



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA



**UNIVERSITAT POLITÈCNICA DE VALÈNCIA**

**Escuela Técnica Superior de Ingeniería del Diseño**

---

**SISTEMA DE LUCES ANTICOLISIÓN  
INDEPENDIENTE PARA RPAS MEDIANTE  
MÓDULO INALÁMBRICO ESP12E**

**TRABAJO FINAL DEL**

**Grado en Ingeniería Aeroespacial**

**REALIZADO POR**

**Guillermo Luengo Arrué**

**TUTORIZADO POR**

**Rafael Masot Peris**

**Miguel Alcañiz Fillol**

**CURSO ACADÉMICO: 2019/2020**



# Agradecimientos

*A los tutores, por ser docentes que han sabido despertar interés y motivación en sus clases.*

*A Pablo, por ser un gran compañero en cada km de esta carrera.*

*A Silvia, por estar ahí siempre.*

*A mis padres, por todo.*

# Resumen

La normativa de seguridad establecida en el Real Decreto 1036/2017, exige a las aeronaves no tripuladas que son operadas mediante control remoto con una envergadura menor de 3m, que estén dotadas de un sistema de luces anticolidión. Las recomendaciones tanto de la AESA (Agencia Estatal de Seguridad Aérea) como de la OACI (Organización de Aviación Civil Internacional) para obstáculos móviles, establecen que dichas luces deben ser amarillas, su frecuencia de parpadeo debe estar entre 60 y 90 ciclos por minuto y su intensidad lumínica efectiva superior a 17 cd.

En el presente trabajo de fin de grado se ha llevado a cabo el diseño y fabricación de un sistema de luces anticolidión compacto, inalámbrico e independiente que puede ser instalado y removido del dron de forma rápida y sencilla obteniéndose así un producto que es capaz de satisfacer las exigencias determinadas por la normativa.

El sistema se compone de cinco módulos basados en el dispositivo de comunicación inalámbrica ESP12E, cada uno de los cuales controla un LED de alta luminosidad. El módulo central envía una señal al resto de módulos para controlar de forma sincronizada el encendido de los LEDs. Se ha desarrollado un código en el entorno Arduido IDE que permite la comunicación entre los módulos y la configuración de la frecuencia de parpadeo de las luces. Se ha diseñado una placa de circuito impreso (PCB) que incorpora el módulo de comunicación ESP12E, el LED y su driver de potencia y el circuito de control de carga/descarga de la batería. También se ha diseñado, mediante software CAD, una caja para alojar todos los componentes.

# Resum

La normativa de seguretat establerta en el Reial Decret 1036/2017, exigeix a les aeronaus no tripulades que són operades mitjançant control remot amb una envergadura menor de 3m, que estiguen dotades d'un sistema de llums anticoll·lisió. Les recomanacions tant de l'AESA (Agència Estatal de Seguretat Aèria) com de l'OACI (Organització d'Aviació Civil Internacional) per a obstacles mòbils, indiquen que aquestes llums han de ser grogues, la seva freqüència de parpelleig ha d'estar entre 60 i 90 cicles per minut i la seua intensitat lumínica efectiva superior a 17 cd.

En el present treball de fi de grau s'ha portat a terme el disseny i la fabricació d'un sistema de llums anticoll·lisió compacte, sense fil i independent que pot ser instal·lat i remogut del dron de manera ràpida i senzilla obtenint-se així un producte que és capaç de satisfer les exigències determinades per la normativa.

El sistema es compon de cinc mòduls basats en el dispositiu de comunicació sense fil ESP12E, cadascun d'aquests controla un LED d'alta lluminositat. El mòdul central envia un senyal a la resta de mòduls per a controlar de forma sincronitzada l'encesa dels LEDs. S'ha desenvolupat un codi en l'entorn Arduido IDE que permet la comunicació entre els mòduls i la configuració de la freqüència de parpelleig de les llums. S'ha dissenyat una placa de circuit imprès (PCB) que incorpora el mòdul de comunicació ESP12E, el LED i el seu driver de potència i el circuit de control de càrrega/descarrega de la bateria. També s'ha dissenyat, mitjançant programari CAD, una caixa per a allotjar tots els components.

# Abstract

According to the safety regulations established by the Real Decreto 1036/2017, all the unmanned aerial vehicles that are remote controlled and have a span lower than 3m, must be equipped with a lights anti-collision system. In this way, the recommendations issued by AESA (Agencia Estatal de Seguridad Aérea) and ICAO (International Civil Aviation Organization) for moving obstacles, point out the need for those lights to be yellow, with a blinking frequency between 60 and 90 cycles per minute and whose effective intensity must be greater than 17 cd.

In this Final Degree Project, the process from the design to the manufacturing of a compact, wireless and independent lights anti-collision system has been carried out. Furthermore, it can be installed and removed from the drone in a fast and easy way, obtaining a product that is able to cover all the regulations determined by the authorities.

The system itself is formed by five modules based on the wireless communication ESP12E, each of them controlling one high brightness LED. The central module sends a signal to the others so that the activation of the LEDs can be controlled in a synchronized manner. A computer code has also been developed in the Arduino's 'Integrated Development Environment' (IDE) that allows the communication between the modules and sets the blinking frequency of the lights. Also, a Printed Circuit Board (PCB) has been designed. This one includes the communication module ESP12E, the LED, its power driver and the charge and discharge control circuit of the battery. Finally, a box has been created by means of CAD software and later 3D printing to gather all the components correctly.

# Índice general

<b>Agradecimientos</b>	<b>II</b>
<b>Resúmenes</b>	<b>III</b>
<b>Índice de figuras</b>	<b>IX</b>
<b>Índice de tablas</b>	<b>X</b>
<b>Índice de códigos</b>	<b>XI</b>
<b>Nomenclatura</b>	<b>XII</b>
<b>1. Introducción</b>	<b>1</b>
1.1. Justificación . . . . .	3
1.2. Objetivos . . . . .	4
1.3. Estado del Arte . . . . .	5
<b>2. Diseño</b>	<b>7</b>
2.1. Diagrama de bloques . . . . .	8
2.1.1. Construcción del sistema . . . . .	8
2.1.2. Esquemático . . . . .	9
2.1.3. Implementación en el dron . . . . .	10
2.2. Hardware . . . . .	11
2.2.1. Alimentación y carga . . . . .	11
2.2.2. ESP12E y LED . . . . .	17
2.3. PCB . . . . .	24
2.4. Protocolos de Comunicación . . . . .	27
2.4.1. Comunicación Asíncrona UART . . . . .	27

2.4.2. Protocolo UDP . . . . .	28
2.5. Software . . . . .	30
2.5.1. Arduino IDE . . . . .	30
2.5.2. Código del punto de acceso . . . . .	31
2.5.3. Código de las estaciones . . . . .	34
2.6. Diseño de la caja . . . . .	38
2.6.1. Tapa superior . . . . .	40
2.6.2. Caja contenedora inferior . . . . .	43
<b>3. Validación</b>	<b>46</b>
3.1. Estructura . . . . .	47
3.2. Visibilidad . . . . .	49
3.2.1. Distancia: 242m . . . . .	49
3.2.2. Distancia: 1km . . . . .	50
<b>4. Conclusión</b>	<b>52</b>
<b>5. Presupuestos</b>	<b>53</b>
5.1. Materiales . . . . .	54
5.2. Mano de obra . . . . .	58
5.3. Coste total . . . . .	59
<b>A. PCB V0</b>	<b>60</b>
<b>B. Fichas de datos técnicos</b>	<b>61</b>
<b>C. Esquemático</b>	<b>62</b>
<b>Bibliografía</b>	<b>63</b>

