarias y será necesario que corra un sistema operativo Linux. Se ha decidido contratar los servicios de DigitalOceans con un servidor básico.

Cant	Descripción	Un	Fabricante	Ref Fabricante	Precio	Total
1	Multiconnect Conduit WiFi+GNSS US/EU/UK	EA	MultiTech Systems	MTCDT-H5-247L-US-EU-GB	467,30 €	467,30 €
1	MultiConnect mCard Accesorio LoRa 868MHz SPI	EA	MultiTech Systems	MTAC-LORA-H-868	164,73 €	164,73 €
1	Antena RF SMA 868MHz 3,2dBi	EA	Taoglas Limited	TI.18.3113	11,62 €	11,62 €
1	Adaptador RF Coaxial SMA a RP-SMA	EA	Amphenol RF	132171RP	5,22 €	5,22 €
744	Servidor VPS Ubuntu 1CPU, 1G, 25GB, 1TB	h	DigitalOceans	-	0,007 €	5,21 €
Total						654,08 €

Tabla 4 - Gasto material punto de acceso y servidor

El coste del material necesario para el punto de acceso y servicios de red asciende a **654,08 €**.

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

3. Mano de obra

La mano de obra representa el mayor gasto en el desarrollo de un proyecto de este tipo, ya que los componentes y materiales son económicos en comparación con las horas que implica el desarrollo de un producto electrónico nuevo.

Los costes de mano de obra de diseño tanto electrónico como de firmware y mecánico se han calculado estimando el precio facturable por hora de un ingeniero recién graduado en una ingeniería de la rama industrial a fecha de junio de 2018. Esto depende de varios factores, pero se ha decidido estimar en **40€/h**. La duración estimada del desarrollo del proyecto se ha estimado en 300 horas debido a que el trabajo representa una carga de 12 ECTS, correspondiendo cada ECTS a 25 horas de dedicación. No obstante, debido a que se han realizado tareas de desarrollo que se salen un poco del marco de trabajo, finalmente se han estimado unas 330 horas de trabajo, que se desglosan en la *Tabla 5*.

Para la estimación de las horas de diseño y desarrollo se ha realizado un desglose estimativo de cada tarea que se ha realizado, acorde con el plan de trabajo o cronograma seguido.

A pesar de que todas las tareas las ha realizado la misma persona, la labor de ensamblaje del prototipo (soldadura, limpieza y testing) representa un trabajo de técnico de menor cualificación y por tanto su precio se ha estimado de manera diferente.

Un	Description	Parcial	Cantidad	Precio	Total
h	Diseño electrónico		100	40,00 €	4.000,00 €
	1 - Diseño del esquema electrónico en Cadence	50			
	2 - Layout del circuito y diseño PCB	48			
	3 - Preparación documentación de fabricación	2			
h	Ensamblaje del prototipo		10	30,00 €	300,00 €
	1 - Montaje de la PCB. Soldadura manual SMD	9			
	2 - Inspección visual	0,25			
	3 - Limpieza de restos de flux	0,25			
	4 - Test eléctrico de continuidad y polaridad manual	0,5			
h	Desarrollo de la programación y depuración		145	40,00 €	5.800,00 €
	1 - Desarrollo librerías de sensores y depuración	25			
	2 - Desarrollo código fuente de test	15			
	3 - Desarrollo código fuente aplicación final	80			
	4 - Depuración aplicación final y programación prototipo	25			
h	Diseño mecánico de la carcasa		15	50,00 €	600,00 €
	1 - Diseño mecánico de la carcasa principal	5			
	2 - Diseño mecánico de la tapa	5			
	3 - Ensamblajes y simulaciones CAD	5			
h	Redacción y generación de documentación		60	50,00 €	4.400,00 €
	1 - Redacción memoria de proyecto	50			
	2 - Generación de planos	5			
	3 - Generación de presupuestos	5			
Total					13.100,00 €

Tabla 5 - Desglose presupuesto mano de obra

El precio final en concepto de mano de obra asciende a **13.100,00 €**.

Presupuesto

4. Amortización de los equipos

Para diseñar se precisa de la utilización de diversos equipos a menudo de alto costo y es necesario hacer una buena estimación de los mismos basada en su amortización con la vida útil de cada equipo, ya que representan una gran inversión que se ha de reflejar en el coste del proyecto.

Los precios de los equipos representan el valor del equipo en el momento de la realización de su compra. Para estimar su vida útil se ha realizado un sencillo calculo basado en sus horas de servicio diarias y su duración en años de esta manera:

$$Vida\ util\ (h) = Vida\ estimada(años) \times Utilización\ diaria\ promedio\ (h) \times 365$$

De esta manera obtenemos el precio que representa una hora de utilización del equipo. El valor del equipo al final de su vida estimada en años se considera cero o despreciable.

Para estimar las horas de utilización se ha tenido en cuenta las horas de mano de obra de la tabla anterior. Por ejemplo, se utiliza en todas las fases de diseño, por lo que se le asigna un gasto de 320 horas para el desarrollo del proyecto. De esta manera obtenemos la *Tabla 6*.

Unidad	Description	Precio Equipo	Vida Útil	Cantidad	Precio	Total
h	Ordenador personal Asus S551	800,00 €	14600	320	0,05 €	17,53 €
h	Multimetro Fluke 115	170,00 €	18250	5	0,01 €	0,05 €
h	Impresora 3D Prusa Mendel	500,00 €	2920	5,5	0,17 €	0,94 €
h	Estación soldadura Hakko FX888D	120,00 €	3650	9	0,03 €	0,30 €
h	Estación de calor Rework Hakko 850B	300,00 €	3650	9	0,08 €	0,74 €
h	Microscopio Digital Andonstar ADSM302	250,00 €	1825	10	0,14 €	1,37 €
Total						20,93 €

Tabla 6 - Desglose gasto amortización equipos

El precio final que representa la utilización de los equipos asciende a **20,93 €**

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

5. Resumen

Finalmente, se elabora una tabla resumen para el cálculo del gasto total del proyecto incluyendo las secciones anteriores.

Sección	Description	Parcial	Total
2	Materiales		734,14 €
2.1	Componentes electrónicos	61,81 €	
2.2	Fabricación circuito impreso	2,20 €	
2.3	Carcasa impresión 3D PLA	9,05 €	
2.4	Punto de acceso y servidor	654,08 €	
3	Mano de obra		13.100,00 €
4	Equipos		20,93 €
Total			13.855,07 €

IVA	21%	2.909,57 €
Gastos adicionales	10%	1.385,51 €
TOTAL COSTE PROYECTO		18.150,14 €

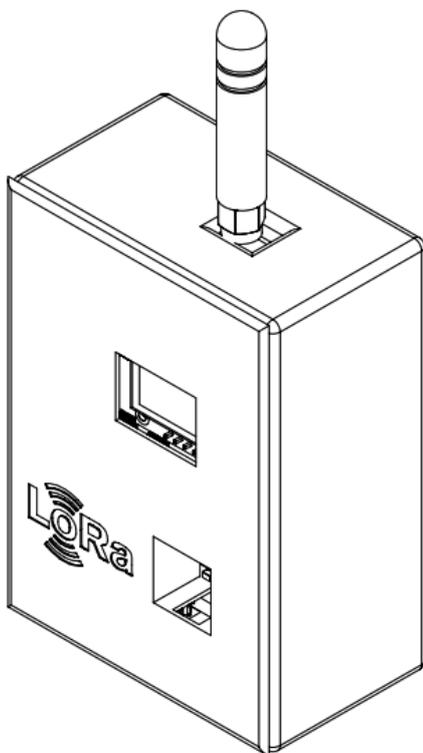
Tabla 7 - Resumen gastos totales impuestos y gastos adicionales incluidos

La elaboración del proyecto tiene un coste total asociado de DIECIOCHO MIL CIENTO CINCUENTA EUROS CON CATORCE CENTIMOS DE EURO (**18.150,14 €**)



DESARROLLO DE UN NODO INALÁMBRICO DE BAJO CONSUMO BASADO EN TÉCNICAS DE IOT PARA MONITORIZACIÓN DE PATRIMONIO CULTURAL

5. ANEXOS



Autor:

D. Jaime Laborda Macario

Tutor:

D. Ángel Perles Ivars

Valencia, junio de 2018

Anexo 1. Código

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

Contenido Anexo 1. Código

A1.1.	Librería sensor SHT31	4
1.1.	Declaración funciones – sht31.h.....	4
1.2.	Definición funciones – sht31.cpp	5
A1.2.	Librería sensor TSL2591	8
2.1.	Declaración funciones – tsl2591.h.....	8
2.2.	Definición funciones – tsl2591.cpp	11
A1.3.	Ejemplo de test medida de los sensores	17
A1.4.	Aplicación End Node Sensor inalámbrico.....	23
4.1.	Cuerpo principal – main.c.....	23
4.2.	Configuración LoRaWAN – Comissioning.h	32

Anexo 1. Código

A1.1. Librería sensor SHT31

1.1. Declaración funciones – sht31.h

```
1.  /*****
2.   This is a library for the SHT31 Digital Humidity & Temp Sensor
3.
4.   Designed specifically to work with the SHT31 Digital sensor from Adafruit
5.   ----> https://www.adafruit.com/products/2857
6.
7.   These displays use I2C to communicate, 2 pins are required to
8.   interface
9.   Adafruit invests time and resources providing this open source code,
10.  please support Adafruit and open-source hardware by purchasing
11.  products from Adafruit!
12.
13.  Written by Limor Fried/Ladyada for Adafruit Industries.
14.  Ported by Jaime Laborda Macario
15.  BSD license, all text above must be included in any redistribution
16.  *****/
17.
18. #include "stm32l0xx_hal.h"
19.
20. #define SHT31_DEFAULT_ADDR    0x44
21. #define SHT31_MEAS_HIGHREP_STRETCH 0x2C06
22. #define SHT31_MEAS_MEDREP_STRETCH 0x2C0D
23. #define SHT31_MEAS_LOWREP_STRETCH 0x2C10
24. #define SHT31_MEAS_HIGHREP    0x2400
25. #define SHT31_MEAS_MEDREP    0x240B
26. #define SHT31_MEAS_LOWREP    0x2416
27. #define SHT31_READSTATUS     0xF32D
28. #define SHT31_CLEARSTATUS    0x3041
29. #define SHT31_SOFTRESET      0x30A2
30. #define SHT31_HEATEREN       0x306D
31. #define SHT31_HEATERDIS     0x3066
32.
33.
34. class SHT31 {
35. public:
36.   bool begin(I2C_HandleTypeDef &i2cHandler, uint8_t i2caddr = SHT31_DEFAULT_ADDR);
37.   float readTemperature(void);
38.   float readHumidity(void);
39.   uint16_t readStatus(void);
40.   void reset(void);
41.   void heater(bool);
42.   uint8_t crc8(const uint8_t *data, int len);
43.
44. private:
45.   bool readTempHum(void); //Lectura conjunta de temperatura y humedad. No accesible al usuario
46.   void writeCommand(uint16_t cmd); //Función para enviar el comando al slave (son 2 bytes y el i2c solo acepta envíos de 1 byte)
47.
48.   uint8_t _i2caddr;
49.   //bool readData(void);
50.   I2C_HandleTypeDef _i2cHandler;
51.
52.   //boolean readData(void); There is no use for that function
53.   float rhum, temp;
54. };
```

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

1.2. Definición funciones – sht31.cpp

```
1.  /*****
2.   This is a library for the SHT31 Digital Humidity & Temp Sensor
3.
4.   Designed specifically to work with the SHT31 Digital sensor from Adafruit
5.   ----> https://www.adafruit.com/products/2857
6.
7.   These displays use I2C to communicate, 2 pins are required to
8.   interface
9.   Adafruit invests time and resources providing this open source code,
10.  please support Adafruit and open-source hardware by purchasing
11.  products from Adafruit!
12.
13.  Written by Limor Fried/Ladyada for Adafruit Industries.
14.  Ported by Jaime Laborda Macario
15.  BSD license, all text above must be included in any redistribution
16.  *****/
17.
18.  #include "sht31.h"
19.
20.  /*SHT31::SHT31() { //Nothing to do at the constructor
21.  }*/
22.
23.  bool SHT31::begin(I2C_HandleTypeDef &i2cHandler, uint8_t i2caddr) {
24.    _i2caddr = i2caddr;
25.    _i2cHandler = i2cHandler;
26.
27.    reset();
28.    //return (readStatus() == 0x40);
29.    return true;
30.  }
31.
32.  void SHT31::writeCommand(uint16_t cmd) {
33.    uint8_t command[2];
34.
35.    command[1] = cmd & 0xFF;
36.    command[0] = cmd >> 8;
37.    HAL_I2C_Master_Transmit(&_i2cHandler, _i2caddr<<1, command, 2, 100);
38.  }
39.
40.  uint16_t SHT31::readStatus(void) {
41.    uint8_t buffer[3];
42.    uint16_t stat;
43.
44.    writeCommand(SHT31_READSTATUS);
45.    HAL_I2C_Master_Receive(&_i2cHandler, _i2caddr<<1, buffer, 3, 100); //Receive from
46.    stat = buffer[0] << 8;
47.    //stat <<= 8;
48.    stat |= buffer[1];
49.    return stat;
50.  }
51.
52.  void SHT31::heater(bool h) {
53.    if (h)
54.      writeCommand(SHT31_HEATEREN);
55.    else
56.      writeCommand(SHT31_HEATERDIS);
57.  }
58.
```

Anexo 1. Código

```
59. void SHT31::reset(void) {
60.     writeCommand(SHT31_SOFTRESET);
61.     HAL_Delay(10);
62. }
63.
64. float SHT31::readTemperature(void) {
65.     if (! readTempHum()) return -1;
66.
67.     return temp;
68. }
69.
70.
71. float SHT31::readHumidity(void) {
72.     if (! readTempHum()) return -1;
73.
74.     return rhum;
75. }
76.
77. bool SHT31::readTempHum(void) {
78.     uint8_t readbuffer[6];
79.
80.     writeCommand(SHT31_MEAS_HIGHREP);
81.
82.     HAL_Delay(500);
83.
84.     HAL_I2C_Master_Receive(&_i2cHandler, _i2caddr<<1, readbuffer, 6, 100); //Receive from I2
C slave 6 bytes and store on buffer
85.
86.     uint16_t ST, SRH;
87.     ST = readbuffer[0];
88.     ST <<=8;
89.     ST |= readbuffer[1];
90.
91.     //checksum
92.     if (readbuffer[2] != crc8(readbuffer, 2)) return false;
93.
94.     SRH = readbuffer[3];
95.     SRH <<= 8;
96.     SRH |= readbuffer[4];
97.
98.     //checksum
99.     if (readbuffer[5] != crc8(readbuffer+3, 2)) return false;
100.
101.     //Temperature conversion to Degrees
102.     temp = ST;
103.     temp *= 175;
104.     temp /= 0xffff;
105.     temp = -45 + temp;
106.
107.     //Relative humidity conversion to %
108.     rhum = SRH;
109.     rhum *= 100;
110.     rhum /= 0xffff;
111.
112.     return true;
113. }
114.
115. uint8_t SHT31::crc8(const uint8_t *data, int len)
116. {
117.     /*
118.     *
119.     * CRC-8 formula from page 14 of SHT spec pdf
120.     *
```

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para
monitorización de patrimonio cultural

```
121.     * Test data 0xBE, 0xEF should yield 0x92
122.     *
123.     * Initialization data 0xFF
124.     * Polynomial 0x31 (x8 + x5 +x4 +1)
125.     * Final XOR 0x00
126.     */
127.
128.     const uint8_t POLYNOMIAL(0x31);
129.     uint8_t crc(0xFF);
130.
131.     for ( int j = len; j; --j ) {
132.         crc ^= *data++;
133.
134.         for ( int i = 8; i; --i ) {
135.             crc = ( crc & 0x80 )
136.                 ? (crc << 1) ^ POLYNOMIAL
137.                 : (crc << 1);
138.         }
139.     }
140.     return crc;
141. }
```

Anexo 1. Código

A1.2. Librería sensor TSL2591

2.1. Declaración funciones – tsl2591.h

```
1.  /*****  
2.  /*!  
3.  @file      Adafruit_TSL2591.h  
4.  @author    KT0WN (adafruit.com)  
5.  @author    Jaime Laborda (jailama@etsid.upv.es)  
6.  
7.  This is a library for the Adafruit TSL2591 breakout board  
8.  This library works with the Adafruit TSL2591 breakout  
9.  ----> https://www.adafruit.com/products/1980  
10.  
11. Check out the links above for our tutorials and wiring diagrams  
12. These chips use I2C to communicate  
13.  
14. Adafruit invests time and resources providing this open source code,  
15. please support Adafruit and open-source hardware by purchasing  
16. products from Adafruit!  
17.  
18. THI LIBRARY HAS BEEN PORTED TO STM32 BY JAIME LABORDA. ALSO, SOME MORE USEFUL FUNCTIONALLI  
    TY HAS BEEN ADDED.  
19.  
20. @section LICENSE  
21.  
22. Software License Agreement (BSD License)  
23.  
24. Copyright (c) 2014 Adafruit Industries  
25. All rights reserved.  
26.  
27. Redistribution and use in source and binary forms, with or without  
28. modification, are permitted provided that the following conditions are met:  
29. 1. Redistributions of source code must retain the above copyright  
30. notice, this list of conditions and the following disclaimer.  
31. 2. Redistributions in binary form must reproduce the above copyright  
32. notice, this list of conditions and the following disclaimer in the  
33. documentation and/or other materials provided with the distribution.  
34. 3. Neither the name of the copyright holders nor the  
35. names of its contributors may be used to endorse or promote products  
36. derived from this software without specific prior written permission.  
37.  
38. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS 'AS IS' AND ANY  
39. EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED  
40. WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE  
41. DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE FOR ANY  
42. DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES  
43. (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;  
44. LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND  
45. ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT  
46. (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS  
47. SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.  
48. */  
49. /*****  
50.  
51. #include "stm32l0xx_hal.h"  
52.  
53. #define TSL2591_VISIBLE          (2)      // channel 0 - channel 1  
54. #define TSL2591_INFRARED        (1)      // channel 1  
55. #define TSL2591_FULLSPECTRUM    (0)      // channel 0  
56.
```

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

```
57. #define TSL2591_DEFAULT_ADDR      (0x29)
58. #define TSL2591_READBIT           (0x01)
59.
60. #define TSL2591_COMMAND_BIT        (0xA0)    // bits 7 and 5 for 'command normal'
61. #define TSL2591_CLEAR_BIT          (0x40)    // Clears any pending interrupt (write 1 to clear)
62. #define TSL2591_WORD_BIT           (0x20)    // 1 = read/write word (rather than byte)
63. #define TSL2591_BLOCK_BIT          (0x10)    // 1 = using block read/write
64.
65. #define TSL2591_ENABLE_POWERON     (0x01)
66. #define TSL2591_ENABLE_POWEROFF    (0x00)
67. #define TSL2591_ENABLE_AEN         (0x02)
68. #define TSL2591_ENABLE_AIEN        (0x10)
69.
70. #define TSL2591_CONTROL_RESET       (0x80)
71.
72. #define TSL2591_LUX_DF               (408.0F)
73. #define TSL2591_LUX_COEFB           (1.64F)  // CH0 coefficient
74. #define TSL2591_LUX_COEFC           (0.59F)  // CH1 coefficient A
75. #define TSL2591_LUX_COEFD           (0.86F)  // CH2 coefficient B
76.
77. enum
78. {
79.     TSL2591_REGISTER_ENABLE = 0x00,
80.     TSL2591_REGISTER_CONTROL = 0x01,
81.     TSL2591_REGISTER_THRESHOLD_LOW = 0x02,
82.     TSL2591_REGISTER_THRESHOLD_HIGH = 0x03,
83.     TSL2591_REGISTER_THRESHOLD_LOW = 0x04,
84.     TSL2591_REGISTER_THRESHOLD_HIGH = 0x05,
85.     TSL2591_REGISTER_INTERRUPT = 0x06,
86.     TSL2591_REGISTER_CRC = 0x08,
87.     TSL2591_REGISTER_ID = 0x0A,
88.     TSL2591_REGISTER_CHAN0_LOW = 0x14,
89.     TSL2591_REGISTER_CHAN0_HIGH = 0x15,
90.     TSL2591_REGISTER_CHAN1_LOW = 0x16,
91.     TSL2591_REGISTER_CHAN1_HIGH = 0x17
92. };
93.
94. typedef enum
95. {
96.     TSL2591_INTEGRATIONTIME_100MS = 0x00,
97.     TSL2591_INTEGRATIONTIME_200MS = 0x01,
98.     TSL2591_INTEGRATIONTIME_300MS = 0x02,
99.     TSL2591_INTEGRATIONTIME_400MS = 0x03,
100.     TSL2591_INTEGRATIONTIME_500MS = 0x04,
101.     TSL2591_INTEGRATIONTIME_600MS = 0x05,
102. }
103.     tsl2591IntegrationTime_t;
104.
105.     typedef enum
106.     {
107.         TSL2591_GAIN_LOW = 0x00,    // low gain (1x)
108.         TSL2591_GAIN_MED = 0x10,    // medium gain (25x)
109.         TSL2591_GAIN_HIGH = 0x20,   // medium gain (428x)
110.         TSL2591_GAIN_MAX = 0x30,    // max gain (9876x)
111.     }
112.     tsl2591Gain_t;
113.
114.
115.     class TSL2591 {
116.     public:
```

Anexo 1. Código

```
117.         //Funciones de usuario
118.         bool      begin( I2C_HandleTypeDef &i2cHandler, uint8_t i2caddr = TSL2591_DEFAU
    LT_ADDR );
119.
120.         uint32_t  calculateLux(uint16_t ch0, uint16_t ch1);
121.         void      setGain(tsl2591Gain_t gain);
122.         void      setTiming(tsl2591IntegrationTime_t integration);
123.         uint16_t  getLuminosity(uint8_t channel);
124.         uint32_t  getFullLuminosity();
125.
126.         //added functions
127.         uint32_t  getLux ();
128.
129.         tsl2591IntegrationTime_t getTiming();
130.         tsl2591Gain_t      getGain();
131.
132.         uint8_t checkId ();
133.         uint8_t checkStatus ();
134.         uint8_t checkRegister ( uint8_t requested_register );
135.
136.
137.
138.     private:
139.         //Funciones no accesibles al usuario
140.         void      enable(void);
141.         void      disable(void);
142.         void      write8(uint8_t r, uint8_t v);
143.         uint16_t  read16(uint8_t reg);
144.         uint8_t   read8(uint8_t reg);
145.
146.         //Private variables
147.         uint8_t _i2caddr;
148.         I2C_HandleTypeDef _i2cHandler;
149.
150.         tsl2591IntegrationTime_t _integration;
151.         tsl2591Gain_t _gain;
152.         bool _initialized;
153.     };
```

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

2.2. Definición funciones – tsl2591.cpp

```
1.  /*****  
2.  /*!  
3.  @file    Adafruit_TSL2591.h  
4.  @author  KT0WN (adafruit.com)  
5.  @author  Jaime Laborda (jailama@etsid.upv.es)  
6.  
7.  This is a library for the Adafruit TSL2591 breakout board  
8.  This library works with the Adafruit TSL2591 breakout  
9.  ----> https://www.adafruit.com/products/1980  
10.  
11. Check out the links above for our tutorials and wiring diagrams  
12. These chips use I2C to communicate  
13.  
14. Adafruit invests time and resources providing this open source code,  
15. please support Adafruit and open-source hardware by purchasing  
16. products from Adafruit!  
17.  
18. THI LIBRARY HAS BEEN PORTED TO STM32 BY JAIME LABORDA. ALSO, SOME MORE USEFUL FUNCTIONALLI  
    TY HAS BEEN ADDED.  
19.  
20. @section LICENSE  
21.  
22. Software License Agreement (BSD License)  
23.  
24. Copyright (c) 2014 Adafruit Industries  
25. All rights reserved.  
26.  
27. Redistribution and use in source and binary forms, with or without  
28. modification, are permitted provided that the following conditions are met:  
29. 1. Redistributions of source code must retain the above copyright  
30. notice, this list of conditions and the following disclaimer.  
31. 2. Redistributions in binary form must reproduce the above copyright  
32. notice, this list of conditions and the following disclaimer in the  
33. documentation and/or other materials provided with the distribution.  
34. 3. Neither the name of the copyright holders nor the  
35. names of its contributors may be used to endorse or promote products  
36. derived from this software without specific prior written permission.  
37.  
38. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS 'AS IS' AND ANY  
39. EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED  
40. WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE  
41. DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE FOR ANY  
42. DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES  
43. (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;  
44. LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND  
45. ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT  
46. (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS  
47. SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.  
48. */  
49. /*****  
50.  
51. #include "tsl2591.h"  
52.  
53. /*TSL2591::TSL2591() { //Nothing to do at the constructor  
54. }*/  
55.  
56. bool TSL2591::begin( I2C_HandleTypeDef &i2cHandler, uint8_t i2caddr ) {  
57.     _i2caddr = i2caddr;
```

Anexo 1. Código

```
58.     _i2cHandler = i2cHandler;
59.
60.     //Set default configuration options
61.     _integration = TSL2591_INTEGRATIONTIME_100MS;
62.     _gain = TSL2591_GAIN_MED;
63.
64.     //Read ID to check if communicates and everything OK
65.     uint8_t id = read8(0x12);
66.     if (id != 0x50)
67.         return false;
68.
69.     _initialized = true;
70.     // Set default integration time and gain
71.     setTiming(_integration);
72.     setGain(_gain);
73.
74.     // Note: by default, the device is in power down mode on bootup
75.     disable();
76.
77.     return true; //If everthing goes well
78. }
79.
80. void TSL2591::enable(void)
81. {
82.     // Enable the device by setting the control bit to 0x01
83.     write8(TSL2591_COMMAND_BIT | TSL2591_REGISTER_ENABLE, TSL2591_ENABLE_POWERON | TSL2591
_ENABLE_AEN | TSL2591_ENABLE_AIEN);
84. }
85.
86. void TSL2591::disable(void)
87. {
88.     // Disable the device by setting the control bit to 0x00
89.     write8(TSL2591_COMMAND_BIT | TSL2591_REGISTER_ENABLE, TSL2591_ENABLE_POWEROFF);
90. }
91.
92. void TSL2591::setGain(tsl2591Gain_t gain)
93. {
94.     enable();
95.     _gain = gain;
96.     write8(TSL2591_COMMAND_BIT | TSL2591_REGISTER_CONTROL, _integration | _gain);
97.     disable();
98. }
99.
100.    tsl2591Gain_t TSL2591::getGain()
101.    {
102.        return _gain;
103.    }
104.
105.    void TSL2591::setTiming(tsl2591IntegrationTime_t integration)
106.    {
107.        enable();
108.        _integration = integration;
109.        write8(TSL2591_COMMAND_BIT | TSL2591_REGISTER_CONTROL, _integration | _gain);
110.        disable();
111.    }
112.
113.    tsl2591IntegrationTime_t TSL2591::getTiming()
114.    {
115.        return _integration;
116.    }
117.
118.    uint32_t TSL2591::calculateLux(uint16_t ch0, uint16_t ch1)
119.    {
```

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para
monitorización de patrimonio cultural

```
120.     uint16_t atime, again;
121.     float    cpl, lux1, lux2, lux;
122.     uint32_t chan0, chan1;
123.
124.     // Check for overflow conditions first
125.     if ((ch0 == 0xFFFF) | (ch1 == 0xFFFF))
126.     {
127.         // Signal an overflow
128.         return 0;
129.     }
130.
131.     // Note: This algorithm is based on preliminary coefficients
132.     // provided by AMS and may need to be updated in the future
133.
134.     switch (_integration)
135.     {
136.     case TSL2591_INTEGRATIONTIME_100MS:
137.         atime = 100.0F;
138.         break;
139.     case TSL2591_INTEGRATIONTIME_200MS:
140.         atime = 200.0F;
141.         break;
142.     case TSL2591_INTEGRATIONTIME_300MS:
143.         atime = 300.0F;
144.         break;
145.     case TSL2591_INTEGRATIONTIME_400MS:
146.         atime = 400.0F;
147.         break;
148.     case TSL2591_INTEGRATIONTIME_500MS:
149.         atime = 500.0F;
150.         break;
151.     case TSL2591_INTEGRATIONTIME_600MS:
152.         atime = 600.0F;
153.         break;
154.     default: // 100ms
155.         atime = 100.0F;
156.         break;
157.     }
158.
159.     switch (_gain)
160.     {
161.     case TSL2591_GAIN_LOW:
162.         again = 1.0F;
163.         break;
164.     case TSL2591_GAIN_MED:
165.         again = 25.0F;
166.         break;
167.     case TSL2591_GAIN_HIGH:
168.         again = 428.0F;
169.         break;
170.     case TSL2591_GAIN_MAX:
171.         again = 9876.0F;
172.         break;
173.     default:
174.         again = 1.0F;
175.         break;
176.     }
177.
178.     // cpl = (ATIME * AGAIN) / DF
179.     cpl = (atime * again) / TSL2591_LUX_DF;
180.
```

Anexo 1. Código

```
181.         lux1 = ((float)ch0 - (TSL2591_LUX_COEFB * (float)ch1)) / cpl;
182.         lux2 = ((TSL2591_LUX_COEFC * (float)ch0) - (TSL2591_LUX_COEFD * (float)ch1)) /
    cpl;
183.
184.         // The highest value is the approximate lux equivalent
185.         lux = lux1 > lux2 ? lux1 : lux2;
186.
187.         // Signal I2C had no errors
188.         return (uint32_t)lux;
189.     }
190.
191.     uint32_t TSL2591::getFullLuminosity(void)
192.     {
193.         // Enable the device
194.         enable();
195.
196.         // Wait x ms for ADC to complete
197.         for (uint8_t d = 0; d <= _integration; d++)
198.         {
199.             HAL_Delay(120);
200.         }
201.
202.         uint32_t x;
203.         x = read16(TSL2591_COMMAND_BIT | TSL2591_REGISTER_CHAN1_LOW);
204.         x <<= 16;
205.         x |= read16(TSL2591_COMMAND_BIT | TSL2591_REGISTER_CHAN0_LOW);
206.
207.         // Commented because it was causing errors. Don't know why??
208.         //disable();
209.
210.         return x;
211.     }
212.
213.     uint16_t TSL2591::getLuminosity(uint8_t channel)
214.     {
215.         uint32_t x = getFullLuminosity();
216.
217.         if (channel == TSL2591_FULLSPECTRUM)
218.         {
219.             // Reads two byte value from channel 0 (visible + infrared)
220.             return (x & 0xFFFF);
221.         }
222.         else if (channel == TSL2591_INFRARED)
223.         {
224.             // Reads two byte value from channel 1 (infrared)
225.             return (x >> 16);
226.         }
227.         else if (channel == TSL2591_VISIBLE)
228.         {
229.             // Reads all and subtracts out just the visible!
230.             return ((x & 0xFFFF) - (x >> 16));
231.         }
232.
233.         // unknown channel!
234.         return 0;
235.     }
236.
237.     uint8_t TSL2591::read8(uint8_t reg)
238.     {
239.         uint8_t buffer;
240.
241.         buffer = 0x80 | 0x20 | reg;
242.
```

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para
monitorización de patrimonio cultural

```
243.         HAL_I2C_Master_Transmit(&_i2cHandler, _i2caddr << 1, &buffer, 1, 100); // Set r
egister wanted to read from
244.
245.         HAL_I2C_Master_Receive(&_i2cHandler, _i2caddr << 1, &buffer, 1, 100); //Receive
from I2C slave 1 byte and store on buffer
246.
247.         return buffer;
248.     }
249.
250.     uint16_t TSL2591::read16(uint8_t reg)
251.     {
252.         uint8_t buffer[2]; //Two bytes buffer to store received data
253.         uint16_t output;
254.
255.
256.         HAL_I2C_Master_Transmit(&_i2cHandler, _i2caddr << 1, ®, 1, 100); // Set registe
r wanted to read from
257.
258.         HAL_I2C_Master_Receive(&_i2cHandler, _i2caddr << 1, buffer, 2, 100); //Receive
from I2C slave 2 bytes and store on buffer
259.
260.         output = buffer[1];
261.         output <<= 8;
262.         output |= buffer[0];
263.
264.         return output;
265.     }
266.
267.     void TSL2591::write8(uint8_t reg, uint8_t value)
268.     {
269.         uint8_t command[2];
270.
271.         //Construct the command
272.         command[1] = reg;
273.         command[0] = value;
274.
275.         //Send command by i2c as a two bytes step
276.         HAL_I2C_Master_Transmit(&_i2cHandler, _i2caddr << 1, command, 2, 100);
277.     }
278.
279.     uint8_t TSL2591::checkId( )
280.     {
281.         //Read ID to check if communicates and everything OK. Needs to return 0x50 if O
K
282.         return read8(0x12);
283.     }
284.
285.     uint8_t TSL2591::checkStatus( )
286.     {
287.         //Read ID to check if communicates and everything OK. Needs to return 0x50 if O
K
288.         return read8(0x13);
289.     }
290.
291.     uint8_t TSL2591::checkRegister( uint8_t requested_register)
292.     {
293.         //Read a register selected by the user
294.         return read8( requested_register );
295.     }
296.
297.     uint32_t TSL2591::getLux()
```

Anexo 1. Código

```
298.     {  
299.         uint32_t lum = getFullLuminosity();  
300.         uint16_t ir, full;  
301.  
302.         ir = lum >> 16;  
303.         full = lum & 0xFFFF;  
304.  
305.         return calculateLux(full, ir);  
306.     }  
307.  
308.     /*****  
309.     *****/
```

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para
monitorización de patrimonio cultural

A1.3. Ejemplo de test medida de los sensores

```
1. /**
2.  *****
3.  * File Name      : main.c
4.  * Description    : Main program body Sensor test
5.  *****
6.  ** This notice applies to any and all portions of this file
7.  * that are not between comment pairs USER CODE BEGIN and
8.  * USER CODE END. Other portions of this file, whether
9.  * inserted by the user or by software development tools
10. * are owned by their respective copyright owners.
11. *
12. * COPYRIGHT(c) 2018 STMicroelectronics
13. *
14. * Redistribution and use in source and binary forms, with or without modification,
15. * are permitted provided that the following conditions are met:
16. *   1. Redistributions of source code must retain the above copyright notice,
17. *     this list of conditions and the following disclaimer.
18. *   2. Redistributions in binary form must reproduce the above copyright notice,
19. *     this list of conditions and the following disclaimer in the documentation
20. *     and/or other materials provided with the distribution.
21. *   3. Neither the name of STMicroelectronics nor the names of its contributors
22. *     may be used to endorse or promote products derived from this software
23. *     without specific prior written permission.
24. *
25. * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
26. * AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
27. * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
28. * DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE
29. * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
30. * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
31. * SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
32. * CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
33. * OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
34. * OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
35. *
36.  *****
37. */
38. /* Includes -----*/
39. #include "main.h"
40. #include "stm32l0xx_hal.h"
41. #include "sht31.h"
42. #include "tsl2591.h"
43.
44. /* USER CODE BEGIN Includes */
45. //#define ADDRESS 0X44
46.
47. /* USER CODE END Includes */
48.
49. /* Private variables -----*/
50. I2C_HandleTypeDef hi2c1;
51.
52. /* USER CODE BEGIN PV */
53. /* Private variables -----*/
54. //float temp, rhum;
55.
56. /* USER CODE END PV */
57.
58. /* Private function prototypes -----*/
```

Anexo 1. Código

```
59. void SystemClock_Config(void);
60. static void MX_GPIO_Init(void);
61. static void MX_I2C1_Init(void);
62.
63. /* USER CODE BEGIN PFP */
64. /* Private function prototypes -----*/
65.
66. /* USER CODE END PFP */
67.
68. /* USER CODE BEGIN 0 */
69. SHT31 sht31;
70. TSL2591 tsl2591;
71.
72. float temperature, humidity;
73. uint16_t lum;
74. uint32_t full_lum;
75. uint32_t lux;
76. int _lux;
77. uint8_t id, status;
78.
79. /* USER CODE END 0 */
80.
81. int main(void)
82. {
83.
84. /* USER CODE BEGIN 1 */
85.
86. /* USER CODE END 1 */
87.
88. /* MCU Configuration-----*/
89.
90. /* Reset of all peripherals, Initializes the Flash interface and the Systick. */
91. HAL_Init();
92.
93. /* USER CODE BEGIN Init */
94.
95. /* USER CODE END Init */
96.
97. /* Configure the system clock */
98. SystemClock_Config();
99.
100. /* USER CODE BEGIN SysInit */
101.
102. /* USER CODE END SysInit */
103.
104. /* Initialize all configured peripherals */
105. MX_GPIO_Init();
106. MX_I2C1_Init();
107.
108. /* USER CODE BEGIN 2 */
109.
110. sht31.begin(hi2c1);
111. tsl2591.begin(hi2c1);
112.
113. /* USER CODE END 2 */
114.
115. /* Infinite loop */
116. /* USER CODE BEGIN WHILE */
117. while (1)
118. {
119. /* USER CODE END WHILE */
120.
121. /* USER CODE BEGIN 3 */
```