# Índice de figuras

1.1. Sistema de luz anticollisión ofrecido por Lumecube [3] . . . . .	5
1.2. Sistema de luz anticollisión ofrecido por Litra [4] . . . . .	6
1.3. Sistema de luz anticollisión ofrecido por Ulanzi [5] . . . . .	6
2.1. Diagrama del sistema diseñado . . . . .	8
2.2. Diagrama del esquemático encargado de la carga y alimentación . . . . .	9
2.3. Diagrama del esquemático correspondiente al ESP12E . . . . .	9
2.4. Diagrama de implantación de los módulos . . . . .	10
2.5. Fotografía del CI TP4056 . . . . .	12
2.6. Sección del esquemático perteneciente al TP4056 . . . . .	12
2.7. Fotografía del CI FS312F-G . . . . .	13
2.8. Sección del esquemático perteneciente al FS312F-G . . . . .	13
2.9. Fotografía del CI MT3608 . . . . .	14
2.10. Sección del esquemático perteneciente al MT3608 . . . . .	14
2.11. Fotografía del CI AMS1117 . . . . .	15
2.12. Sección del esquemático perteneciente al AMS1117 . . . . .	15
2.13. Curva de descarga típica de batería LiPo . . . . .	16
2.14. Fotografía de la batería LiPo . . . . .	17
2.15. Pinout del módulo ESP12E . . . . .	18
2.16. Sección del esquemático perteneciente al ESP12E . . . . .	19
2.17. Sección del esquemático perteneciente al ADC . . . . .	20
2.18. Sección del esquemático perteneciente al los leds del estado de la batería . . . . .	20
2.19. MOSFET canal N . . . . .	21
2.20. MOSFET AO3400 y curva característica . . . . .	21
2.21. LED Golden Dragon de OSRAM . . . . .	22

2.22. Sección del esquemático perteneciente al LED de alta intensidad . . . . .	23
2.23. Imagen de la PCB (diseño) . . . . .	25
2.24. Imagen de la PCB . . . . .	26
2.25. Comunicación UART asíncrona . . . . .	27
2.26. Cable adaptador comunicación USB a UART . . . . .	28
2.27. Impresora M3D Micro . . . . .	38
2.28. Paso 1 del diseño de la tapa superior . . . . .	40
2.29. Agujeros tapa superior . . . . .	40
2.30. Pieza de metacrilato superior . . . . .	41
2.31. Sketch 2n parte . . . . .	41
2.32. Paso 1 del diseño de la tapa superior . . . . .	42
2.33. Pieza de metacrilato inferior . . . . .	42
2.34. Modelo de la PCB con el interruptor . . . . .	43
2.35. Bloque inferior . . . . .	43
2.36. Bloque inferior hueco . . . . .	44
2.37. Bloque inferior hueco . . . . .	44
2.38. Bloque final . . . . .	45
2.39. Caja impresa en 3D . . . . .	45
3.1. Módulo ensamblado . . . . .	47
3.2. Módulo montado en dron . . . . .	48
3.3. Visibilidad a 242m . . . . .	50
3.4. Visibilidad a 1km . . . . .	51
A.1. Imagen de la PCB descartada (diseño) . . . . .	60
A.2. Imagen de la PCB descartada . . . . .	60

# Índice de tablas

2.1. Lista de componentes del sistema . . . . .	11
2.2. Configuración de los pines para los tipos de arranque . . . . .	18
2.3. Listado de pines utilizados y su uso . . . . .	19
2.4. Modos de funcionamiento de los módulos del sistema . . . . .	30
3.1. Situación de los puntos . . . . .	49
3.2. Situación de los puntos . . . . .	50
5.1. Coste total en materiales . . . . .	54
5.2. Coste en personal . . . . .	58
5.3. Coste total . . . . .	59

# Índice de códigos

2.1. Código del módulo configurado como punto de acceso . . . . .	31
2.2. Código del módulo configurado como estación . . . . .	34

# Nomenclatura

## Magnitudes físicas

$\Omega$  Ohmio, unidad de resistencia eléctrica en el sistema internacional

$I$  Intensidad de corriente eléctrica [A]

$V$  Tensión, cuantifica la diferencia de potencial eléctrico entre dos puntos [V]

## Otros símbolos

*ADC* Conversor Analógico a Digital

$C$  Tasa de carga/descarga segura para una batería [A]

*CAD* Diseño Asistido por Computadora

*LED* Diodo emisor de luz

*LiPo* Designación común para las baterías de polímero de iones de litio

*PCB* Placa de circuito impreso

*SoC* System on Chip

*UART* Receptor y Transmisor Asíncrono Universal

*UDP* Protocolo de Datagramas de Usuario



# 1. Introducción

Es objetivo de este Trabajo de Fin de Grado, la realización de un sistema de luces anticolidión que cumpla con todos los requisitos de la normativa para ser aplicado de forma segura. Además, dicho sistema tiene independencia total del resto de sistemas del dron así como un diseño cuya principal meta es la facilidad de instalación y modularidad, pudiéndose modificar con facilidad el número de luces instaladas en el dron.

Satisfacer las condiciones de la normativa para poder volar un dron de noche habilita a los usuarios a realizar actividades con el mismo que no pueden ser cubiertas en el transcurso del día. Existen servicios de carácter público, como pueden ser emergencias o policías, para los que la operatividad nocturna resulta indispensable o crucial.

Por ejemplo, un dron puede ser operado por bomberos para, durante un incendio, sobrevolar la zona afectada facilitando así una visión general del estado de la situación sin la necesidad de que una aeronave tripulada, como un helicóptero tenga que exponerse al peligro directo que supone sobrevolar las llamas. Equipos de rescate pueden hacer uso también de un dron en condiciones de nocturnidad para encontrar a las víctimas e incluso para una vez localizada la posición realizar un reconocimiento de la zona para evaluar la ruta más favorable que permita un acceso seguro. Estos son solo un pequeño extracto de las posibilidades que brinda un dron a los trabajadores de emergencias y la importancia de no ver limitada su operatividad con respecto a la nocturnidad.

No obstante, también existen otras actividades más centradas en ocio, cultura o ámbitos privados que pueden aprovechar las capacidades que ofrece un dron. En grabaciones de películas/series, donde la acción tenga lugar en franjas horarias nocturnas, el uso de un dron en lugar de un helicóptero, para obtener escenas mediante grabación aérea, resulta muy ventajoso ya que se tiene una mayor disponibilidad y menores costes directos e indirectos. También existe una tendencia al alza de empresas de fotografía que los utilizan en eventos nocturnos como pueden ser reportajes de bodas, fiestas particulares, etc. Tampoco hay que olvidar a las personas que quieren realizar vuelos nocturnos con su dron simplemente por diversión y respetando las condiciones de seguridad.

Se procede, a continuación, a presentar como se ha estructurado el trabajo. Para facilitar la comprensión del mismo se ha seguido una estructura lo más lógica posible evitando, de este modo, que el lector se pierda a lo largo de la lectura del mismo.

Para ello, el trabajo se estructura utilizando capítulos, conformando unidades temáticas por sí mismos. A su vez, éstos se encuentran divididos en secciones y subsecciones cuando se ha considerado necesario para facilitar el entendimiento de los apartados.