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para
monitorización de patrimonio cultural

```
122.         HAL_GPIO_WritePin(GPIOB, GPIO_PIN_5, GPIO_PIN_SET);
123.         HAL_Delay(10);
124.         HAL_GPIO_WritePin(GPIOB, GPIO_PIN_5, GPIO_PIN_RESET);
125.         HAL_Delay(10);
126.
127.         temperature = sht31.readTemperature();
128.         humidity = sht31.readHumidity();
129.
130.         id = tsl2591.checkId ();
131.         status = tsl2591.checkStatus ();
132.
133.         full_lum = tsl2591.getFullLuminosity();
134.         lum = tsl2591.getLuminosity(TSL2591_VISIBLE);
135.         lux = tsl2591.getLux();
136.
137.         _lux = lux;
138.
139.         HAL_Delay(100);
140.
141.     }
142.     /* USER CODE END 3 */
143.
144. }
145.
146. /** System Clock Configuration
147. */
148. void SystemClock_Config(void)
149. {
150.
151.     RCC_OscInitTypeDef RCC_OscInitStruct;
152.     RCC_ClkInitTypeDef RCC_ClkInitStruct;
153.     RCC_PeriphCLKInitTypeDef PeriphClkInit;
154.
155.     /**Configure the main internal regulator output voltage
156.     */
157.     __HAL_PWR_VOLTAGESCALING_CONFIG(PWR_REGULATOR_VOLTAGE_SCALE1);
158.
159.     /**Initializes the CPU, AHB and APB busses clocks
160.     */
161.     RCC_OscInitStruct.OscillatorType = RCC_OSCILLATORTYPE_MSI;
162.     RCC_OscInitStruct.MSIState = RCC_MSI_ON;
163.     RCC_OscInitStruct.MSICalibrationValue = 0;
164.     RCC_OscInitStruct.MSIClockRange = RCC_MSIRANGE_5;
165.     RCC_OscInitStruct.PLL.PLLState = RCC_PLL_NONE;
166.     if (HAL_RCC_OscConfig(&RCC_OscInitStruct) != HAL_OK)
167.     {
168.         _Error_Handler(__FILE__, __LINE__);
169.     }
170.
171.     /**Initializes the CPU, AHB and APB busses clocks
172.     */
173.     RCC_ClkInitStruct.ClockType = RCC_CLOCKTYPE_HCLK|RCC_CLOCKTYPE_SYSCLK
174.                                   |RCC_CLOCKTYPE_PCLK1|RCC_CLOCKTYPE_PCLK2;
175.     RCC_ClkInitStruct.SYSCLKSource = RCC_SYSCLKSOURCE_MSI;
176.     RCC_ClkInitStruct.AHBCLKDivider = RCC_SYSCLK_DIV1;
177.     RCC_ClkInitStruct.APB1CLKDivider = RCC_HCLK_DIV1;
178.     RCC_ClkInitStruct.APB2CLKDivider = RCC_HCLK_DIV1;
179.
180.     if (HAL_RCC_ClockConfig(&RCC_ClkInitStruct, FLASH_LATENCY_0) != HAL_OK)
181.     {
182.         _Error_Handler(__FILE__, __LINE__);
```

Anexo 1. Código

```
183.     }
184.
185.     PeriphClkInit.PeriphClockSelection = RCC_PERIPHCLK_I2C1;
186.     PeriphClkInit.I2c1ClockSelection = RCC_I2C1CLKSOURCE_PCLK1;
187.     if (HAL_RCCEx_PeriphCLKConfig(&PeriphClkInit) != HAL_OK)
188.     {
189.         _Error_Handler(__FILE__, __LINE__);
190.     }
191.
192.     /**Configure the SysTick interrupt time
193.     */
194.     HAL_SYSTICK_Config(HAL_RCC_GetHCLKFreq()/1000);
195.
196.     /**Configure the SysTick
197.     */
198.     HAL_SYSTICK_CLKSourceConfig(SYSTICK_CLKSOURCE_HCLK);
199.
200.     /* SysTick_IRQn interrupt configuration */
201.     HAL_NVIC_SetPriority(SysTick_IRQn, 0, 0);
202. }
203.
204. /* I2C1 init function */
205. static void MX_I2C1_Init(void)
206. {
207.
208.     hi2c1.Instance = I2C1;
209.     hi2c1.Init.Timing = 0x00000708;
210.     hi2c1.Init.OwnAddress1 = 0;
211.     hi2c1.Init.AddressingMode = I2C_ADDRESSINGMODE_7BIT;
212.     hi2c1.Init.DualAddressMode = I2C_DUALADDRESS_DISABLE;
213.     hi2c1.Init.OwnAddress2 = 0;
214.     hi2c1.Init.OwnAddress2Masks = I2C_OA2_NOMASK;
215.     hi2c1.Init.GeneralCallMode = I2C_GENERALCALL_DISABLE;
216.     hi2c1.Init.NoStretchMode = I2C_NOSTRETCH_DISABLE;
217.     if (HAL_I2C_Init(&hi2c1) != HAL_OK)
218.     {
219.         _Error_Handler(__FILE__, __LINE__);
220.     }
221.
222.     /**Configure Analogue filter
223.     */
224.     if (HAL_I2CEx_ConfigAnalogFilter(&hi2c1, I2C_ANALOGFILTER_ENABLE) != HAL_OK)
225.     {
226.         _Error_Handler(__FILE__, __LINE__);
227.     }
228.
229.     /**Configure Digital filter
230.     */
231.     if (HAL_I2CEx_ConfigDigitalFilter(&hi2c1, 0) != HAL_OK)
232.     {
233.         _Error_Handler(__FILE__, __LINE__);
234.     }
235.
236. }
237.
238. /** Configure pins as
239.     * Analog
240.     * Input
241.     * Output
242.     * EVENT_OUT
243.     * EXTI
244. */
245. static void MX_GPIO_Init(void)
```

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para
monitorización de patrimonio cultural

```
246.     {
247.
248.         GPIO_InitTypeDef GPIO_InitStructure;
249.
250.         /* GPIO Ports Clock Enable */
251.         __HAL_RCC_GPIOB_CLK_ENABLE();
252.         __HAL_RCC_GPIOA_CLK_ENABLE();
253.
254.         /*Configure GPIO pin Output Level */
255.         HAL_GPIO_WritePin(GPIOB, GPIO_PIN_5|GPIO_PIN_6|GPIO_PIN_7, GPIO_PIN_RESET);
256.
257.         /*Configure GPIO pin Output Level */
258.         HAL_GPIO_WritePin(GPIOA, GPIO_PIN_5, GPIO_PIN_RESET);
259.
260.         /*Configure GPIO pins : PB5 PB6 PB7 */
261.         GPIO_InitStructure.Pin = GPIO_PIN_5|GPIO_PIN_6|GPIO_PIN_7;
262.         GPIO_InitStructure.Mode = GPIO_MODE_OUTPUT_PP;
263.         GPIO_InitStructure.Pull = GPIO_NOPULL;
264.         GPIO_InitStructure.Speed = GPIO_SPEED_FREQ_LOW;
265.         HAL_GPIO_Init(GPIOB, &GPIO_InitStructure);
266.
267.         /*Configure GPIO pin : PA5 */
268.         GPIO_InitStructure.Pin = GPIO_PIN_5;
269.         GPIO_InitStructure.Mode = GPIO_MODE_OUTPUT_PP;
270.         GPIO_InitStructure.Pull = GPIO_NOPULL;
271.         GPIO_InitStructure.Speed = GPIO_SPEED_FREQ_LOW;
272.         HAL_GPIO_Init(GPIOA, &GPIO_InitStructure);
273.
274.     }
275.
276.     /* USER CODE BEGIN 4 */
277.
278.     /* USER CODE END 4 */
279.
280.     /**
281.      * @brief This function is executed in case of error occurrence.
282.      * @param None
283.      * @retval None
284.      */
285.     void _Error_Handler(char * file, int line)
286.     {
287.         /* USER CODE BEGIN Error_Handler_Debug */
288.         /* User can add his own implementation to report the HAL error return state */
289.         while(1)
290.         {
291.         }
292.         /* USER CODE END Error_Handler_Debug */
293.     }
294.
295.     #ifdef USE_FULL_ASSERT
296.
297.     /**
298.      * @brief Reports the name of the source file and the source line number
299.      * where the assert_param error has occurred.
300.      * @param file: pointer to the source file name
301.      * @param line: assert_param error line source number
302.      * @retval None
303.      */
304.     void assert_failed(uint8_t* file, uint32_t line)
305.     {
306.         /* USER CODE BEGIN 6 */
```

Anexo 1. Código

```
307.      /* User can add his own implementation to report the file name and line number,  
308.         ex: printf("Wrong parameters value: file %s on line %d\r\n", file, line) */  
309.      /* USER CODE END 6 */  
310.  
311.    }  
312.  
313.    #endif  
314.  
315.    /**  
316.     * @}  
317.     */  
318.  
319.    /**  
320.     * @}  
321.     */  
322.  
323.    /***** (C) COPYRIGHT STMicroelectronics *****/
```

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

A1.4. Aplicación End Node Sensor inalámbrico

4.1. Cuerpo principal – main.c

```
1.  /******  
2.  * @file    main.c  
3.  * @author  J. Laborda  
4.  * @version V1.0  
5.  * @date    Marzo 2018  
6.  * @brief   End Node Application  
7.  *****/  
8.  */  
9.  
10. #define CAYENNE_LPP  
11. //#define SIMULATE_SENSORS  
12. //#define DEBUG  
13. #define LOW_POWER_DISABLE  
14.  
15. /* Includes -----*/  
16. #include "hw.h"  
17. #include "stm32l0xx_hal.h"  
18. #include "low_power_manager.h"  
19. #include "lora.h"  
20. //#include "bsp.h"  
21. #include "timeServer.h"  
22. #include "vcom.h"  
23. #include "version.h"  
24. #include <stdbool.h>  
25.  
26. /* Sensors library */  
27. #include "sht31.h"  
28. #include "ts2591.h"  
29.  
30. /* Private typedef -----*/  
31. /* Private define -----*/  
32.  
33. /*!  
34. * CAYENNE_LPP is myDevices Application server.  
35. * Usamos CAYENNE_LPP para la codificación hexadecimal de la información  
36. * de los sensores  
37. */  
38. #define CAYENNE_LPP  
39. #define LPP_DATATYPE_DIGITAL_INPUT    0x0  
40. #define LPP_DATATYPE_HUMIDITY         0x68  
41. #define LPP_DATATYPE_TEMPERATURE      0x67  
42. #define LPP_DATATYPE_ILLUMINANCE     0x65  
43.  
44. #define LPP_APP_PORT 99  
45. /*!  
46. * Defines the application data transmission duty cycle. 5s, value in [ms].  
47. */  
48. #define APP_TX_DUTYCYCLE                900000 //15 minutes = 900.000 ms  
49. /*!  
50. * LoRaWAN Adaptive Data Rate  
51. * @note Please note that when ADR is enabled the end-device should be static  
52. */  
53. #define LORAWAN_ADR_STATE LORAWAN_ADR_ON  
54. /*!  
55. * LoRaWAN Default data Rate Data Rate
```

Anexo 1. Código

```
56. * @note Please note that LORAWAN_DEFAULT_DATA_RATE is used only when ADR is disabled
57. * DR_0 = 12SF125BW
58. * DR_3 = 9SF125BW
59. * DR_5 = 7SF125BW
60. */
61. #define LORAWAN_DEFAULT_DATA_RATE DR_5 //Spreading Factor 7, Bandwidth 125kHz
62. /*!
63. * LoRaWAN application port
64. * @note do not use 224. It is reserved for certification
65. */
66. #define LORAWAN_APP_PORT 2
67. /*!
68. * LoRaWAN default endNode class port
69. */
70. #define LORAWAN_DEFAULT_CLASS CLASS_A //Clase del dispositivo
71. /*!
72. * LoRaWAN default confirm state
73. */
74. #define LORAWAN_DEFAULT_CONFIRM_MSG_STATE LORAWAN_UNCONFIRMED_MSG
75. /*!
76. * User application data buffer size
77. */
78. #define LORAWAN_APP_DATA_BUFF_SIZE 64
79. /*!
80. * User application data
81. */
82. static uint8_t AppDataBuff[LORAWAN_APP_DATA_BUFF_SIZE];
83.
84. /*!
85. * User application data structure
86. */
87. static lora_AppData_t AppData={ AppDataBuff, 0 ,0 };
88. /* Private macro -----*/
89. /* Private function prototypes -----*/
90.
91. /* call back when LoRa endNode has received a frame*/
92. static void LORA_RxData( lora_AppData_t *AppData);
93.
94. /* call back when LoRa endNode has just joined*/
95. static void LORA_HasJoined( void );
96.
97. /* call back when LoRa endNode has just switch the class*/
98. static void LORA_ConfirmClass ( DeviceClass_t Class );
99.
100. /* call back when server needs endNode to send a frame*/
101. static void LORA_TxNeeded ( void );
102.
103. /* LoRa endNode send request*/
104. static void Send( void );
105.
106. /* start the tx process*/
107. static void LoraStartTx(TxEventType_t EventType);
108.
109. /* tx timer callback function*/
110. static void OnTxTimerEvent( void );
111.
112. /* i2c init function */
113. static void MX_I2C1_Init(void);
114. static void MX_GPIO_Init(void);
115.
116. /* Private variables -----*/
117. /* load Main call backs structure*/
118. static LoRaMainCallback_t LoRaMainCallbacks = { HW_GetBatteryLevel,
```

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para
monitorización de patrimonio cultural

```
119.                                     HW_GetTemperatureLevel,
120.                                     HW_GetUniqueId,
121.                                     HW_GetRandomSeed,
122.                                     LORA_RxData,
123.                                     LORA_HasJoined,
124.                                     LORA_ConfirmClass,
125.                                     LORA_TxNeeded};
126.
127.     /*!
128.      * Specifies the state of the application LED
129.      */
130.     static uint8_t AppLedStateOn = RESET;
131.
132.     static TimerEvent_t TxTimer;
133.
134.     #ifdef USE_B_L072Z_LRWAN1
135.     /*!
136.      * Timer to handle the application Tx Led to toggle
137.      */
138.     static TimerEvent_t TxLedTimer;
139.
140.     static void OnTimerLedEvent( void );
141.     #endif
142.     /* !
143.      *Initialises the Lora Parameters
144.      */
145.     static LoRaParam_t LoRaParamInit= {LORAWAN_ADR_STATE,
146.                                         LORAWAN_DEFAULT_DATA_RATE,
147.                                         LORAWAN_PUBLIC_NETWORK};
148.
149.     /* Sensors objects */
150.     I2C_HandleTypeDef hi2c1; //i2c object
151.     SHT31 sht31;
152.     TSL2591 tsl2591;
153.
154.     /* Private functions -----*/
155.
156.     /**
157.      * @brief Main program
158.      * @param None
159.      * @retval None
160.      */
161.
162.     float temp = 0;
163.
164.     int main( void )
165.     {
166.         /* STM32 HAL library initialization*/
167.         HAL_Init();
168.
169.         /* Configure the system clock*/
170.         SystemClock_Config();
171.
172.         /* Configure the debug mode*/
173.         //DBG_Init();
174.
175.         /* Configure the hardware*/
176.         HW_Init();
177.         MX_GPIO_Init();
178.
179.         /* Sensors initialization */
```

Anexo 1. Código

```
180.     MX_I2C1_Init();
181.     PRINTF("i2c Initialized\n");
182.
183.     /*Disable Stand-by mode*/
184.     LPM_SetOffMode(LPM_APPLI_Id , LPM_Disable );
185.
186.     /* Configure the Lora Stack*/
187.     LORA_Init( &LoRaMainCallbacks, &LoRaParamInit);
188.
189.     PRINTF("VERSION: %X\n\r", VERSION);
190.
191.
192.     sht31.begin(&hi2c1, 0x44);
193.
194.     PRINTF("sensors Initialized\n");
195.
196.     LORA_Join(); //Join LoRaWAN OTAA
197.
198.     LoraStartTx( TX_ON_TIMER) ;
199.
200.
201.     while( 1 )
202.     {
203.         DISABLE_IRQ( );
204.         /* if an interrupt has occurred after DISABLE_IRQ, it is kept pending
205.          * and cortex will not enter low power anyway */
206.
207.         #ifndef LOW_POWER_DISABLE
208.             LPM_EnterLowPower( );
209.         #endif
210.
211.         ENABLE_IRQ();
212.
213.         /* USER CODE BEGIN 2 */
214.         /* USER CODE END 2 */
215.     }
216. }
217.
218. static void LORA_HasJoined( void )
219. {
220.     #if( OVER_THE_AIR_ACTIVATION != 0 )
221.         PRINTF("JOINED\n\r");
222.         #ifdef DEBUG
223.             LED_On( LED_GREEN );
224.         #endif
225.     #endif
226.     LORA_RequestClass( LORAWAN_DEFAULT_CLASS );
227. }
228.
229. static void Send( void )
230. {
231.     /* USER CODE BEGIN 3 */
232.     int16_t temperature = 0;
233.     uint16_t humidity = 0;
234.     uint16_t illuminance = 0;
235.     uint8_t batteryLevel;
236.
237.     if ( LORA_JoinStatus () != LORA_SET)
238.     {
239.         /*Not joined, try again later*/
240.         LORA_Join();
241.         return;
242.     }
```

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para
monitorización de patrimonio cultural

```
243.
244.     DBG_PRINTF("SEND REQUEST\n\r");
245.
246.     #ifdef USE_B_L072Z_LRWAN1
247.         TimerInit( &TxLedTimer, OnTimerLedEvent );
248.
249.         TimerSetValue( &TxLedTimer, 200);
250.
251.         LED_On( LED_RED1 );
252.
253.         TimerStart( &TxLedTimer );
254.     #endif
255.
256.     #ifdef CAYENNE_LPP
257.
258.     #ifndef SIMULATE_SENSORS
259.         uint8_t cchannel=0;
260.         //PRINTF("Temp raw: %f", sht31.readTemperature());
261.         temperature = ( int16_t )( sht31.readTemperature() * 10 );    /* in °C * 10 */
262.
263.         PRINTF("%d\n", temperature);
264.         humidity     = ( uint16_t )( sht31.readHumidity * 2 );        /* in %*2 */
265.         illuminance = ( uint16_t )( tsl2591.getLux());
266.         uint32_t i = 0;
267.         batteryLevel = HW_GetBatteryLevel( );                        /* 1 (very low) to 25
268.         4 (fully charged) */
269.     #else
270.         uint8_t cchannel=0;
271.         temperature = ( int16_t )( 21.5 * 10 );    /* in °C * 10 */
272.         humidity     = ( uint16_t )( 51.6 * 2 );    /* in %*2 */
273.         illuminance = ( uint16_t )( 199);
274.         uint32_t i = 0;
275.         batteryLevel = HW_GetBatteryLevel( );      /* 1 (very low) to 25
276.         4 (fully charged) */
277.     #endif
278.
279.     //Relleno el buffer y construyo el payload
280.     AppData.Port = LPP_APP_PORT;
281.
282.     AppData.Buff[i++] = cchannel++;
283.     AppData.Buff[i++] = LPP_DATATYPE_TEMPERATURE;
284.     AppData.Buff[i++] = ( temperature >> 8 ) & 0xFF;
285.     AppData.Buff[i++] = temperature & 0xFF;
286.     AppData.Buff[i++] = cchannel++;
287.     AppData.Buff[i++] = LPP_DATATYPE_HUMIDITY;
288.     AppData.Buff[i++] = humidity & 0xFF;
289.     AppData.Buff[i++] = cchannel++;
290.     AppData.Buff[i++] = LPP_DATATYPE_ILLUMINANCE;
291.     AppData.Buff[i++] = ( illuminance >> 8 ) & 0xFF;
292.     AppData.Buff[i++] = illuminance & 0xFF;
293.     #if defined( REGION_US915 ) || defined( REGION_US915_HYBRID ) || defined( REGION_A
294.     U915 )
295.         /* The maximum payload size does not allow to send more data for lowest DRs */
296.     #else
297.         AppData.Buff[i++] = cchannel++;
298.         AppData.Buff[i++] = LPP_DATATYPE_DIGITAL_INPUT;
299.         AppData.Buff[i++] = batteryLevel*100/254;
300.         //AppData.Buff[i++] = cchannel++;
301.         //AppData.Buff[i++] = LPP_DATATYPE_DIGITAL_OUTPUT;
```