En el *Capítulo 1* se aborda la parte introductoria del trabajo. Se expone el cometido del mismo así como su justificación. Se exponen, también, los objetivos que se han impuesto como meta y la situación del mercado actual con lo que respecta a compañías que buscan satisfacer, a través de sus productos, las mismas necesidades que se cubren con el sistema desarrollado en el presente trabajo.

Tras esta parte introductoria, en el *Capítulo 2* se presenta el diseño del sistema. Este capítulo representa el grueso del trabajo y, en él, se detallan los pasos seguidos en la realización del proyecto. Este capítulo, debido a su extensión, se ha dividido en secciones en las que se abordan distintos pasos en el proceso del diseño. Primero, mediante el uso de diagramas de bloques se introduce el diseño y su aplicación. Posteriormente se presentan los componentes utilizados, el diseño de la placa de circuito impreso, los protocolos de comunicación que intervienen en el sistema, el código que utilizan los módulos y, por último, la caja donde se aloja la PCB y la batería.

El siguiente paso, la validación del sistema, se lleva a cabo en el *Capítulo 3*. En él, se comprueba el funcionamiento y la adecuación de los elementos diseñados. También se comprueba la distancia desde la que es posible distinguir la luz del dispositivo.

En el *Capítulo 4* se exponen las conclusiones alcanzadas en la realización del presente Trabajo Final de Grado.

Finalmente, en el *Capítulo 5* se muestra un desglose aproximado del presupuesto que se debería de disponer para la realización del presente trabajo.

## 1.1. Justificación

La buena visibilidad de un RPAS (*Remotely Piloted Aircraft System*) durante su operación en horas de poca visibilidad es un problema de seguridad de una importancia muy relevante y, derivado del auge de este tipo de aeronaves, el Gobierno de España, mediante un Real Decreto en 2017 [ver 1, Art. 25] dictamina, entre otros, los requisitos que debe de cumplir un RPAS durante un vuelo nocturno. Concretizando más en este aspecto, la Agencia Estatal de Seguridad Aérea en [ver 2, Punto 5], detalla de una forma más clara las exigencias descritas por el Real Decreto.

En la actualidad, no todos los drones a la venta llevan este sistema integrado. Así mismo, tanto drones de construcción propia como drones más antiguos carecen, de partida, de este sistema. Por otra parte, desde el Dpto. de Ingeniería Cartográfica, Geodesia y Fotogrametría de la UPV se informó a los tutores de este trabajo acerca de que, habiendo adquirido soluciones en el mercado y no obteniendo resultados satisfactorios, tenían la necesidad de un sistema de luces anticolidión capaz de funcionar de forma independiente con su propia batería.

Como se ha expuesto, existe una necesidad a cubrir tanto desde el mercado general como dentro de la propia universidad fruto de la cual surge el presente Trabajo de Fin de Grado con el que se satisface el requerimiento normativo del modo más eficiente posible en términos tanto temporales como económicos sin mermar la calidad ni la fiabilidad del producto demandado.

Así pues, los principales requisitos a cubrir por el sistema son: la posibilidad de comunicarse de forma inalámbrica, un bajo consumo que haga viable la independencia energética del sistema con respecto a la alimentación del drone, un tamaño reducido que no tenga un gran impacto sobre la aerodinámica, aspecto ni funcionamiento del drone.

El primer problema que surge al inicio del diseño viene dado por la necesidad de comunicación de forma inalámbrica que el sistema va a necesitar para la coordinación de las luces anticolidión, es por esto que la parte central de todo el sistema es el módulo ESP12E, dicho módulo está basado en el SoC ESP8266 y permite, sin necesidad de añadir periféricos, la conexión inalámbrica a través de redes wifi. Dicha comunicación se lleva a cabo mediante el uso de una red wifi creada por uno de los módulos que forma el sistema, funcionando como punto de acceso, a dicha red se conectan el resto de módulos que funcionan en modo estación. De esta manera, el módulo que crea el punto de acceso envía mediante protocolo UDP la señal que el resto de estaciones reciben y es usada para activar las luces de forma síncrona.

Por otro lado, para garantizar la independencia del sistema de alimentación de drone, cada módulo que forma el sistema de luces anticolidión tiene su propia batería con su sistema de protección de carga y de descarga así como la circuitería necesaria para administrar el proceso de carga. Todo esto viene integrado en una misma PCB diseñada para alojar todos los circuitos así como para ser contenida en una caja creada para facilitar la instalación del sistema en los brazos del drone.

## 1.2. Objetivos

En este Trabajo Final de Grado se lleva a cabo la fabricación de un sistema de luces anticolidión fiable, modular e independiente del drone en el que se instala. De este modo, los objetivos que determinan los pasos seguidos en la ejecución de este trabajo son los siguientes:

- Crear el código mediante el software Arduino IDE tanto para el módulo que va a funcionar como punto de acceso como para los que van a funcionar como estación
- Diseñar el circuito de carga para la batería así como el circuito de protección de descarga y sobrecarga
- Diseñar el circuito de control del LED
- Implementar una señal de control para comprobar el estado de carga de la batería

### 1.3. Estado del Arte

Desde que en 2017 se hiciera obligado el uso de luces anticolidión han aparecido empresas que ofrecen diferentes productos para satisfacer esta necesidad. En este trabajo final de grado se ofrece un sistema para cumplir con la normativa en vigor diferente a lo ofertado en el mercado por diferentes empresas privadas. Como se ha comentado previamente, los módulos creados en este trabajo son independientes del dron y capaces de comunicarse entre sí de forma inalámbrica además de informar del estado de la batería y de ser, para un usuario medio, sencillos de reprogramar.

Como parte de la justificación de este trabajo se presenta aquí un análisis del estado del arte actual en lo referente al tratamiento, por parte de empresas en el mercado, de esta imposición de requerimientos dada por la normativa.

Realizando una búsqueda en el mercado a través de internet se encuentran algunas de las soluciones más populares en la actualidad para este tipo de sistemas que se exponen a continuación. <https://uavcoach.com/drone-anti-collision-lights/>

Como primer ejemplo se tiene a la compañía Lumecube que presenta, con un precio de 47€, un diseño pequeño con ópticas intercambiables que permiten emitir luces de distintos colores. Entre otras características, enuncian un peso de 10 gramos, una duración de batería de 6 horas y diferentes frecuencias de parpadeo de la luz. Este dispositivo se sujeta al dron mediante velcros.



Figura 1.1: Sistema de luz anticolidión ofrecido por Lumecube [3]

Por otra parte, la compañía Litra ofrece su módulo por 90€. En sus especificaciones anuncian un rango entre 800 y 100 lumens, un haz de 5700K y una batería recargable mediante conexión Micro-Usb sin citar horas de autonomía. Las sujeciones para el dron se venden de forma separada a un precio de 35€ y los módulos no tienen ningún tipo de conexión inalámbrica.



(a) Luz anticollisión

(b) Correas de sujeción

Figura 1.2: Sistema de luz anticollisión ofrecido por Litra [4]

Por otro lado, Ulanzi propone otro dispositivo, similar al ofrecido por Lumecube, por un precio de 60€. Como especificaciones de este modelo destacan 14 gramos de peso, 9 tipos diferentes de iluminación y hasta 20 horas de operación continua. Al igual que el dispositivo de Lumecube, este se sujeta al dron mediante velcros.



Figura 1.3: Sistema de luz anticollisión ofrecido por Ulanzi [5]

Como puede verse, los sistemas más populares que se encuentran en el mercado son muy semejantes, los principales inconvenientes que se pueden destacar es el carácter individual de los módulos, la poca seguridad de sus anclajes así como destacar el elevado precio que no se ve reflejado en los componentes que utilizan.