Anexo 1. Código

```
300.     //AppData.Buff[i++] = AppLedStateOn;
301.     #endif /* REGION_XX915 */
302.     #else /* not CAYENNE_LPP */
303.     #endif /* CAYENNE_LPP */
304.     AppData.BuffSize = i;
305.
306.     //Envio parametros por LoRaWAN
307.     LORA_send( &AppData, LORAWAN_DEFAULT_CONFIRM_MSG_STATE);
308.
309.     /* USER CODE END 3 */
310. }
311.
312.
313. static void LORA_RxData( lora_AppData_t *AppData )
314. {
315.     /* USER CODE BEGIN 4 */
316.     DBG_PRINTF("PACKET RECEIVED ON PORT %d\n\r", AppData->Port);
317.
318.     switch (AppData->Port)
319.     {
320.     case 3:
321.         /*this port switches the class*/
322.         if( AppData->BuffSize == 1 )
323.         {
324.             switch ( AppData->Buff[0] )
325.             {
326.             case 0:
327.                 {
328.                     LORA_RequestClass(CLASS_A);
329.                     break;
330.                 }
331.             case 1:
332.                 {
333.                     LORA_RequestClass(CLASS_B);
334.                     break;
335.                 }
336.             case 2:
337.                 {
338.                     LORA_RequestClass(CLASS_C);
339.                     break;
340.                 }
341.             default:
342.                 break;
343.             }
344.         }
345.         break;
346.     case LORAWAN_APP_PORT:
347.         if( AppData->BuffSize == 1 )
348.         {
349.             AppLedStateOn = AppData->Buff[0] & 0x01;
350.             if ( AppLedStateOn == RESET )
351.             {
352.                 PRINTF("LED OFF\n\r");
353.                 LED_Off( LED_BLUE );
354.             }
355.             else
356.             {
357.                 PRINTF("LED ON\n\r");
358.                 LED_On( LED_BLUE );
359.             }
360.         }
361.         break;
362.     case LPP_APP_PORT:
```

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para
monitorización de patrimonio cultural

```
363.     {
364.         AppLedStateOn= (AppData->Buff[2] == 100) ? 0x01 : 0x00;
365.         if ( AppLedStateOn == RESET )
366.         {
367.             PRINTF("LED OFF\n\r");
368.             LED_Off( LED_BLUE );
369.
370.         }
371.         else
372.         {
373.             PRINTF("LED ON\n\r");
374.             LED_On( LED_BLUE );
375.         }
376.         break;
377.     }
378.     default:
379.         break;
380.     }
381.     /* USER CODE END 4 */
382. }
383.
384. static void OnTxTimerEvent( void )
385. {
386.     Send( );
387.     /*Wait for next tx slot*/
388.     TimerStart( &TxTimer);
389. }
390.
391. static void LoraStartTx(TxEventType_t EventType)
392. {
393.     if (EventType == TX_ON_TIMER)
394.     {
395.         /* send everytime timer elapses */
396.         TimerInit( &TxTimer, OnTxTimerEvent );
397.         TimerSetValue( &TxTimer, APP_TX_DUTYCYCLE);
398.         OnTxTimerEvent();
399.     }
400.     else
401.     {
402.         /* send everytime button is pushed */
403.         GPIO_InitTypeDef initStruct={0};
404.
405.         initStruct.Mode =GPIO_MODE_IT_RISING;
406.         initStruct.Pull = GPIO_PULLUP;
407.         initStruct.Speed = GPIO_SPEED_HIGH;
408.
409.         HW_GPIO_Init( USER_BUTTON_GPIO_PORT, USER_BUTTON_PIN, &initStruct );
410.         HW_GPIO_SetIrq( USER_BUTTON_GPIO_PORT, USER_BUTTON_PIN, 0, Send );
411.     }
412. }
413.
414. static void LORA_ConfirmClass ( DeviceClass_t Class )
415. {
416.     PRINTF("switch to class %c done\n\r", "ABC"[Class] );
417.
418.     /*Optionnal*/
419.     /*informs the server that switch has occurred ASAP*/
420.     AppData.BuffSize = 0;
421.     AppData.Port = LORAWAN_APP_PORT;
422.
423.     LORA_send( &AppData, LORAWAN_UNCONFIRMED_MSG);
```

Anexo 1. Código

```
424.     }
425.
426.     static void LORA_TxNeeded ( void )
427.     {
428.         AppData.BuffSize = 0;
429.         AppData.Port = LORAWAN_APP_PORT;
430.
431.         LORA_send( &AppData, LORAWAN_UNCONFIRMED_MSG);
432.     }
433.
434.     #ifdef USE_B_L072Z_LRWAN1
435.     static void OnTimerLedEvent( void )
436.     {
437.         LED_Off( LED_RED1 ) ;
438.     }
439.     #endif
440.
441.
442.     /* I2C1 init function */
443.     static void MX_I2C1_Init(void)
444.     {
445.
446.         hi2c1.Instance = I2C1;
447.         hi2c1.Init.Timing = 0x00000708;
448.         hi2c1.Init.OwnAddress1 = 0;
449.         hi2c1.Init.AddressingMode = I2C_ADDRESSINGMODE_7BIT;
450.         hi2c1.Init.DualAddressMode = I2C_DUALADDRESS_DISABLE;
451.         hi2c1.Init.OwnAddress2 = 0;
452.         hi2c1.Init.OwnAddress2Masks = I2C_OA2_NOMASK;
453.         hi2c1.Init.GeneralCallMode = I2C_GENERALCALL_DISABLE;
454.         hi2c1.Init.NoStretchMode = I2C_NOSTRETCH_DISABLE;
455.
456.         if (HAL_I2C_Init(&hi2c1) != HAL_OK)
457.         {
458.             PRINTF("i2c initialization error\n");
459.             Error_Handler();
460.         }
461.
462.
463.         if (HAL_I2CEx_ConfigAnalogFilter(&hi2c1, I2C_ANALOGFILTER_ENABLE) != HAL_OK)
464.         {
465.             PRINTF("i2c ConfigAnalogFilter error\n");
466.             Error_Handler();
467.         }
468.
469.
470.         if (HAL_I2CEx_ConfigDigitalFilter(&hi2c1, 0) != HAL_OK)
471.         {
472.             PRINTF("i2c ConfigDigitalFilter error\n");
473.             Error_Handler();
474.         }
475.
476.     }
477.
478.     static void MX_GPIO_Init(void)
479.     {
480.
481.         GPIO_InitTypeDef GPIO_InitStruct;
482.
483.         /* GPIO Ports Clock Enable */
484.         __HAL_RCC_GPIOB_CLK_ENABLE();
485.         __HAL_RCC_GPIOA_CLK_ENABLE();
486.
```

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para
monitorización de patrimonio cultural

```
487.     /*Configure GPIO pin Output Level */
488.     HAL_GPIO_WritePin(GPIOB, GPIO_PIN_5|GPIO_PIN_6|GPIO_PIN_7, GPIO_PIN_RESET);
489.
490.     /*Configure GPIO pin Output Level */
491.     HAL_GPIO_WritePin(GPIOA, GPIO_PIN_5, GPIO_PIN_RESET);
492.
493.     /*Configure GPIO pins : PB5 PB6 PB7 */
494.     GPIO_InitStruct.Pin = GPIO_PIN_5|GPIO_PIN_6|GPIO_PIN_7;
495.     GPIO_InitStruct.Mode = GPIO_MODE_OUTPUT_PP;
496.     GPIO_InitStruct.Pull = GPIO_NOPULL;
497.     GPIO_InitStruct.Speed = GPIO_SPEED_FREQ_LOW;
498.     HAL_GPIO_Init(GPIOB, &GPIO_InitStruct);
499.
500.     /*Configure GPIO pin : PA5 */
501.     GPIO_InitStruct.Pin = GPIO_PIN_5;
502.     GPIO_InitStruct.Mode = GPIO_MODE_OUTPUT_PP;
503.     GPIO_InitStruct.Pull = GPIO_NOPULL;
504.     GPIO_InitStruct.Speed = GPIO_SPEED_FREQ_LOW;
505.     HAL_GPIO_Init(GPIOA, &GPIO_InitStruct);
506.
507.     }
508.
509.
510.     /****** (C) COPYRIGHT STMicroelectronics *****END OF FILE*****/
```

Anexo 1. Código

4.2. Configuración LoRaWAN – Comissioning.h

```
1.  /*!  
2.  * @file    commissioning.h  
3.  * @author  Jaime Laborda  
4.  * @version V1.0  
5.  * @date    Marzo-2018  
6.  * @brief   End device commissioning parameters  
7.  *  
8.  */  
9.  /* Define to prevent recursive inclusion -----*/  
10. #ifndef __LORA_COMMISSIONING_H__  
11. #define __LORA_COMMISSIONING_H__  
12.  
13. #ifdef __cplusplus  
14. extern "C" {  
15. #endif  
16.  
17. /* Exported types -----*/  
18. /* Exported constants -----*/  
19. /* External variables -----*/  
20. /* Exported macros -----*/  
21. /* Exported functions -----*/  
22. /*!  
23. * When set to 1 the application uses the Over-the-Air activation procedure  
24. * When set to 0 the application uses the Personalization activation procedure  
25. */  
26. #define OVER_THE_AIR_ACTIVATION          1  
27.  
28. /*!  
29. * Indicates if the end-device is to be connected to a private or public network  
30. */  
31. #define LORAWAN_PUBLIC_NETWORK          true  
32.  
33. /*!  
34. * When set to 1 DevEui is LORAWAN_DEVICE_EUI  
35. * When set to 0 DevEui is automatically generated by calling  
36. *       BoardGetUniqueId function  
37. */  
38. #define STATIC_DEVICE_EUI                0  
39.  
40. /*!  
41. * Mote device IEEE EUI (big endian)  
42. *  
43. * \remark see STATIC_DEVICE_EUI comments  
44. */  
45. #define LORAWAN_DEVICE_EUI                { 0x34, 0x34, 0x37, 0x31, 0x7F, 0x36,  
46.     0x7B, 0x0E }  
47.  
48. /*!  
49. * Application IEEE EUI (big endian)  
50. */  
51. #define LORAWAN_APPLICATION_EUI          { 0x70, 0xB3, 0xD5, 0x7E, 0xD0, 0x00,  
52.     0xB0, 0xC1 }  
53. /*!  
54. * AES encryption/decryption cipher application key  
55. */  
56. #define LORAWAN_APPLICATION_KEY          { 0x67, 0x87, 0x6F, 0x73, 0xD8, 0xE2,  
57.     0x52, 0xFC, 0x47, 0x6D, 0xF0, 0x74, 0x7F, 0x00, 0x50, 0x79 }  
58.
```

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para
monitorización de patrimonio cultural

```
58. #if( OVER_THE_AIR_ACTIVATION == 0 )
59.
60. /*!
61. * Current network ID
62. */
63. #define LORAWAN_NETWORK_ID                ( uint32_t )0
64.
65. /*!
66. * When set to 1 DevAdd is LORAWAN_DEVICE_ADDRESS
67. * When set to 0 DevAdd is automatically generated using
68. * a pseudo random generator seeded with a value derived from
69. * BoardUniqueId value
70. */
71. #define STATIC_DEVICE_ADDRESS              0
72. /*!
73. * Device address on the network (big endian)
74. *
75. * \remark see STATIC_DEVICE_ADDRESS comments
76. */
77. #define LORAWAN_DEVICE_ADDRESS            ( uint32_t )0x0100000a
78.
79. /*!
80. * AES encryption/decryption cipher network session key
81. */
82. #define LORAWAN_NWKSKEY                    { 0x2B, 0x7E, 0x15, 0x16, 0x28, 0xAE,
83. 0xD2, 0xA6, 0xAB, 0xF7, 0x15, 0x88, 0x09, 0xCF, 0x4F, 0x3C }
84. /*!
85. * AES encryption/decryption cipher application session key
86. */
87. #define LORAWAN_APPSKEY                    { 0x2B, 0x7E, 0x15, 0x16, 0x28, 0xAE,
88. 0xD2, 0xA6, 0xAB, 0xF7, 0x15, 0x88, 0x09, 0xCF, 0x4F, 0x3C }
89. #endif /* OVER_THE_AIR_ACTIVATION == 0 */
90.
91.
92. #ifdef __cplusplus
93. }
94. #endif
95.
96. #endif /* __LORA_COMMISSIONING_H__ */
```


Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para
monitorización de patrimonio cultural

Contenido Anexo 2. Hojas de características

A2.1.	Hoja de datos del sensor Sensirion SHT31	2
A2.2.	Hoja de datos del sensor AMS TSL2591	11
A2.3.	Hoja de datos batería Energizer L92-FR03	19
A2.4.	Hoja de datos B-L072Z-LRWAN1 Discovery LoRa.....	21
A2.1.	Esquema eléctrico B-L072Z-LRWAN1 Discovery LoRa	31

Anexo 2. Hojas de características

A2.1. Hoja de datos del sensor Sensirion SHT31



Datasheet SHT3x-DIS

Humidity and Temperature Sensor

- Fully calibrated, linearized, and temperature compensated digital output
- Wide supply voltage range, from 2.15 V to 5.5 V
- I2C Interface with communication speeds up to 1 MHz and two user selectable addresses
- Typical accuracy of ± 1.5 %RH and ± 0.1 °C for SHT35
- Very fast start-up and measurement time
- Tiny 8-Pin DFN package



Product Summary

SHT3x-DIS is the next generation of Sensirion's temperature and humidity sensors. It builds on a new CMOSens® sensor chip that is at the heart of Sensirion's new humidity and temperature platform. The SHT3x-DIS has increased intelligence, reliability and improved accuracy specifications compared to its predecessor. Its functionality includes enhanced signal processing, two distinctive and user selectable I2C addresses and communication speeds of up to 1 MHz. The DFN

package has a footprint of 2.5 x 2.5 mm² while keeping a height of 0.9 mm. This allows for integration of the SHT3x-DIS into a great variety of applications. Additionally, the wide supply voltage range of 2.15 V to 5.5 V guarantees compatibility with diverse assembly situations. All in all, the SHT3x-DIS incorporates 15 years of knowledge of Sensirion, the leader in the humidity sensor industry.

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

Benefits of Sensirion's CMOSens® Technology

- High reliability and long-term stability
- Industry-proven technology with a track record of more than 15 years
- Designed for mass production
- High process capability
- High signal-to-noise ratio

Content

1	Sensor Performance.....	2
2	Specifications	6
3	Pin Assignment	8
4	Operation and Communication.....	9
5	Packaging.....	16
6	Shipping Package	18
7	Quality	19
8	Ordering Information.....	19
9	Further Information.....	19

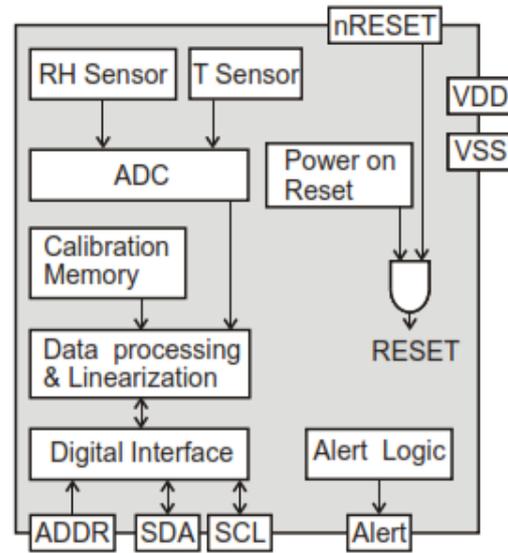


Figure 1 Functional block diagram of the SHT3x-DIS. The sensor signals for humidity and temperature are factory calibrated, linearized and compensated for temperature and supply voltage dependencies.

1 Sensor Performance

Humidity Sensor Specification

Parameter	Condition	Value	Units
SHT30 Accuracy tolerance ¹	Typ.	±2	%RH
	Max.	Figure 2	-
SHT31 Accuracy tolerance ¹	Typ.	±2	%RH
	Max.	Figure 3	-
SHT35 Accuracy tolerance ¹	Typ.	±1.5	%RH
	Max.	Figure 4	-
Repeatability ²	Low, typ.	0.21	%RH
	Medium, typ.	0.15	%RH
	High, typ.	0.08	%RH
Resolution	Typ.	0.01	%RH
Hysteresis	at 25°C	±0.8	%RH
Specified range ³	extended ⁴	0 to 100	%RH
Response time ⁵	τ63%	8 ⁶	s
Long-term drift	Typ. ⁷	<0.25	%RH/yr

Table 1 Humidity sensor specification.

Anexo 2. Hojas de características

Temperature Sensor Specification

Parameter	Condition	Value	Units
SHT30 Accuracy tolerance ¹	typ., 0°C to 65°C	±0.2	°C
	Max.	Figure 8	-
SHT31 Accuracy tolerance ¹	typ., 0°C to 90°C	±0.2	°C
	Max.	Figure 9	-
SHT35 Accuracy tolerance ¹	typ., 20°C to 60°C	±0.1	°C
	Max.	Figure 10	-
Repeatability ²	Low, typ.	0.15	°C
	Medium, typ.	0.08	°C
	High, typ.	0.04	°C
Resolution	Typ.	0.01	°C
Specified Range	-	-40 to 125	°C
Response time ⁸	$\tau_{63\%}$	>2	s
Long Term Drift	max	<0.03	°C/yr

Table 2 Temperature sensor specification.

Humidity Sensor Performance Graphs

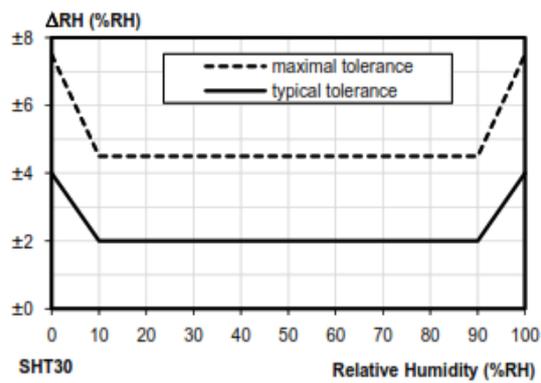


Figure 2 Tolerance of RH at 25°C for SHT30.

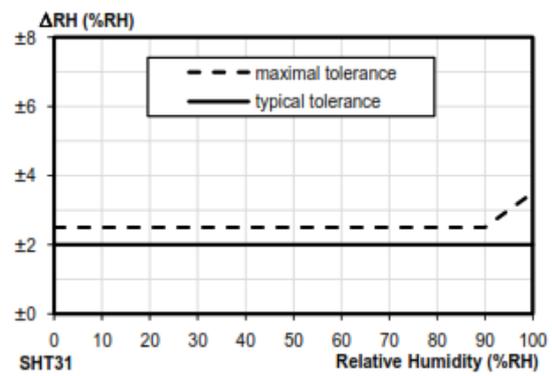


Figure 3 Tolerance of RH at 25°C for SHT31.

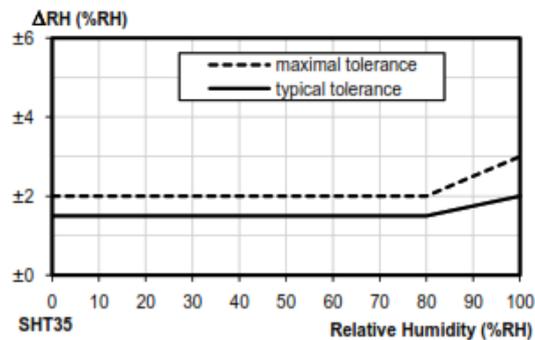


Figure 4 Tolerance of RH at 25°C for SHT35.