## 2. Diseño

En esta sección del trabajo pasa a exponerse de forma de tallada cada una de las partes del diseño del proyecto realizado. Para ello, el capítulo se subdivide en 3 secciones, cada una de las cuales se centra en diferentes partes específicas: software, hardware y diseño de la caja.

Para empezar, en la *Sección 2.1*, se muestra en forma de diagrama de bloques el ensamblaje del sistema así como la forma de implementarlo en un dron de manera introductoria.

En el segundo apartado, *Sección 2.2*, se exponen los elementos que conforman el conjunto electrónico del sistema, especificando su función, así como el diseño de la circuitería en detalle. Posteriormente, en la *Sección 2.3* se expone el diseño de la PCB. Sobre dicha placa de circuito impreso se ha llevado a cabo el montaje final del proyecto para su presentación.

En la *Sección 2.4* se pasa a dar una descripción de los protocolos de comunicación que participan, de algún modo, a lo largo del desarrollo del presente Trabajo Final de Grado.

Seguidamente, en la quinta sección, *Sección 2.5*, se presenta el código elaborado mediante Arduino IDE [6] tanto para los módulos que funcionan como estación como para el que funciona como punto de acceso. Nombrando las librerías utilizadas, clarificando su aplicación y describiendo de forma exhaustiva las funciones utilizadas en el código.

Tras la presentación del código y como parte final de este capítulo, en la *Sección 2.6* se describe en detalle la caja exterior y sus medidas, realizada mediante software CAD. En ella se aloja la PCB y sirve de soporte para anclar el sistema al dron para que se mantenga fijado de forma segura durante toda la operación del mismo.

## 2.1. Diagrama de bloques

En esta sección, haciendo uso de diagramas de bloques, se presenta el sistema de luces anticollisión desarrollado en el presente Trabajo Final de Grado. De esta manera, se busca realizar una introducción al contenido del *Capítulo 2* mediante una representación gráfica.

Para ello, se ha dividido la sección en tres partes. En la primera se expone el diseño realizado a lo largo del trabajo. En la segunda se presenta el diseño que sigue el esquemático del circuito. Por último, en la tercera, se presenta la forma más efectiva de implantar el sistema en un dron.

### 2.1.1. Construcción del sistema

En la *Figura 2.1* se muestra el diagrama correspondiente al diseño del sistema de luces anticollisión desarrollado en el presente Trabajo Final de Grado.

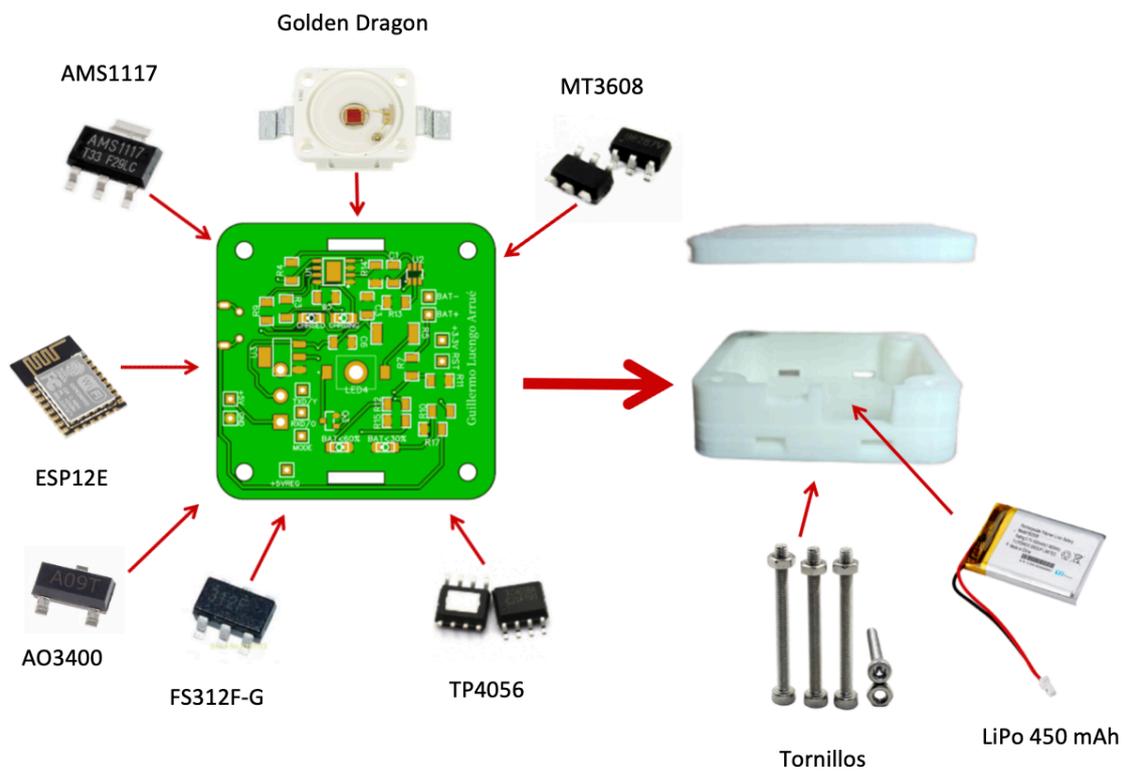


Figura 2.1: Diagrama del sistema diseñado

### 2.1.2. Esquemático

En esta subsección se muestra, en forma de bloques, el esquema seguido para diseñar el circuito del sistema.

En la *Figura 2.2* se muestra el diagrama que describe la parte del esquemático encargada de la alimentación y la carga del sistema.

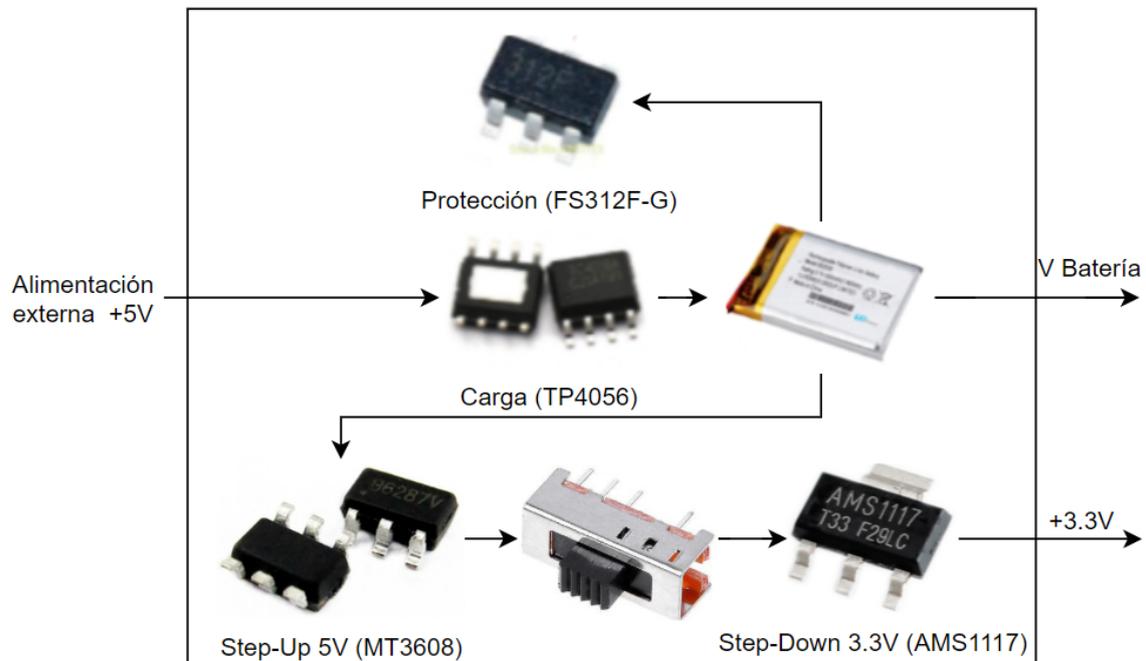


Figura 2.2: Diagrama del esquemático encargado de la carga y alimentación

En la *Figura 2.2* se muestra el diagrama que describe la parte del esquemático que cubre las conexiones del módulo ESP12E.