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

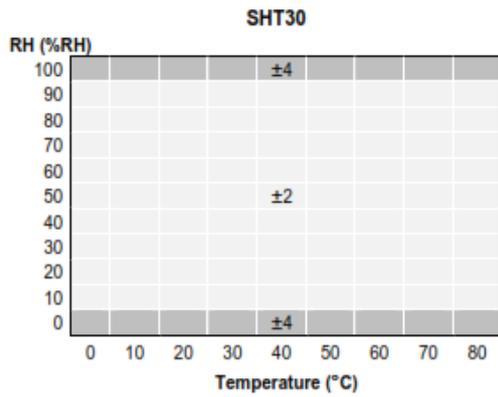


Figure 5 Typical tolerance of RH over T for SHT30.

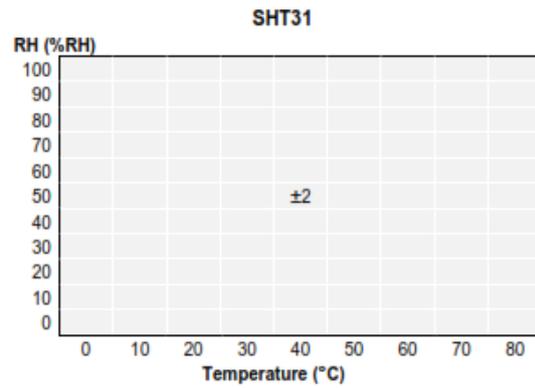


Figure 6 Typical tolerance of RH over T for SHT31.

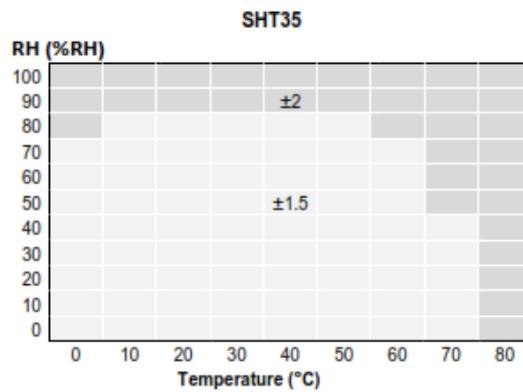


Figure 7 Typical tolerance of RH over T for SHT35.

Anexo 2. Hojas de características

Temperature Sensor Performance Graphs

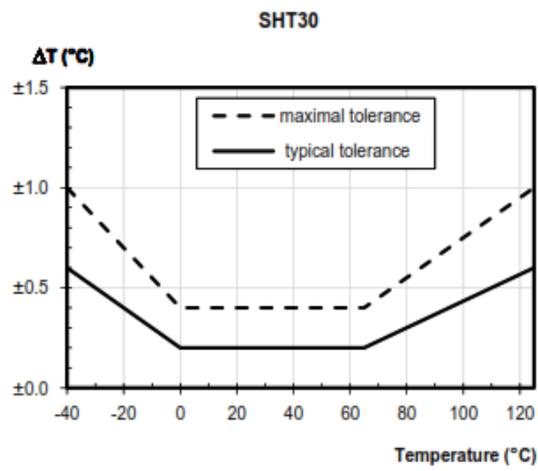


Figure 8 Temperature accuracy of the SHT30 sensor.

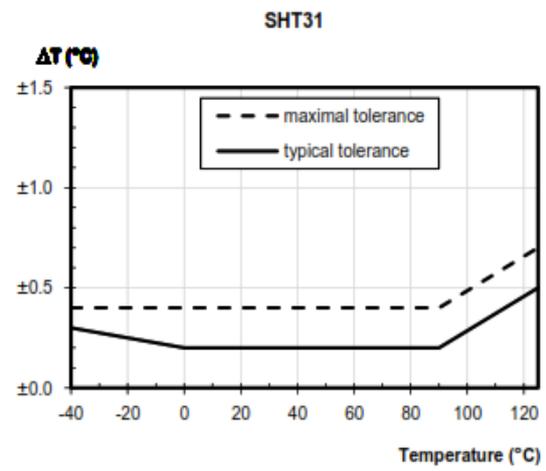


Figure 9 Temperature accuracy of the SHT31 sensor.

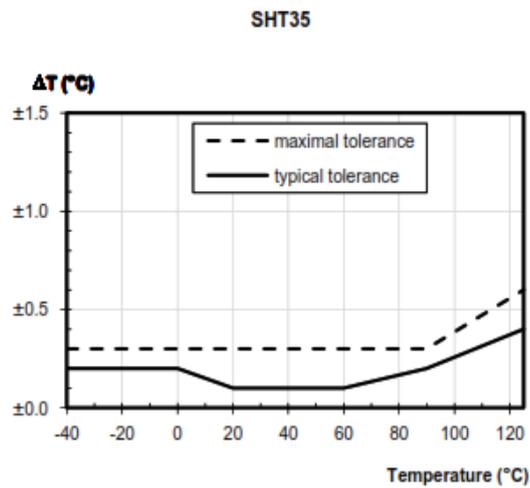


Figure 10 Temperature accuracy of the SHT35 sensor.

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

2 Specifications

2.1 Electrical Specifications

Parameter	Symbol	Condition	Min.	Typ.	Max.	Units	Comments
Supply voltage	V _{DD}		2.15	3.3	5.5	V	
Power-up/down level	V _{POR}		1.8	2.10	2.15	V	
Slew rate change of the supply voltage	V _{DD,slew}		-	-	20	V/ms	Voltage changes on the VDD line between V _{DD,min} and V _{DD,max} should be slower than the maximum slew rate; faster slew rates may lead to reset;
Supply current	I _{DD}	idle state (single shot mode) T=25°C	-	0.2	2.0	μA	Current when sensor is not performing a measurement during single shot mode
		idle state (single shot mode) T=125°C	-	-	6.0		
		idle state (periodic data acquisition mode)	-	45	-	μA	Current when sensor is not performing a measurement during periodic data acquisition mode
		Measuring	-	600	1500	μA	Current consumption while sensor is measuring
		Average	-	1.7	-	μA	Current consumption (operation with one measurement per second at lowest repeatability, single shot mode)
Alert Output driving strength	IOH			1.5x V _{DD}		mA	See also section 3.5
Heater power	P _{Heater}	Heater running	3.6	-	33	mW	Depending on the supply voltage

Table 3 Electrical specifications, typical values are valid for T=25°C, min. & max. values for T=-40°C ... 125°C

2.3 Absolute Minimum and Maximum Ratings

Stress levels beyond those listed in Table 6 may cause permanent damage to the device or affect the reliability of the sensor. These are stress ratings only and functional operation of the device at these conditions is not guaranteed. Ratings are only tested each at a time.

Parameter	Rating	Units
Supply voltage V _{DD}	-0.3 to 6	V
Max Voltage on pins (pin 1 (SDA); pin 2 (ADDR); pin 3 (ALERT); pin 4 (SCL); pin 6 (nRESET))	-0.3 to V _{DD} +0.3	V
Input current on any pin	±100	mA
Operating temperature range	-40 to 125	°C
Storage temperature range	-40 to 150	°C
ESD HBM (human body model) ⁹	4	kV
ESD CDM (charge device model) ¹⁰	750	V

Table 6 Minimum and maximum ratings; voltages may only be applied for short time periods.

Anexo 2. Hojas de características

3 Pin Assignment

The SHT3x-DIS comes in a 8-pin DFN package – see Table 7.

Pin	Name	Comments
1	SDA	Serial data; input / output
2	ADDR	Address pin; input; connect to either logic high or low, do not leave floating
3	ALERT	Indicates alarm condition; output; must be left floating if unused
4	SCL	Serial clock; input / output
5	VDD	Supply voltage; input
6	nRESET	Reset pin active low; input; if not used it is recommended to be left floating; can be connected to VDD with a series resistor of $R \geq 2 \text{ k}\Omega$
7	R	No electrical function; to be connected to VSS
8	VSS	Ground

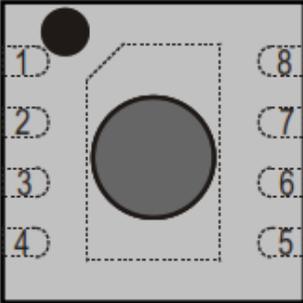


Table 7 SHT3x-DIS pin assignment (transparent top view). Dashed lines are only visible if viewed from below. The die pad is internally connected to VSS.

SHT3x-DIS	I2C Address in Hex. representation	Condition
I2C address A	0x44 (default)	ADDR (pin 2) connected to logic low
I2C address B	0x45	ADDR (pin 2) connected to logic high

Table 8 I2C device addresses.

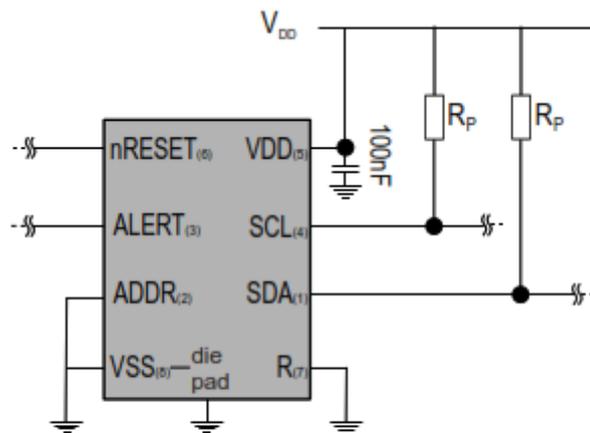


Figure 11 Typical application circuit. Please note that the positioning of the pins does not reflect the position on the real sensor. This is shown in Table 7.

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

4.14 Communication Timing

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Units	Comments
SCL clock frequency	f _{SCL}		0	-	1000	kHz	
Hold time (repeated) START condition	t _{HD,STA}	After this period, the first clock pulse is generated	0.24	-	-	μs	
LOW period of the SCL clock	t _{LOW}		0.53	-	-	μs	
HIGH period of the SCL clock	t _{HIGH}		0.26	-	-	μs	
SDA hold time	t _{HD,DAT}		0	-	250	ns	Transmitting data
			0	-	-	ns	Receiving data
SDA set-up time	t _{SU,DAT}		100	-	-	ns	
SCL/SDA rise time	t _R		-	-	300	ns	
SCL/SDA fall time	t _F		-	-	300	ns	
SDA valid time	t _{VD,DAT}		-	-	0.9	μs	
Set-up time for a repeated START condition	t _{SU,STA}		0.26	-	-	μs	
Set-up time for STOP condition	t _{SU,STO}		0.26	-	-	μs	
Capacitive load on bus line	CB		-	-	400	pF	
Low level input voltage	V _{IL}		0	-	0.3xV _{DD}	V	
High level input voltage	V _{IH}		0.7xV _{DD}	-	1xV _{DD}	V	
Low level output voltage	V _{OL}	3 mA sink current	-	-	0.4	V	

Table 21 Timing specifications for I2C communication, valid for T=-40°C ... 125°C and VDD = VDD_{min} ... VDD_{max}. The nomenclature above is according to the I2C (UM10204, Rev. 6, April 4, 2014).

Relative humidity conversion formula (result in %RH):

$$RH = 100 \cdot \frac{S_{RH}}{2^{16} - 1}$$

Temperature conversion formula (result in °C & °F):

$$T [^{\circ}C] = -45 + 175 \cdot \frac{S_T}{2^{16} - 1}$$

$$T [^{\circ}F] = -49 + 315 \cdot \frac{S_T}{2^{16} - 1}$$

S_{RH} and S_T denote the raw sensor output for humidity and temperature, respectively. The formulas work only correctly when S_{RH} and S_T are used in decimal representation.

Property	Value
Name	CRC-8
Width	8 bit
Protected data	read and/or write data
Polynomial	0x31 (x ⁸ + x ⁵ + x ⁴ + 1)
Initialization	0xFF
Reflect input	False
Reflect output	False
Final XOR	0x00
Examples	CRC (0xBEEF) = 0x92

Table 20 I2C CRC properties.

Anexo 2. Hojas de características

5.2 Package Outline

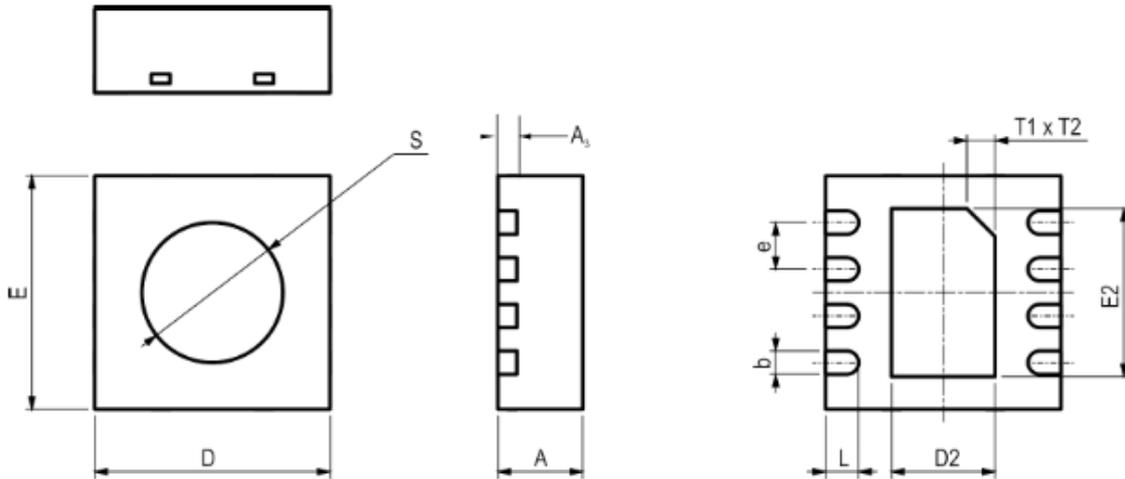
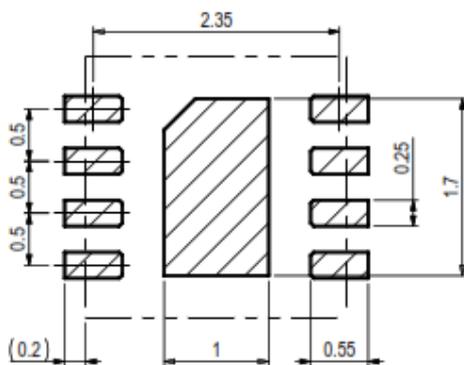


Figure 14 Dimensional drawing of SHT3x-DIS sensor package

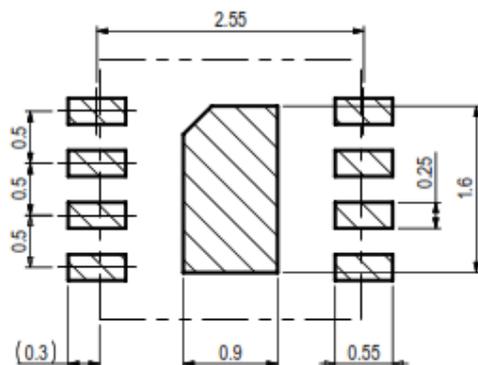
Parameter	Symbol	Min	Nom.	Max	Units	Comments
Package height	A	0.8	0.9	1	mm	
Leadframe height	A3	-	0.2	-	mm	
Pad width	b	0.2	0.25	0.3	mm	
Package width	D	2.4	2.5	2.6	mm	
Center pad length	D2	1	1.1	1.2	mm	
Package length	E	2.4	2.5	2.6	mm	
Center pad width	E2	1.7	1.8	1.9	mm	
Pad pitch	e	-	0.5		mm	
Pad length	L	0.25	0.35	0.45	mm	
Max cavity	S	-	-	1.5	mm	Only as guidance. This value includes all tolerances, including displacement tolerances. Typically the opening will be smaller.
Center pad marking	T1xT2	-	0.3x45°	-	mm	indicates the position of pin 1

Table 22 Package outline.

Recommended Land Pattern



Recommended Stencil Aperture



Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

A2.2. Hoja de datos del sensor AMS TSL2591



TSL2591

Datasheet - Apr. 2013 - ams163.5

General Description

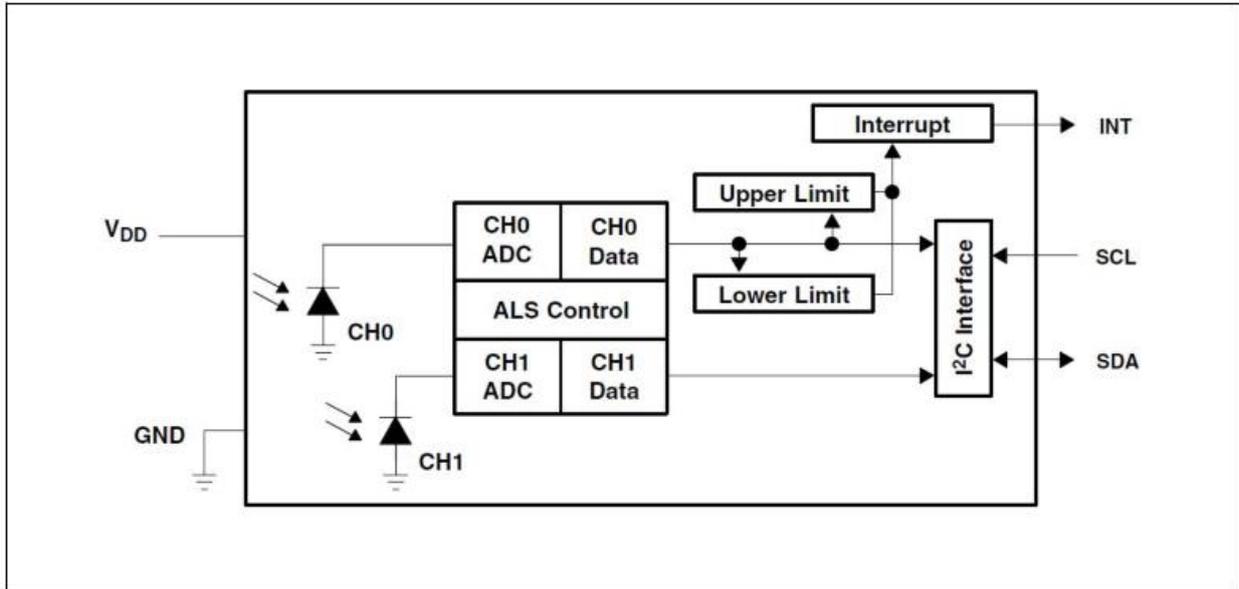
The TSL2591 is a very-high sensitivity light-to-digital converter that transforms light intensity into a digital signal output capable of direct I²C interface. The device combines one broadband photodiode (visible plus infrared) and one infrared-responding photodiode on a single CMOS integrated circuit. Two integrating ADCs convert the photodiode currents into a digital output that represents the irradiance measured on each channel. This digital output can be input to a microprocessor where illuminance (ambient light level) in lux is derived using an empirical formula to approximate the human eye response. The TSL2591 supports a traditional level style interrupt that remains asserted until the firmware clears it.

Figure TSL2591 – 1:
 Key Benefits and Features

Benefits	Features
Approximates Human Eye Response	Dual Diode
Flexible Operation	Programmable Analog Gain and Integration Time
Suited for Operation Behind Dark Glass	600M:1 Dynamic Range
Low Operating Overhead	<ul style="list-style-type: none"> • Two Internal Interrupt Sources • Programmable Upper and Lower Thresholds • One Interrupt Includes Programmable Persistence Filter
Low Power 3.0 μA Sleep State	User Selectable Sleep Mode
I ² C Fast Mode Compatible Interface	<ul style="list-style-type: none"> • Data Rates up to 400 kbit/s • Input Voltage Levels Compatible with 3.0V Bus

Anexo 2. Hojas de características

Figure TSL2591 – 2:
 Block Diagram



Pin Assignment

The TSL2591 pin assignments are described below.

Figure TSL2591 – 3:
 Pin Diagram

Package FN Dual Flat No-Lead (Top View): Package drawing is not to scale.

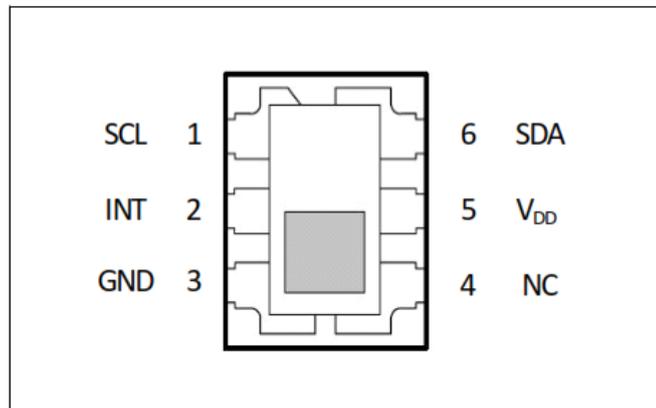


Figure TSL2591 – 4:
 Pin Description

Pin Number	Pin Name	Description
1	SCL	I ² C serial clock input terminal
2	INT	Interrupt — open drain output (active low).
3	GND	Power supply ground. All voltages are referenced to GND.
4	NC	No connect — do not connect.
5	V _{DD}	Supply voltage
6	SDA	I ² C serial data I/O terminal

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

Absolute Maximum Ratings

Stresses beyond those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. These are stress ratings only. Functional operation of the device at these or any other conditions beyond those indicated under “Operating Conditions” is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Figure TSL2591 – 6:
Absolute Maximum Ratings

Parameter	Min	Max	Units	Comments
Supply voltage, V_{DD}		3.8	V	All voltages are with respect to GND
Input terminal voltage	-0.5	3.8	V	
Output terminal voltage	-0.5	3.8	V	
Output terminal current	-1	20	mA	
Storage temperature range, T_{stg}	-40	85	°C	
ESD tolerance, human body model		2000	V	

Anexo 2. Hojas de características

Electrical Characteristics

All limits are guaranteed. The parameters with min and max values are guaranteed with production tests or SQC (Statistical Quality Control) methods.

Figure TSL2591 – 7:
Recommended Operating Conditions

Symbol	Parameter	Min	Typ	Max	Units
V_{DD}	Supply voltage	2.7	3	3.6	V
T_A	Operating free-air temperature	-30		70	°C

Figure TSL2591 – 8:
Operating Characteristics, $V_{DD}=3V$, $T_A=25^\circ C$ (unless otherwise noted)

Symbol	Parameter	Conditions	Min	Typ	Max	Units
I_{DD}	Supply Current	Active Sleep state - no I ² C activity		275 2.3	325 4	μA
V_{OL}	INT, SDA output low voltage	3mA sink current 6mA sink current	0 0		0.4 0.6	V
I_{LEAK}	Leakage current, SDA, SCL, INT pins		-5		5	μA
V_{IH}	SCL, SDA input high voltage		0.7 V_{DD}			V
V_{IL}	SCL, SDA input low voltage				0.3 V_{DD}	V

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

Timing Characteristics

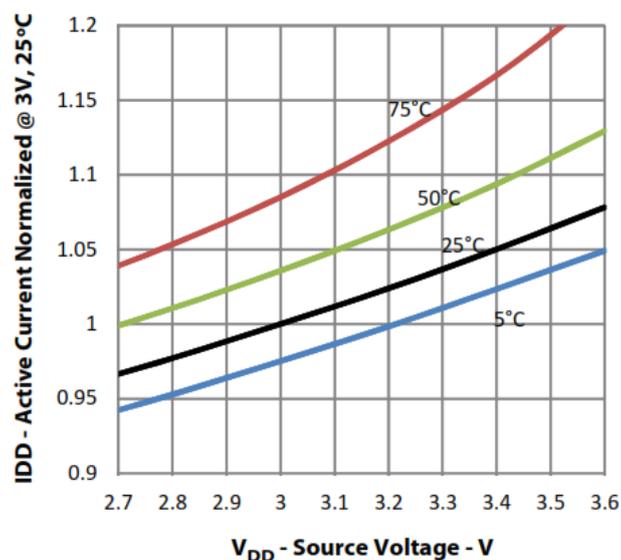
The timing characteristics of TSL2591 are given below.

Figure TSL2591 – 10:
 AC Electrical Characteristics, $V_{DD} = 3\text{ V}$, $T_A = 25^\circ\text{C}$ (unless otherwise noted)

Parameter†	Description	Min	Typ	Max	Units
$f_{(SCL)}$	Clock frequency (I ² C only)	0		400	kHz
$t_{(BUF)}$	Bus free time between start and stop condition	1.3			μs
$t_{(HDSTA)}$	Hold time after (repeated) start condition. After this period, the first clock is generated.	0.6			μs
$t_{(SUSTA)}$	Repeated start condition setup time	0.6			μs
$t_{(SUSTO)}$	Stop condition setup time	0.6			μs
$t_{(HDDAT)}$	Data hold time	0			μs
$t_{(SUDAT)}$	Data setup time	100			ns
$t_{(LOW)}$	SCL clock low period	1.3			μs
$t_{(HIGH)}$	SCL clock high period	0.6			μs
t_F	Clock/data fall time			300	ns
t_R	Clock/data rise time			300	ns
C_i	Input pin capacitance			10	pF

Figure TSL2591 – 14:
 Normalized I_{DD} vs. V_{DD} and Temperature

I_{DD} vs. V_{DD} vs. Temp: Effect of supply voltage and temperature on active current.

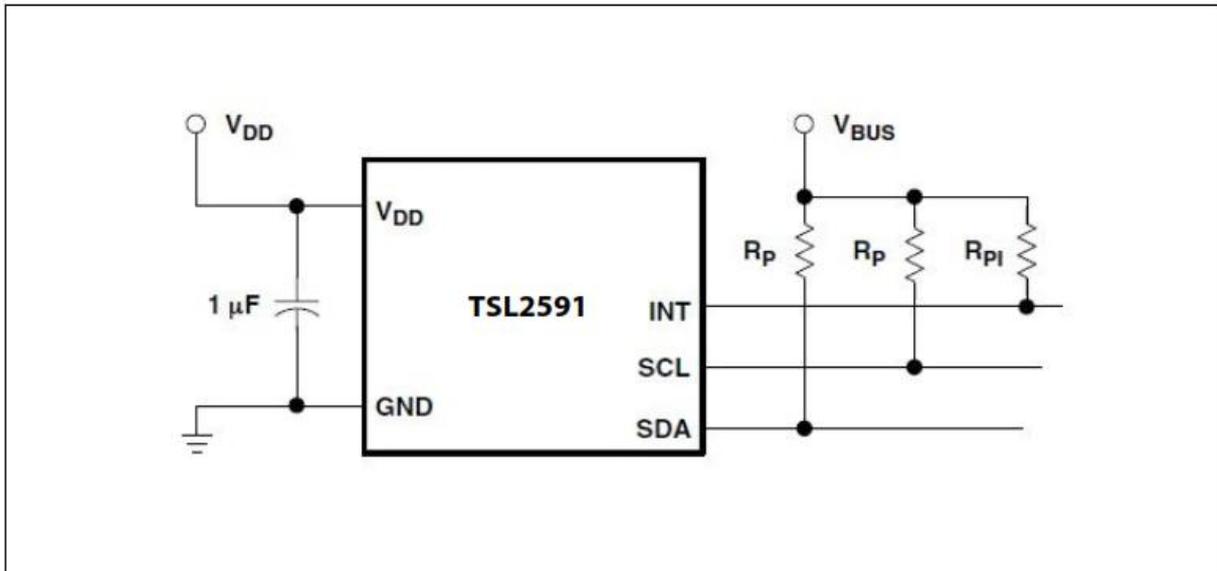


Anexo 2. Hojas de características

Application Information

Figure TSL2591 - 17 shows a typical hardware application circuit. A 1- μF low-ESR decoupling capacitor should be placed as close as possible to the V_{DD} pin. V_{BUS} in this figure refers to the I²C bus voltage, which is equal to V_{DD} .

Figure TSL2591 - 17:
Typical Application Hardware Circuit



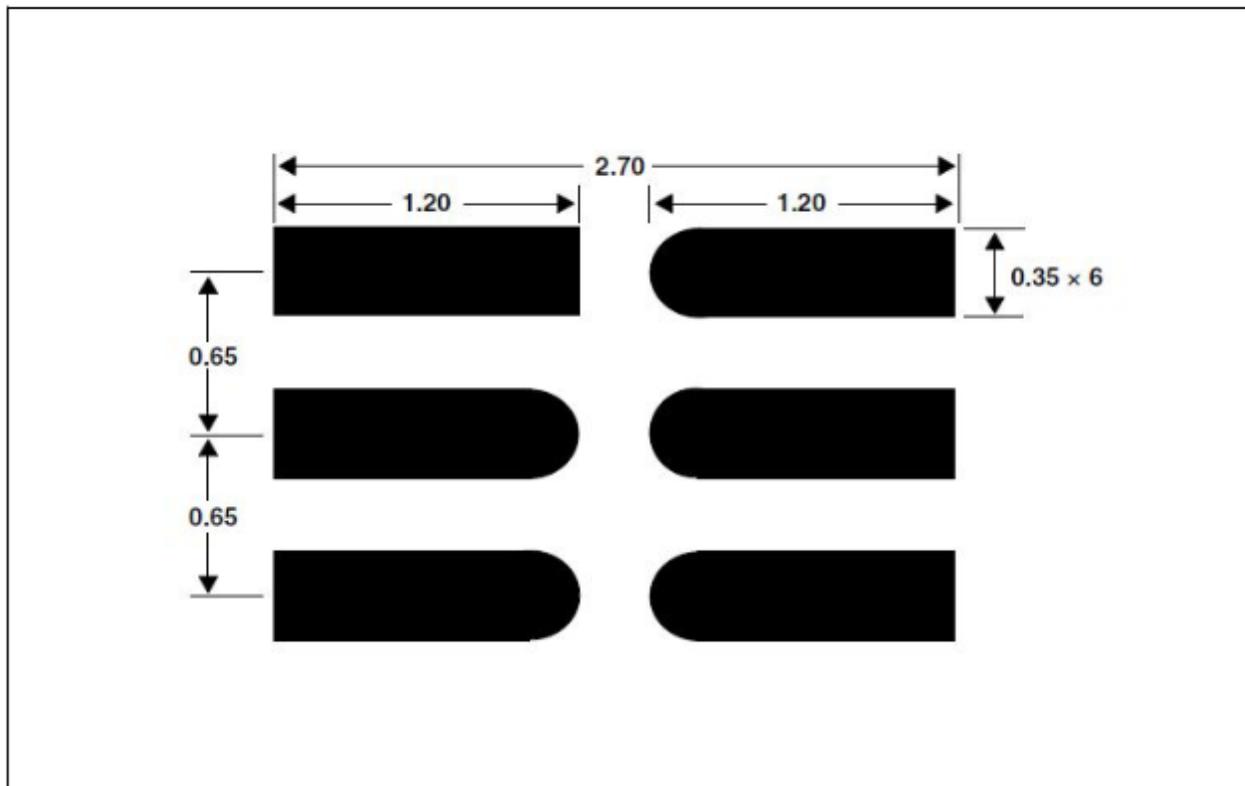
The I²C signals and the Interrupt are open-drain outputs and require pull-up resistors. The pull-up resistor (R_{P}) value is a function of the I²C bus speed, the I²C bus voltage, and the capacitive load. The ams EVM running at 400 kbps, uses 1.5-k Ω resistors. A 10-k Ω pull-up resistor (R_{PI}) can be used for the interrupt line.

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

PCB Pad Layout

Suggested land pattern based on the IPC-7351B Generic Requirements for Surface Mount Design and Land Pattern Standard (2010) for the small outline no-lead (SON) package is shown in Figure TSL2591 - 18.

Figure TSL2591 - 18:
Suggested FN Package PCB Layout (Top View)



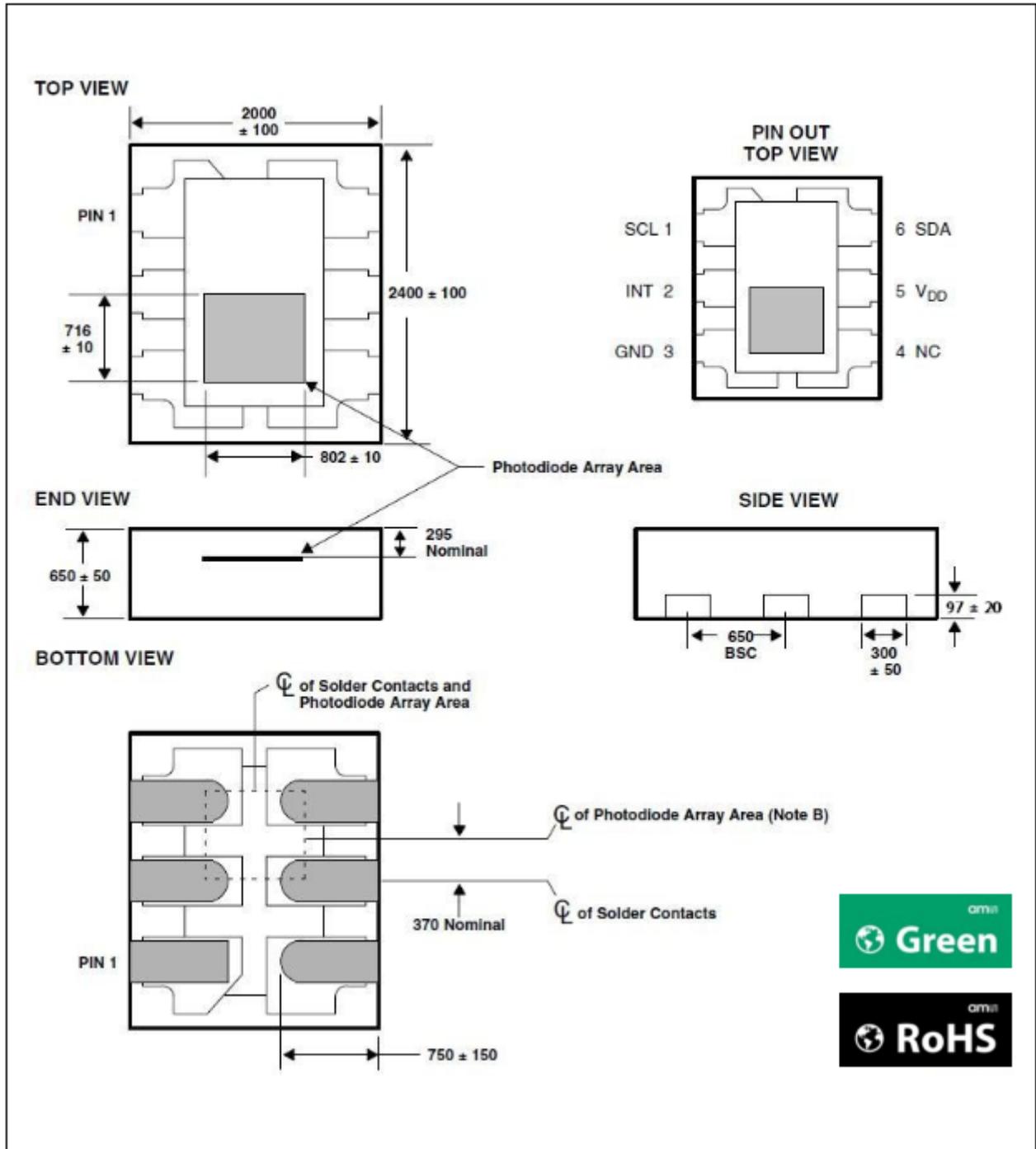
Notes:

1. All linear dimensions are in millimeters.
2. This drawing is subject to change without notice.

Anexo 2. Hojas de características

Package Drawings and Markings

Figure TSL2591 – 19:
 FN Package – Dual Flat No-Lead Packaging Configuration



Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

A2.3. Hoja de datos batería Energizer L92-FR03

PRODUCT DATASHEET



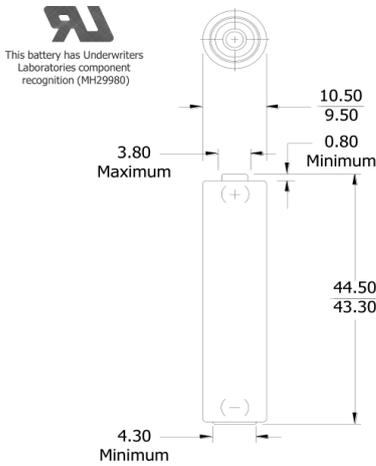
European Region
+ 44 (0) 208 920 2306
www.energizer.eu

ENERGIZER L92-FR03

AAA



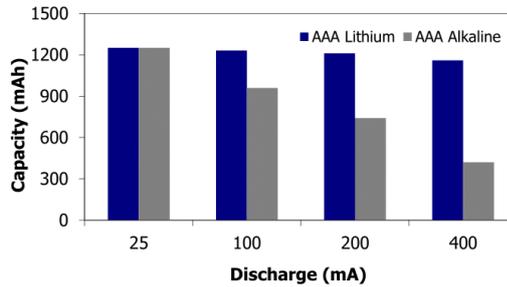
Industry Standard Dimensions (millimeters)



Classification:	"Cylindrical Lithium"
Chemical System:	Lithium/Iron Disulfide (Li/FeS ₂)
Designation:	IEC-FR03
Nominal Voltage:	1.5 Volts
Storage Temp:	-40°C to 60°C
Operating Temp:	-40°C to 60°C
Typical Weight:	7.6 grams
Typical Volume:	3.8 cubic centimeters
Max Discharge:	1.5 Amps Continuous
(single battery only)	2.0 Amps Pulse (2 sec on / 8 sec off)
Max Rev Current:	2 uA
Typical Li Content:	0.5 grams
Typical IR:	90 to 150 milliohms
Shelf Life:	15 years at 21°C

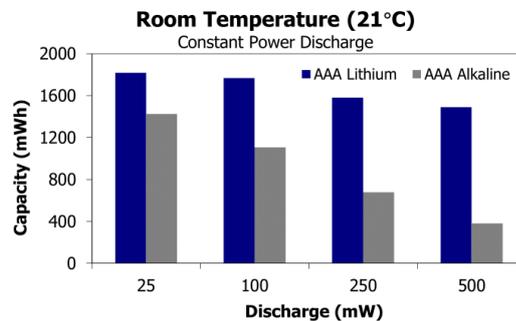
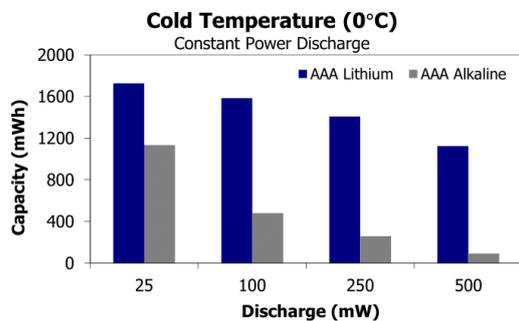
Milliamp-Hours Capacity