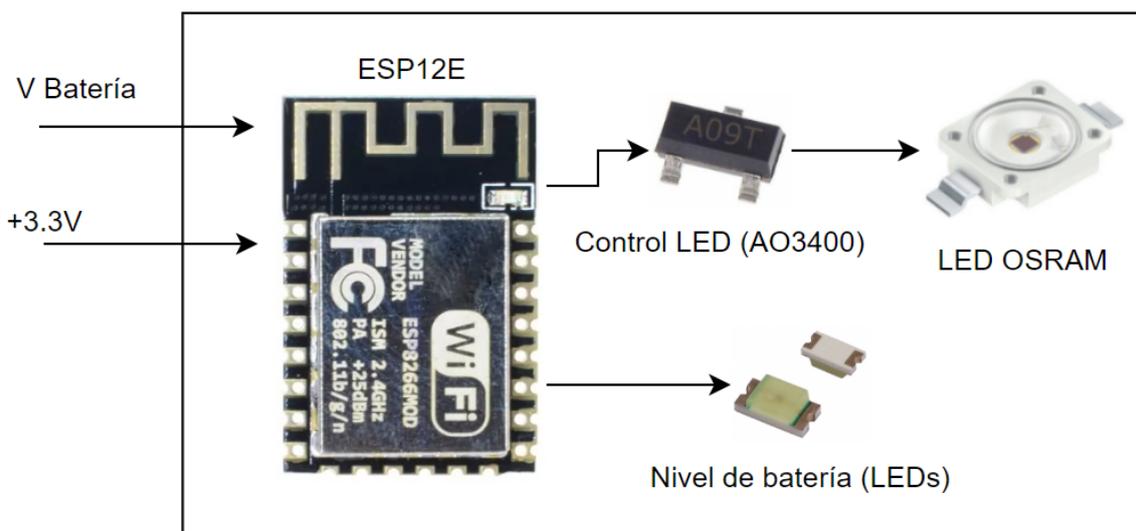


Figura 2.3: Diagrama del esquemático correspondiente al ESP12E

### 2.1.3. Implementación en el dron

En la *Figura 2.4* se muestra el diagrama correspondiente a la implementación del sistema de luces anticolisión en un dron cuadricóptero.



Figura 2.4: Diagrama de implantación de los módulos

## 2.2. Hardware

En este apartado se presentan cada uno de los elementos que forman parte del sistema de forma detallada seguidamente se muestran sus conexiones en el esquemático, este se encuentra completo en el *Apéndice C*.

Para empezar, de forma introductoria se muestra en la *Tabla 2.1* el listado de los componentes principales que conforman el sistema junto con una breve descripción.

Componente	Descripción
TP4056	Cargador lineal para baterías de 1 celda
FS312F-G	CI protección carga/descarga batería
FS8205	Mosfet dual canal N
MT3608	Convertidor de voltaje
AMS1117-3.3	Regulador de voltaje
ESP-12E	Módulo WiFi con procesador ESP8266
LY W5SM	LED golden DRAGON de OSRAM
AO3400	Mosfet canal N

Tabla 2.1: Lista de componentes del sistema

Los componentes del sistema van a ser presentados siguiendo un orden tal que facilite la comprensión del esquemático y los motivos de la inclusión de cada componente en el mismo. Con dicho fin, se va a seguir una ruta lo más lineal posible a través del esquemático del circuito; evitando, siempre que sea posible, viajar hacia atrás en el circuito. En primer lugar se presentan los componentes encargados de la alimentación y de la regulación de la tensión en el circuito, tras lo cual se expone la parte del módulo ESP12E así como del LED de alta potencia.

### 2.2.1. Alimentación y carga

El primer componente en introducirse es el TP4056 (*Figura 2.5*), cargador de batería, este componente permite realizar la carga de una batería LiPo de forma segura fijando la tensión de carga a 4.2V. Una de sus características más destacable es que permite programar la corriente de carga de la batería mediante el uso de una resistencia (*Ecuación 2.1*), este aspecto tiene una relevante importancia ya que no es conveniente cargar las baterías LiPo a más de 1C (*se detalla en su apartado*).

$$I_{BAT} = \frac{1200}{R_{PROG}} \quad (2.1)$$

Como puede verse en la *Figura 2.6*, la  $R_{PROG}$  es de  $3.3k\Omega$ . Por lo tanto la batería se carga con una intensidad de 0.36A. Por otro lado, el CI TP4056 dispone también de dos pines de estado utilizados para indicar que se ha terminado la carga y la presencia de tensión en la entrada.



Figura 2.5: Fotografía del CI TP4056

La alimentación externa se puede realizar por dos vías, puede utilizarse el pad de +5V conjuntamente con el pad de GND o puede utilizarse el conector micro-USB hembra, el esquemático de esta parte puede verse en la (Figura 2.6). No obstante, hay que mencionar que cuando el sistema está ensamblado dentro de la caja, la única posibilidad de alimentación externa es la de la entrada micro-USB puesto que los pads quedan ocultos. Para la realización del esquemático se ha seguido lo dispuesto en la aplicación típica dentro de la hoja de datos técnicos que se puede encontrar en el *Apéndice B*

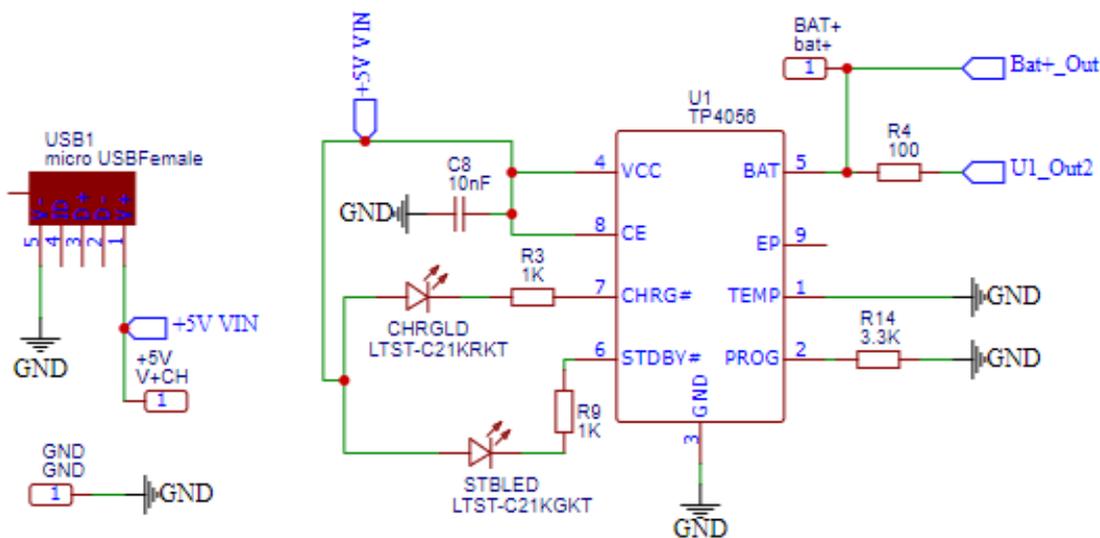


Figura 2.6: Sección del esquemático perteneciente al TP4056

En la parte derecha superior de la *Figura 2.6* se aprecian 2 puertos que llevan al circuito del convertidor de tensión MT3608 (*Bat+\_Out*) y al del protector de la batería FS312F-G (*U1\_Out2*), siendo este último el siguiente componente en el análisis.