Constant Current Discharge to 0.9 Volts at 21°C



Milliwatt-Hours Capacity at Cold/Room Temperature

Constant Power Discharge to 1.0 Volts at 0°C and 21°C
Recommended Lithium Operating/Storage Temperature: -40°C to 60°C



Important Notice

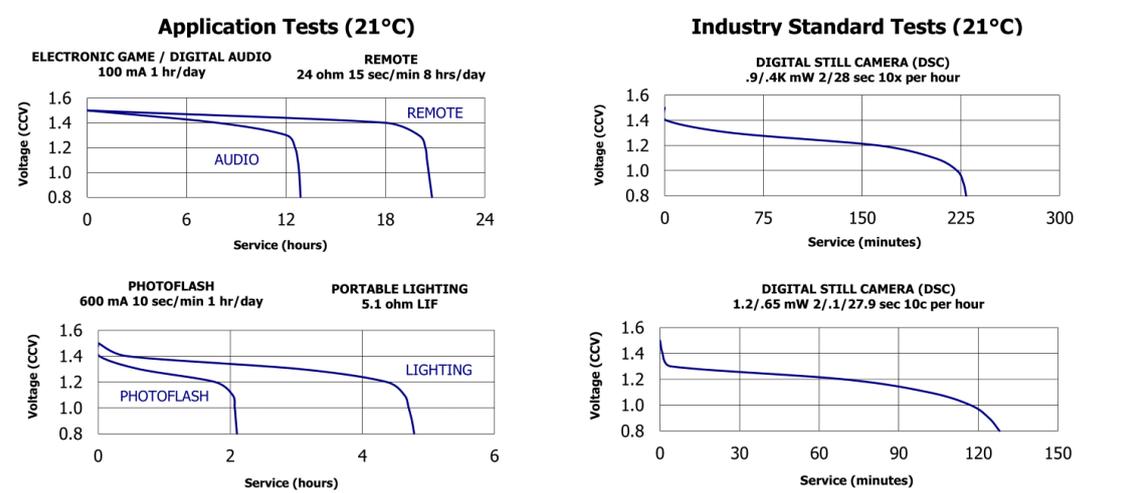
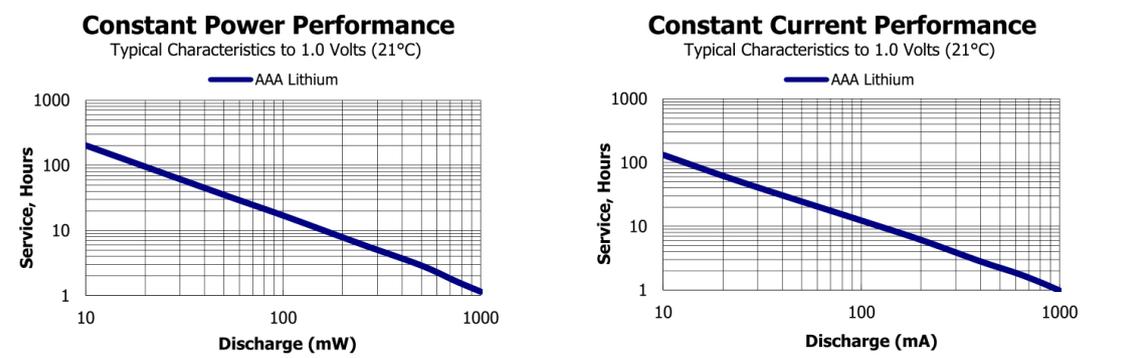
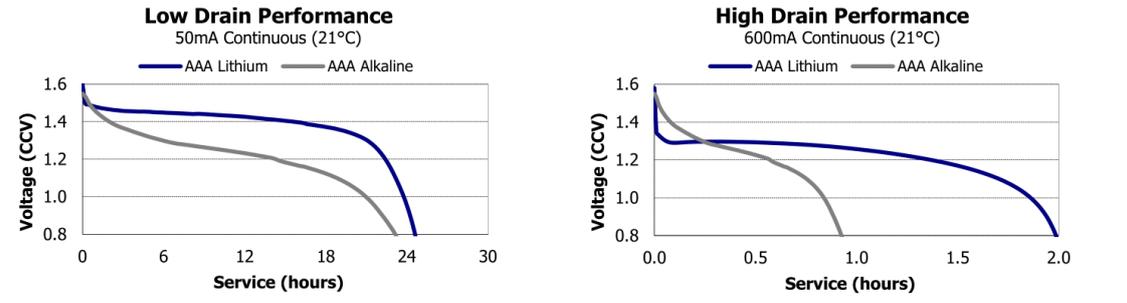
This datasheet contains typical information specific to products manufactured at the time of its publication.
©Energizer Holdings, Inc. - Contents herein do not constitute a warranty.

Anexo 2. Hojas de características

PRODUCT DATASHEET **Energizer**
European Region
+ 44 (0) 208 920 2306
www.energizer.eu

ENERGIZER L92-FR03 **AAA**

Typical Discharge Curve Characteristics
Constant Current Discharge at 21°C (low and high drains)



Important Notice
This datasheet contains typical information specific to products manufactured at the time of its publication.
©Energizer Holdings, Inc. - Contents herein do not constitute a warranty.

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

A2.4. Hoja de datos B-L072Z-LRWAN1 Discovery LoRa



UM2115 User manual

Discovery kit for LoRaWAN™, Sigfox™, and LPWAN protocols with STM32L0

Introduction

The B-L072Z-LRWAN1 Discovery kit embeds the CMWX1ZZABZ-091 LoRa®/Sigfox™ module (Murata). This Discovery kit allows users to develop easily applications with the STM32L072CZ and the LoRa®/Sigfox™ RF connectivity in one single module.

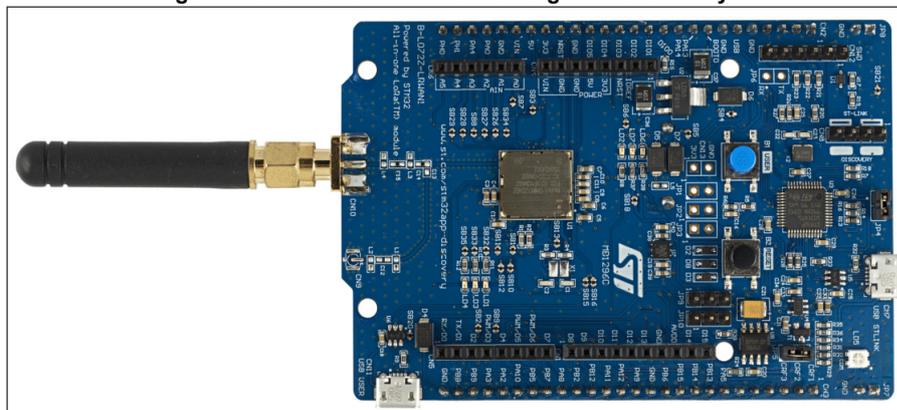
The B-L072Z-LRWAN1 Discovery kit has the full set of features available in the STM32L0 Series and offers ultra-low-power and LoRa®/Sigfox™ RF features. The B-L072Z-LRWAN1 Discovery kit is a low-cost and easy-to-use development kit to quickly evaluate and start a development with an STM32L072CZ microcontroller.

The B-L072Z-LRWAN1 Discovery kit includes LoRa®/Sigfox™ RF interface, LEDs, push-buttons, antenna, Arduino™ Uno V3 connectors, USB 2.0 FS connector in Micro-B format. The integrated ST-LINK/V2-1 provides an embedded in-circuit debugger and programmer for the STM32L0 MCUs.

The LoRaWAN™ stack is certified class A and C compliant. It is available inside the I-CUBE-LRWAN firmware package. The Sigfox™ stack is RCZ1, RCZ2, and RCZ4 certified. It is available inside the X-CUBE-SFOX expansion package.

To help users setting up a complete node (LoRaWAN™, Sigfox™, or both), the B-L072Z-LRWAN1 Discovery kit comes with the STM32 comprehensive free software libraries and examples available with the STM32Cube package, as well as a direct access to the Arm® Mbed Enabled™ resources at the <http://mbed.org> website.

Figure 1. B-L072Z-LRWAN1 LoRa®/Sigfox™ Discovery kit



1. Picture is not contractual.



Anexo 2. Hojas de características

1 Features

The B-L072Z-LRWAN1 Discovery kit offers the following features:

- CMWX1ZZABZ-091 LoRa[®]/Sigfox[™] module (Murata)
 - Embedded ultra-low-power STM32L072CZ Series MCUs, based on Arm[®] Cortex[®]-M0+ core, with 192 Kbytes of Flash memory, 20 Kbytes of RAM, 20 Kbytes of EEPROM
 - USB 2.0 FS
 - 4-channel, 12-bit ADC, 2xDAC
 - 6-bit timers, LP-UART, I²C and SPI
 - Embedded SX1276 transceiver
 - LoRa[®], FSK, GFSK, MSK, GMSK and OOK modulations (+ Sigfox[™] compatibility)
 - +14 dBm or +20 dBm selectable output power
 - 157 dB maximum link budget
 - Programmable bit rate up to 300 kbit/s
 - High sensitivity: down to -137 dBm
 - Bullet-proof front end: IIP3 = -12.5 dBm
 - 89 dB blocking immunity
 - Low RX current of 10 mA, 200 nA register retention
 - Fully integrated synthesizer with a resolution of 61 Hz
 - Built-in bit synchronizer for clock recovery
 - Sync word recognition
 - Preamble detection
 - 127 dB+ dynamic range RSSI
- Including 50 ohm SMA RF antenna
- 1 user and reset push-buttons
- Board connectors:
 - USB FS connector
 - SMA and U.FL RF
- Board expansion connectors:
 - Arduino[™] Uno V3
- 7 LEDs:
 - 4 general-purpose LEDs
 - 5 V-power LED
 - ST-LINK-communication LED
 - Fault-power LED
- Flexible power-supply options: ST-LINK USB V_{BUS} or external sources
- On-board ST-LINK/V2-1 debugger/programmer with USB re-enumeration capability: mass storage, virtual COM port and debug port
- Comprehensive free software libraries and examples available with the STM32Cube package

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

UM2115

Hardware layout and configuration

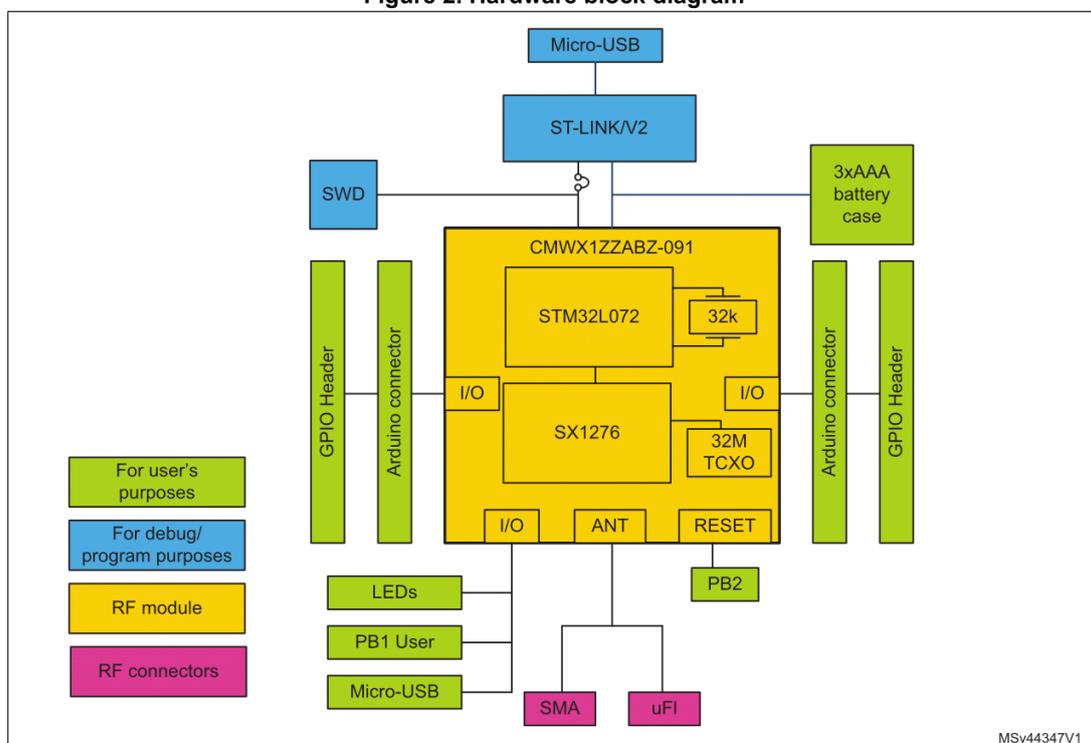
8 Hardware layout and configuration

The B-L072Z-LRWAN1 Discovery kit has been designed around the Murata LoRa®/Sigfox™ module including the STM32L072CZ microcontroller in a 49-pin WLCSP package.

Figure 2 illustrates the connection between the Murata LoRa®/Sigfox™ module and the peripherals (ST-LINK/V2, RF Antenna, LEDs, push-buttons, USB 2.0 FS Micro-B connector, 3xAAA battery holder).

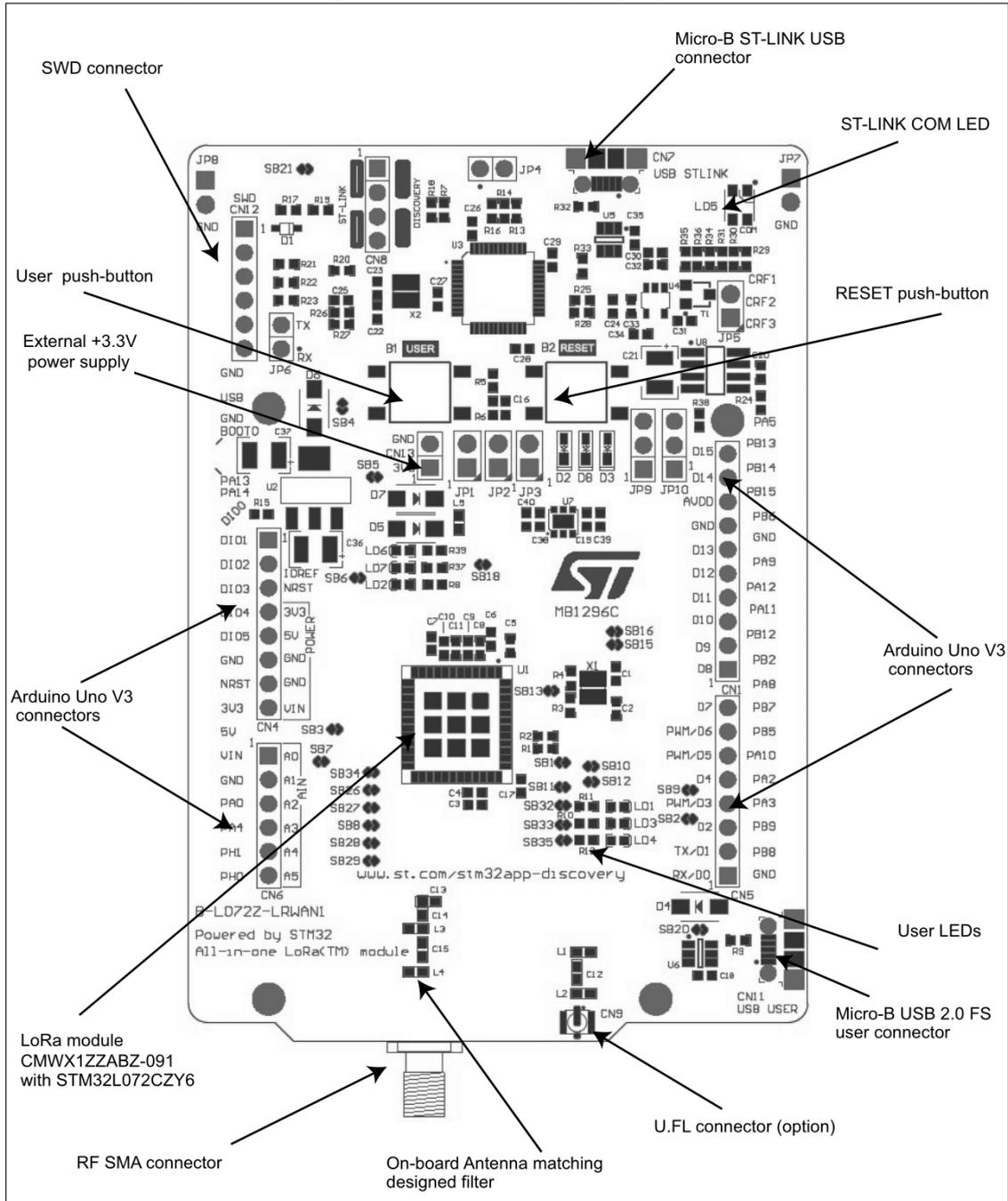
Figure 3 and *Figure 4* help users to locate these features on the STM32L072 Discovery kit.