El circuito integrado FS312F-G (*Figura 2.7*) está diseñado para proteger baterías LiPo de daños y degradación provocados por llevarlas a rangos de tensión superior o inferior a los de diseño, por lo tanto ofrece una seguridad añadida a la ofrecida por el fabricante de la batería en caso de que este haya incluido algún sistema de protección en ella. Como peculiaridad hay que señalar que la primera vez que se conecta una batería al un circuito que tiene implementado este CI para que este permita la descarga de la batería y con ello la operatividad del circuito, se debe conectar todo el sistema a un cargador o, en su defecto, puentear los pines CS y VSS del FS312F-G tal y como viene indicado en la hoja técnica el CI cuyo enlace se encuentra en el *Apéndice B*.



Figura 2.7: Fotografía del CI FS312F-G

En cuanto al circuito para la implementación de este CI se han seguido las indicaciones contenidas en su hoja de especificaciones y es el que puede verse en la *Figura 2.8*, como puede verse en esta parte del circuito se conecta el electrodo negativo de la batería también se hace uso del Mosfet dual que viene indicado en la *Tabla 2.1*.

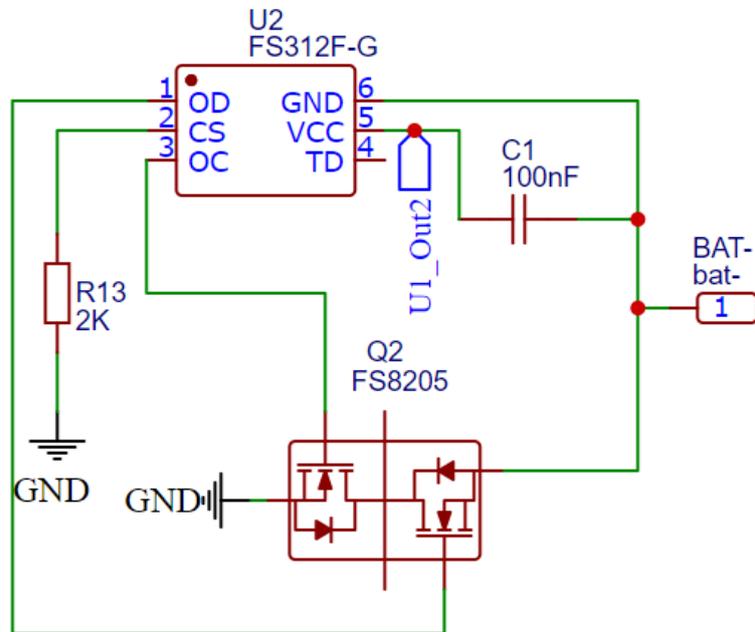


Figura 2.8: Sección del esquemático perteneciente al FS312F-G

Tomando ahora el enlace del puerto *Bat +\_Out* (*Figura 2.6*) se llega al circuito encargado de subir la tensión de la batería a 5V. De esto se encarga el CI MT3608, que puede verse en la *Figura 2.9*. Este circuito integrado es un step-up para aplicaciones de baja potencia que tiene un rango de tensión de entrada de 2V a 24V que es más de lo necesario para la fluctuación de la batería LiPo usada en este proyecto, que tiene una tensión de 4.2V cuando se encuentra totalmente cargada y de 3V cuando se considera descargada.

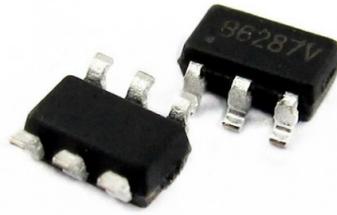


Figura 2.9: Fotografía del CI MT3608

Mediante los valores de R2 y de R8 se configura la tensión de salida tal y como viene indicado en su hoja de especificaciones cuyo enlace, de nuevo, se encuentra adjunto en el *Apéndice B*. En el caso del presente trabajo, las resistencias se han configurado de tal manera que la tensión a la salida sea de 5V, para el resto de los elementos que conforman esta parte del circuito se han seguido las recomendaciones de la hoja técnica.

Como puede verse en la *Figura 2.10*, la tensión de la batería es la que llega a este CI. Para poder realizar medidas con un multímetro, y así comprobar que todo funciona correctamente, se crea el pad +5VREG donde debe de medirse un salto de 5V con respecto de GND. También se dibuja, a la salida, el puerto +5VReg que sirve de nexo para la siguiente sección del circuito.