Figure 2. Hardware block diagram



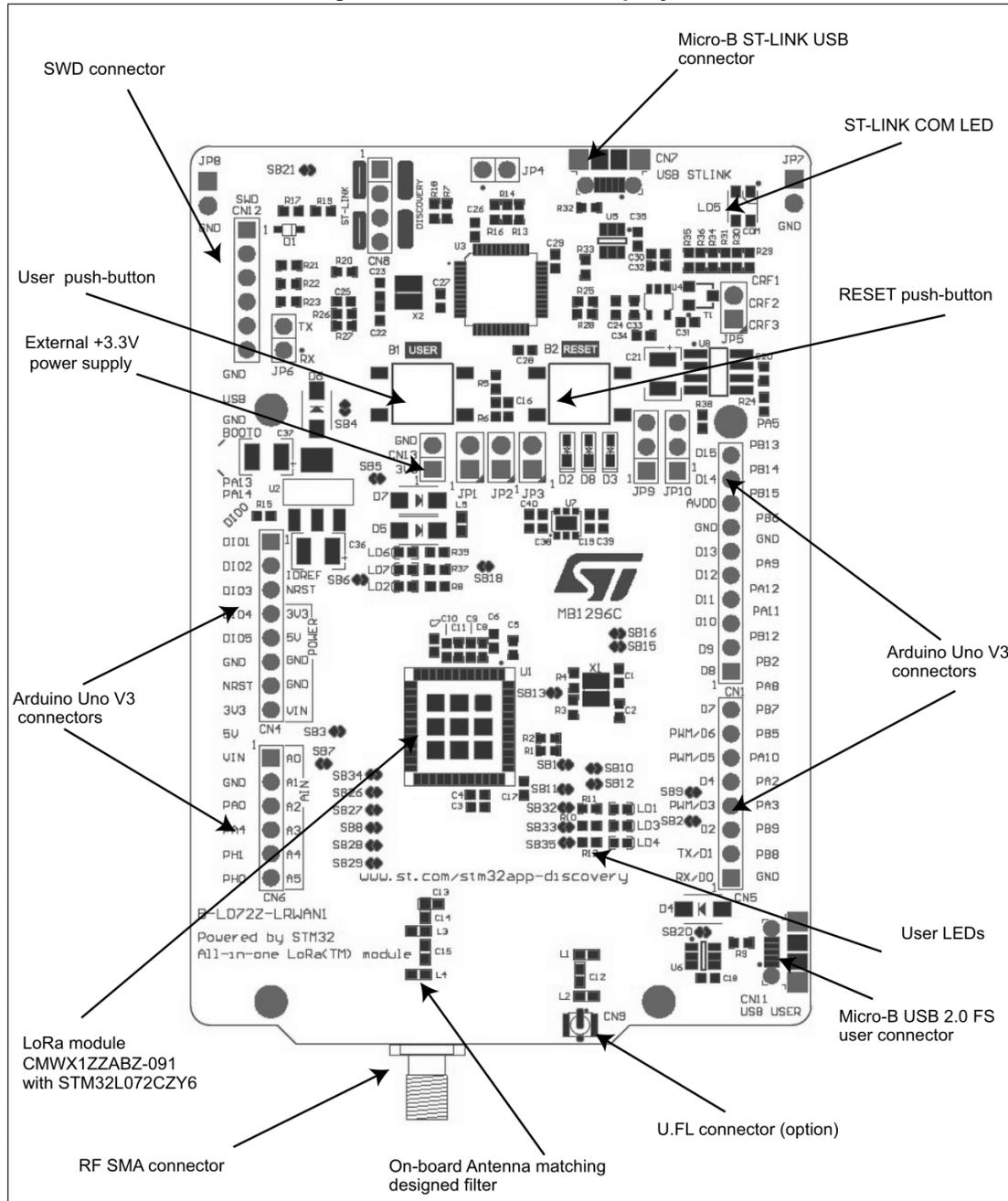
Anexo 2. Hojas de características

Figure 3. B-L072Z-LRWAN1 top layout



Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

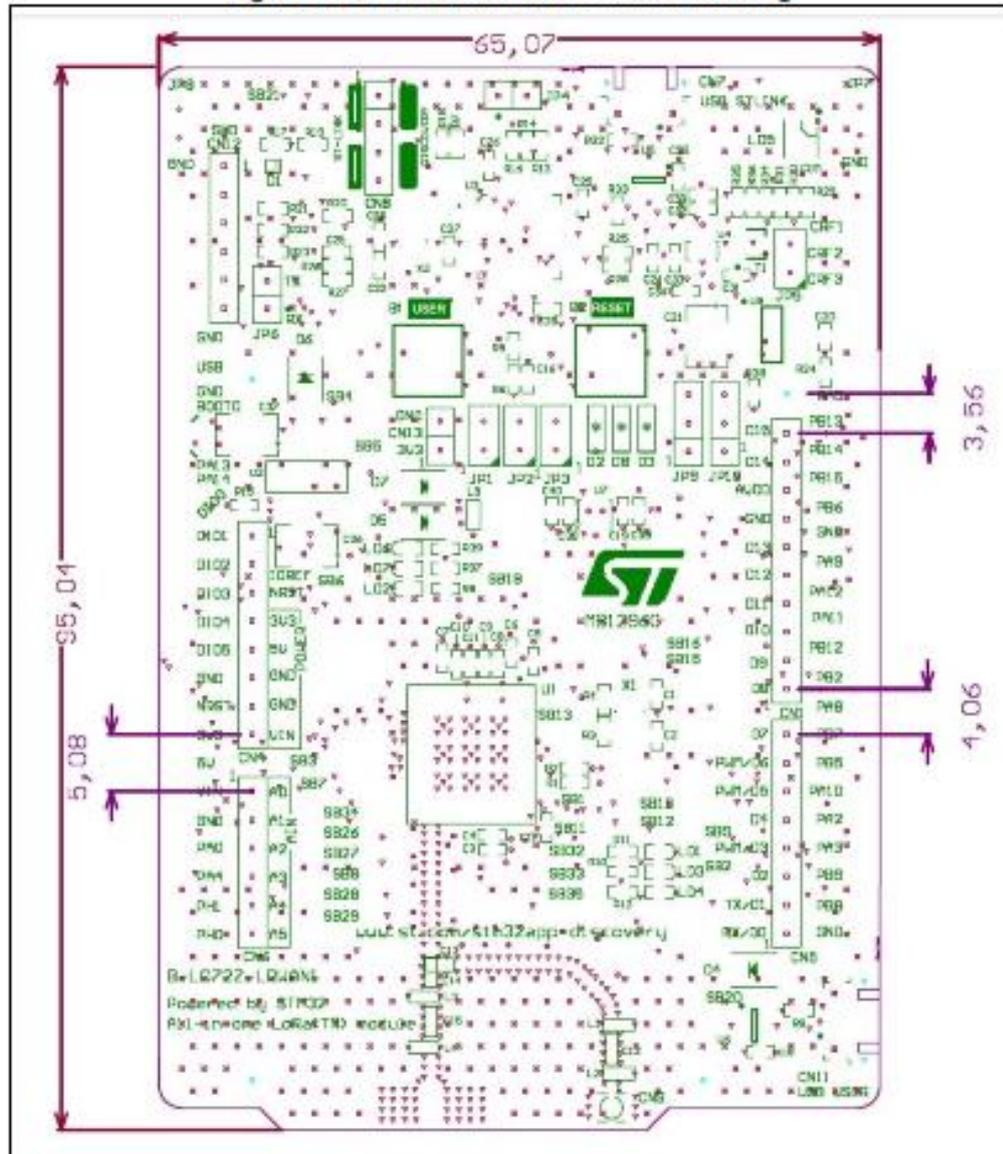
Figure 3. B-L072Z-LRWAN1 top layout



Anexo 2. Hojas de características

8.1 B-L072Z-LRWAN1 Discovery kit mechanical drawing

Figure 5. B-L072Z-LRWAN1 mechanical drawing



1. Plastic spacer height = 14 mm, overall height = 22 mm +/- 1 mm.
2. The overall dimensions of the B-L072Z-LRWAN1 is 106 x 65,07 mm including antenna and USB connectors.

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

Connectors

UM2115

9 Connectors

9.1 Arduino Uno V3 connectors

Table 5. Arduino Uno V3 connectors

Connector	Pin	Pin name	STM32 Pin	Function
CN1	10	D15	PB8	I2C1_SCL
	9	D14	PB9	I2C1_SDA
	8	AVDD	VREF+	VREF+
	7	GND	GND	Ground
	6	D13	PA5 or PB13	SPI1_SCK or SPI2_SCK
	5	D12	PB14	SPI2_MISO
	4	D11	PB15	SPI2_MOSI
	3	D10	PB6	LPTIM1_ETR
	2	D9	PB12	SPI2_NSS
	1	D8	PA9	USART1_TX
CN4	1	NC	-	-
	2	IOREF	-	+3.3 V Ref
	3	RESET	NRST	MCU_nRST
	4	+3.3 V	-	+3.3 V input/output
	5	+5 V	-	5 V output
	6	GND	-	Ground
	7	GND	-	Ground
	8	VIN	-	Power input
CN5	8	D7	PA8	MCO
	7	D6	PB2	LPTIM1_OUT
	6	D5	PB7	LPTIM1_IN2
	5	D4	PB5	LPTIM1_IN1
	4	D3	PB13 or NC	TIM21_CH1 or NC
	3	D2	PA10	USART1_RX
	2	D1	PA2	USART2_TX
	1	D0	PA3	USART2_RX

Anexo 2. Hojas de características

UM2115

Connectors

Table 5. Arduino Uno V3 connectors (continued)

Connector	Pin	Pin name	STM32 Pin	Function
CN6	1	A0	PA0	ADC_IN0
	2	A1	NC or PA0	NC or ADC_IN0
	3	A2	PA4	ADC_IN4
	4	A3	NC or PA4	NC or ADC_IN4
	5	A4	PH1 or PB9	OSC_IN or I2C1_SDA
	6	A5	PH0 or PB8	OSC_OUT or I2C1_SCL

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

UM2115

Connectors

Table 10. Description of the jumpers (continued)

Jumper	Pin number	Designation	Default state	Function
JP9	3	TCXO selection	2-3	Selection TCXO to VDD or external TCXO power
JP10	3	Reset source selection	1-2	Reset source selection between STSAFE or PA11

9.5 Configuration of the solder bridges

Refer to [Figure 8](#) and [Figure 9](#) to locate the solder bridges.

Table 11. Configuration of the solder bridges

Solder bridges	Designation	Default state	Function
SB19	Short VDD_USB_LRA	ON	Short VDD_USB_LRA connection
SB14	Short VDD_MCU_LRA	ON	Short VDD_MCU_LRA connection
SB17	Short VDD_RF_LRA	ON	Short VDD_RF_LRA connection
SB20	Short D4	OFF	D4 bypass
SB4	Short D6	OFF	D6 bypass
SB5	Short D7	OFF	D7 bypass
SB18	+5 V LED	ON	+5 V power supply ON
SB6	+3.3 V regulator output	ON	Used to disconnect internal +3.3 V regulator when external source applied on External 3.3 V pin
SB38, SB40, SB22, SB24	ST-LINK default	ON	Reserved
SB39, SB41, SB23, SB25	ST-LINK reserved	OFF	Reserved
SB37	ST-LINK RESET	ON	Connection between ST-LINK reset signal and LoRa [®] /Sigfox [™] module reset
SB36	ST-LINK MCO	OFF	Optional ST-LINK MCO redirected to LoRa [®] /Sigfox [™] module input clock OSC_IN
SB6	ST-LINK +5 V power	ON	Optional ST-LINK regulator disconnected from +5 V
SB21	ST-LINK force RESET	OFF	Reserved
SB26	PA5 to DIO4	OFF	Reserved to LoRa [®] /Sigfox [™] module debug
SB27	PA4 to DIO5	OFF	Reserved to LoRa [®] /Sigfox [™] module debug
SB28	ST-LINK TX	ON	Virtual COM port TX
SB29	ST-LINK RX	ON	Virtual COM port RX
SB15	LRA_USB_DP	OFF	Optional USB_DP connection
SB16	LRA_USB_DM	OFF	Optional USB_DM connection

Anexo 2. Hojas de características

Connectors

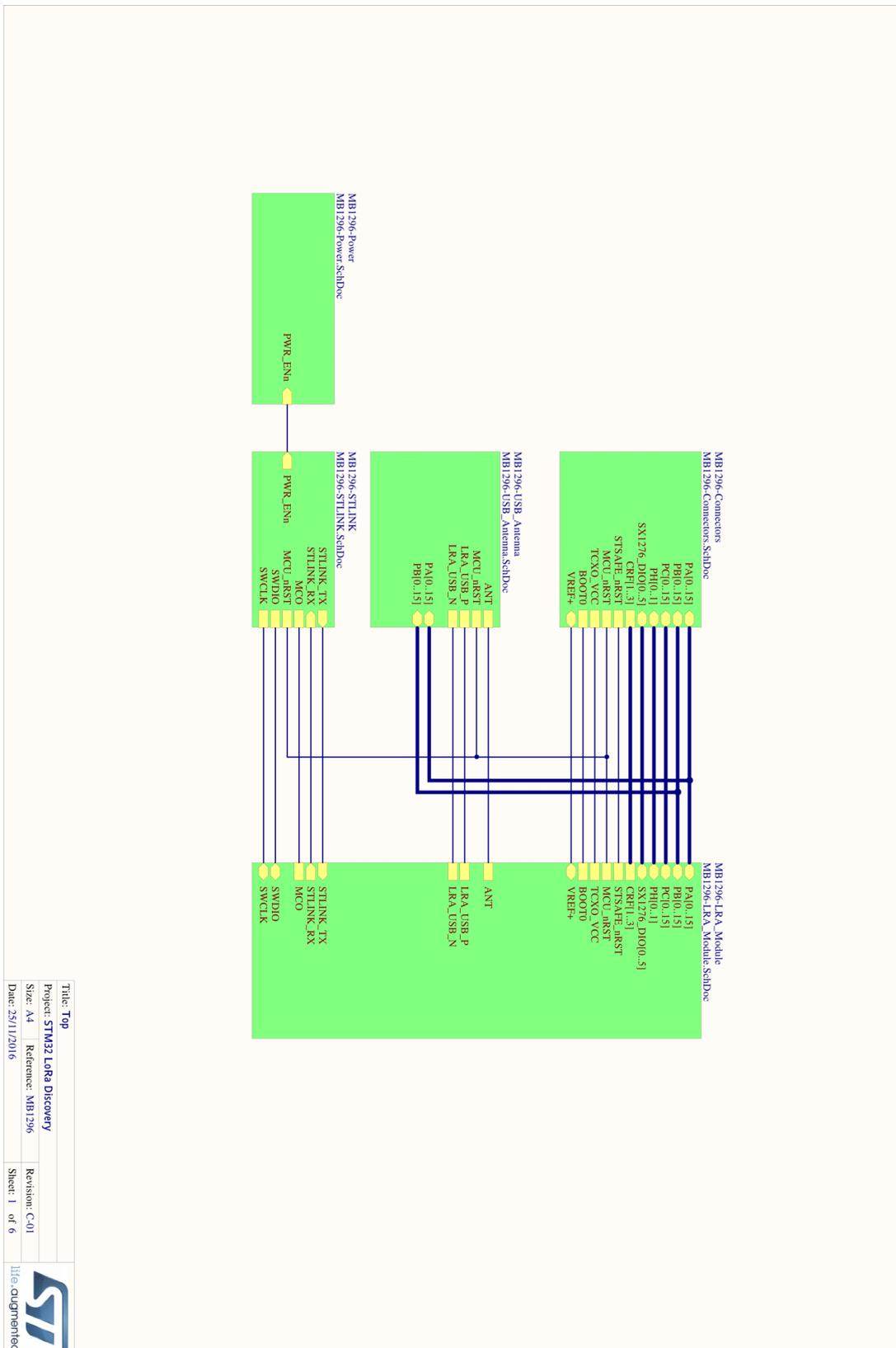
UM2115

Table 11. Configuration of the solder bridges (continued)

Solder bridges	Designation	Default state	Function
SB13	TCXO_OUT to OSC_IN	OFF	Allows connection of TXCO output to STM32L072CZY6 OSC_IN input
SB31	USER button PB2	ON	User push-button connected to PB2
SB30	USER button PA0	OFF	User push-button connected to PA0
SB32	LED LD1	ON	LD1 ON
SB33	LED LD3	ON	LD3 ON
SB35	LED LD4	ON	LD4 ON
SB34	LED LD2	ON	LD2 ON
SB10	PH1 Arduino	OFF	Connection A5(CN6) Arduino to PH1
SB3	PA5 Arduino	ON	Connection D13(CN1) Arduino to PA5
SB9	PB13 Arduino	ON	Connection D3(CN5) Arduino to PB13
SB2	PB13 Arduino	OFF	Connection D13(CN5) Arduino to PB13
SB7	PA0 alias Arduino	OFF	Connection A1(CN6) Arduino to PA0
SB8	PA4 alias Arduino	OFF	Connection A3(CN6) Arduino to PA4
SB11	PB9 Arduino	OFF	Connection A4(CN6) Arduino to PB9
SB12	PB8 Arduino	OFF	Connection A54(CN6) Arduino to PB8
SB1	PH0 Arduino	OFF	Connection A4(CN6) Arduino to PH0

Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

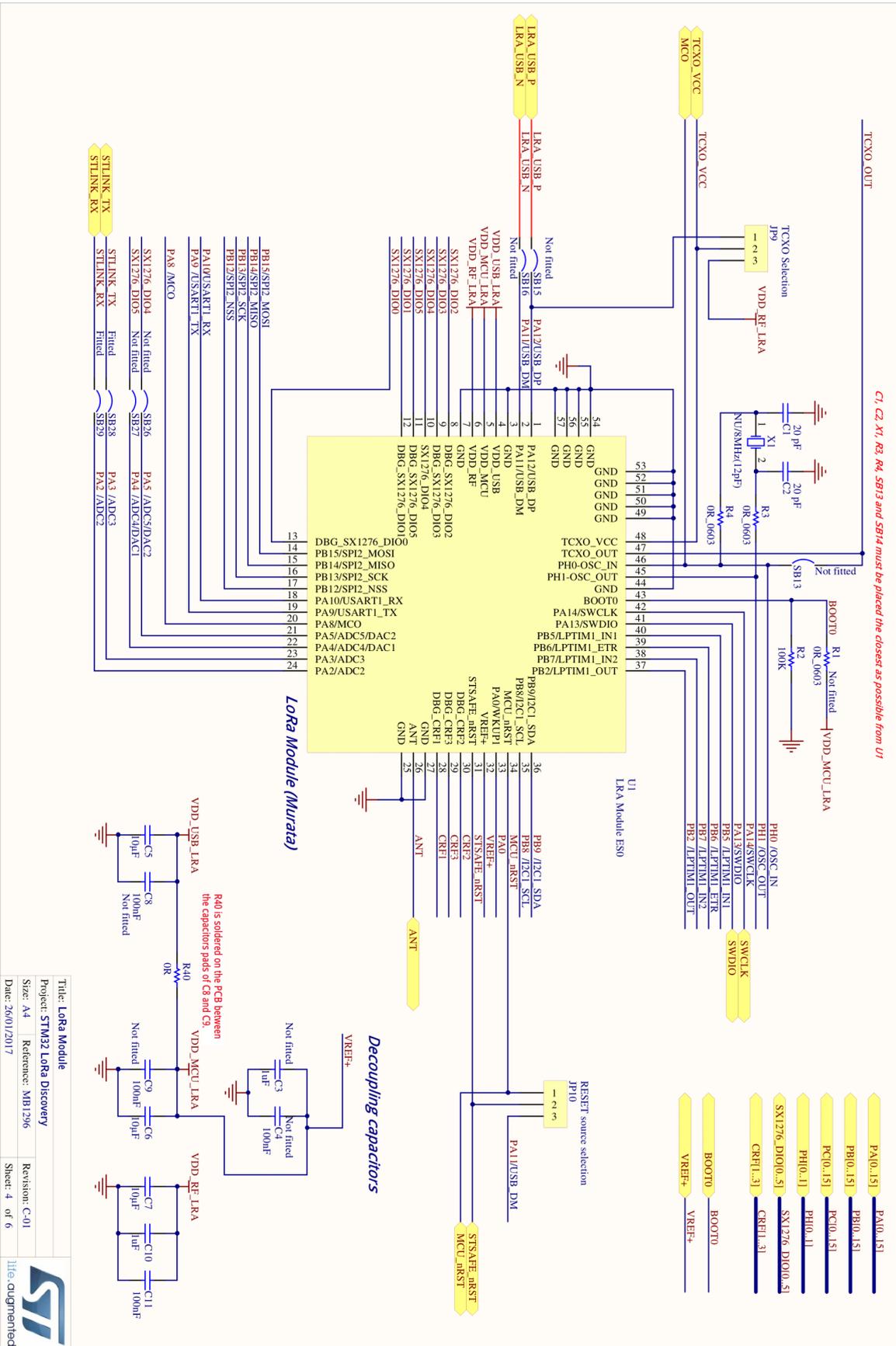
A2.1. Esquema eléctrico B-L072Z-LRWAN1 Discovery LoRa



Title: Top	
Project: STM32 LoRa Discovery	Revision: C-01
Size: A4	Reference: MBI1296
Date: 25/11/2016	Sheet: 1 of 6



Anexo 2. Hojas de características



Title:	LoRa Module
Project:	STM32 LoRa Discovery
Size:	A4
Reference:	MJD1296
Date:	26/01/2017
Revision:	C-01
Sheet:	4 of 6



Desarrollo de un nodo inalámbrico de bajo consumo basado en técnicas de IoT para monitorización de patrimonio cultural