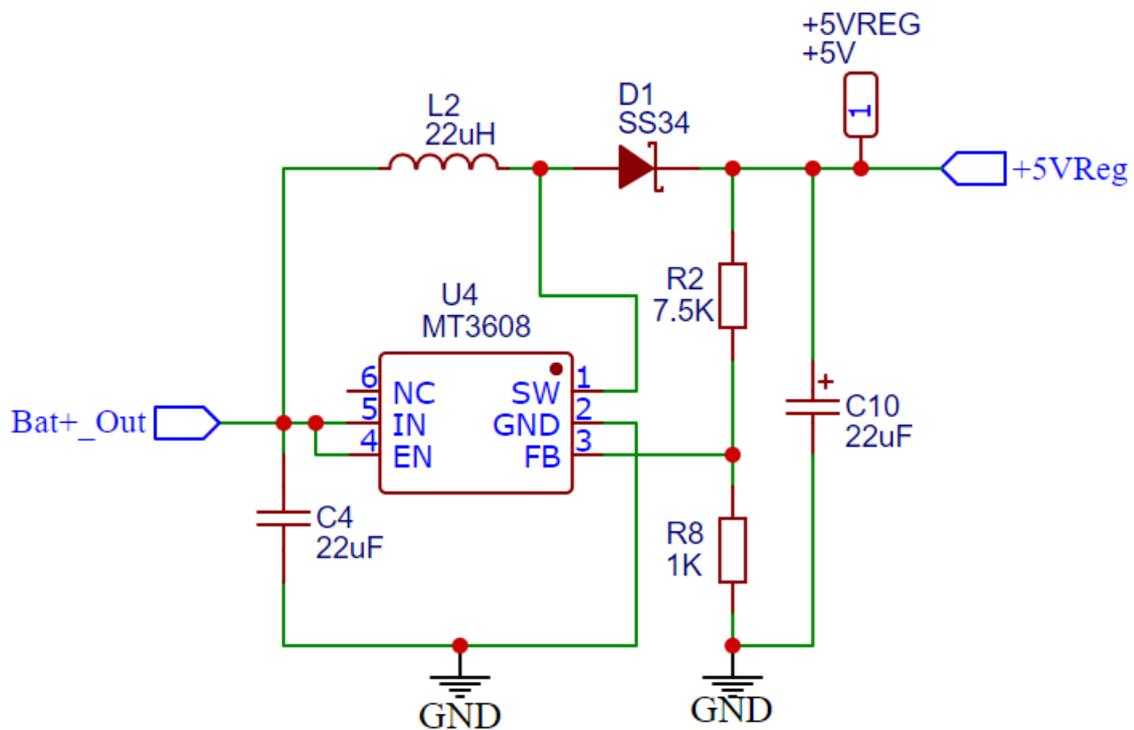


Figura 2.10: Sección del esquemático perteneciente al MT3608

En la siguiente parte del circuito y, como se ve en la *Figura 2.12* tras un interruptor, se encuentra el AMS1117-3.3 (*Figura 2.11*). Se trata de un regulador fijo de tensión que garantiza una entrega de corriente de hasta 1A y una tensión a la salida de 3.3V. Previamente en el circuito se ha subido la tensión ofrecida por la batería a 5V para, ahora, bajarla a 3.3V, esto se debe a que el ESP12E debe de ser alimentado a una tensión de 3.3V y la batería tiene un rango que va desde los 4.2V hasta los 3V. Por esto es necesario primero subir la tensión a la salida de la batería para luego poder administrar una tensión de 3.3V constante al ESP12E a lo largo de todo el rango de capacidad de la batería.



Figura 2.11: Fotografía del CI AMS1117

Como puede verse en la *Figura 2.12* el interruptor tiene dos posiciones, cuando conecta la salida 1 con la 2 el sistema está apagado y, por el contrario, cuando conecta las salidas 2 y 3 el sistema está encendido. Los 5V que salen del circuito correspondiente a la *Figura 2.10* son los que entran al circuito del AMS1117.

Siguiendo las recomendaciones de la hoja de especificaciones, cuyo enlace se encuentra en el *Apéndice B*, se han instalado en el circuito una serie de condensadores para mejorar la estabilidad y la posible respuesta transitoria.

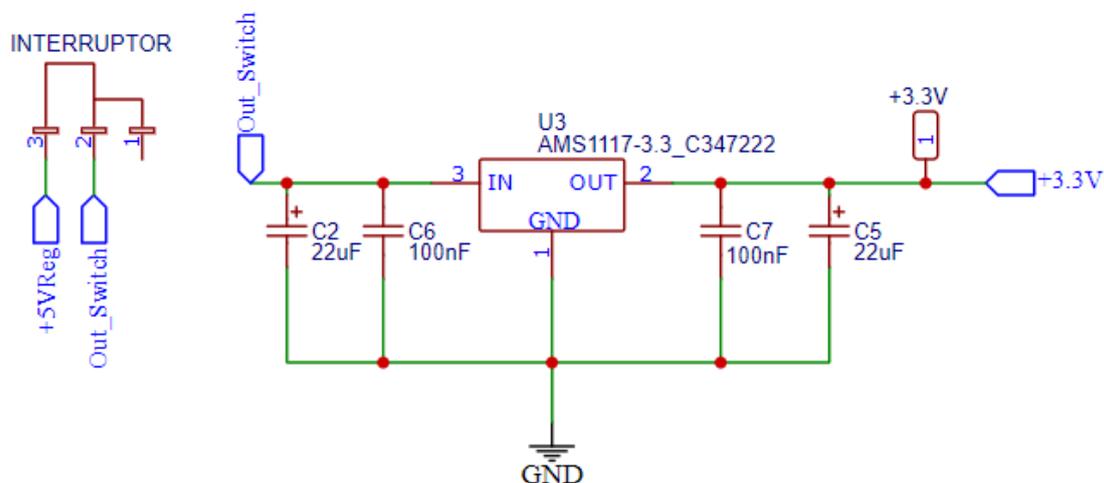


Figura 2.12: Sección del esquemático perteneciente al AMS1117

Para finalizar con este apartado, se presenta ahora la batería. En el presente trabajo se utiliza una batería de polímero de iones de litio. El uso de las baterías de polímero de litio es de lo más extendido, desde sistemas de radio-control en aviones de modelismo, cuadricopteros y coches teledirigidos hasta proyectos de electrónica personales pasando por todo tipo de proyectos portátiles. Debido a que tienen una gran densidad energética, al ser comparadas con baterías menos avanzadas tecnológicamente como las baterías de NiCd y NiMH, son utilizadas en gran cantidad de proyectos y sistemas donde se tiene una limitación tanto en peso como en espacio. Otra de sus ventajas es que tienen una curva de descarga bastante plana, yendo desde los 4.2V cuando están completamente cargadas, hasta los 3V cuando se descargan; teniendo una tensión nominal de 3.7V

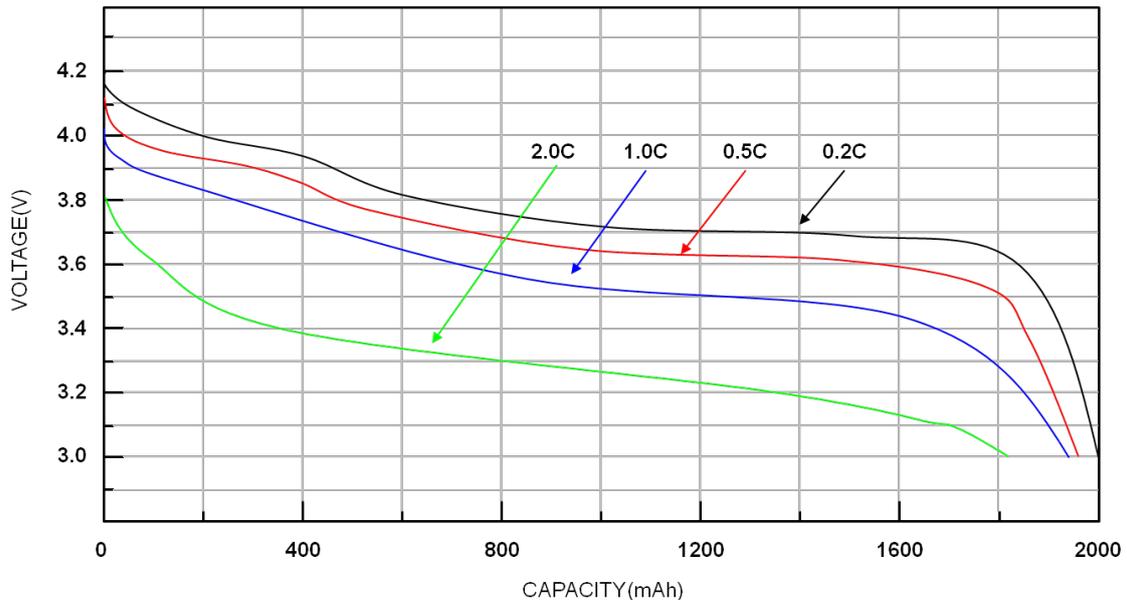


Figura 2.13: Curva de descarga típica de batería LiPo

Por otro lado, este tipo de baterías requiere de especial cuidado ya que pueden explotar o prenderse fuego si se cargan por encima de cierto nivel de tensión, al igual que si sufren daños estructurales que dejen expuestos los elementos del interior de la batería. Así mismo, también se debe de tener especial cuidado en la descarga de la misma, ya que al bajar de un valor cercano a 3V la batería queda inutilizable.

Estas baterías también tienen otra limitación que viene dada por la cantidad de corriente que puede entregar la batería de forma continua sin sufrir degradación así como la máxima cantidad de corriente que se le puede entregar a la batería en la carga. Dichas limitaciones de intensidad vienen indicadas por el termino de tasa de carga/descarga, representado por la letra C; este parámetro es la capacidad de la batería en mAh dividida entre 1h, por lo tanto, 1C para una batería con una capacidad de 1000 mAh es igual a 1A. En baterías LiPo estándar este parámetro de seguridad suele estar entorno a 1C. En concreto la batería utilizada en el presente trabajo consta de una capacidad de 450mAh y unas medidas de 30x30x5mm (*Figura 2.14*).

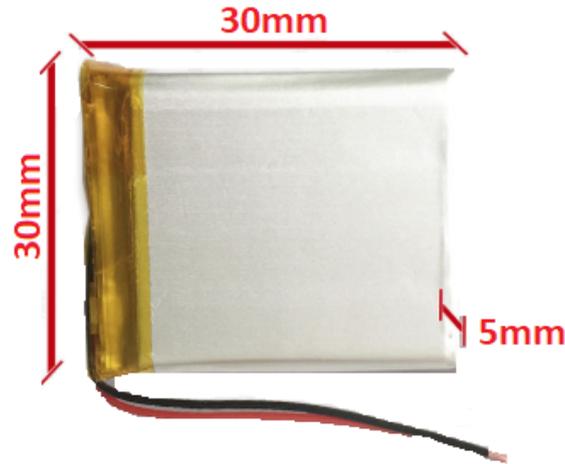


Figura 2.14: Fotografía de la batería LiPo

Se da aquí por finalizada la presente subsección habiendo cubierto tanto los componentes como las partes del circuito relacionados con la alimentación del sistema. En la parte derecha del esquemático de la *Figura 2.12* se encuentra un puerto denominado +3.3V que enlaza con la alimentación del módulo ESP12E. La descripción de los componentes y del esquemático sigue ahora en la siguiente subsección, dedicada en exclusiva al módulo ESP12E y a las partes del esquemático a él conectadas.

### 2.2.2. ESP12E y LED

Tal y como se ha comentado, la principal ventaja del proyecto desarrollado en el presente trabajo es la habilidad de comunicación de forma inalámbrica, es por ello que todo el proyecto gira entorno al módulo ESP12E, fabricado por Ai-Thinker. Dicho módulo proviene de una familia originada en base al SoC ESP8266, fabricado por la compañía Espressif Systems. Con lo que respecta a la conectividad inalámbrica, este SoC tiene soporte para IPv4 y los protocolos TCP/HTTP/FTP/UDP, siendo este último el utilizado en el presente trabajo.

La elección del módulo ESP12E, frente a otros de la familia, viene dada por ser uno de los que pone a disposición del usuario la mayoría de los pines del SoC. Como se puede ver en la (*Figura 2.15*) el módulo da acceso a 22 pines. En este proyecto tienen relevancia los correspondientes a GPIO, el pin ADC y los pines de comunicación UART utilizados para realizar la programación.

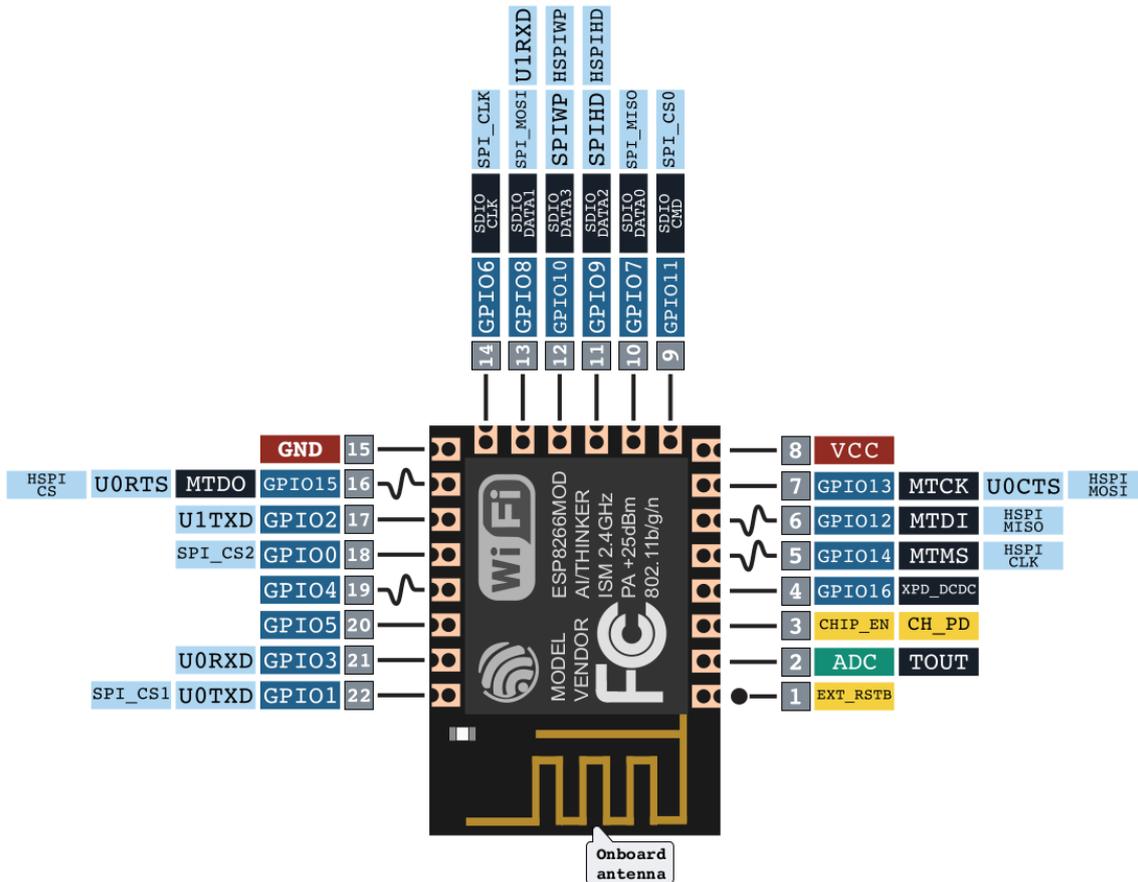


Figura 2.15: Pinout del módulo ESP12E

El ESP8266 dispone de diferentes modos de arranque. En este trabajo se hace uso de dos de ellos. El modo Flash Start-up ejecuta el programa alojado en la memoria flash del módulo, por otra parte el modo descarga UART se utiliza para subir un programa mediante UART a la memoria flash del módulo. La configuración de los pines para acceder a los mencionados modos de arranque se muestra en la *Tabla 2.2*. El GPIO2 puede estar al aire ya que en arranque está en valor alto, con respecto al GPIO0 para arrancar en modo normal también es suficiente con que se encuentre al vuelo, por el contrario, para entrar en modo programación UART sí que debe de contarse a GND.

	GPIO0	GPIO2	GPIO15
Flash Start-up	1	1	0
Descarga UART	0	1	0

Tabla 2.2: Configuración de los pines para los tipos de arranque

Cuando se utiliza cualquier módulo de la familia de dicho SoC, se debe de tener en especial consideración el hecho de que se alimenta y trabaja a una tensión de 3.3V. Por ello, en los casos como el del presente trabajo en el que la alimentación venga dada por una batería, es necesario regular la tensión de salida de la misma para alimentar de forma correcta el módulo para que no reciba una tensión fuera de su rango de seguridad y acabe inutilizándolo el SoC.

Por otro lado, la entrada analógica, ADC (*Analog to Digital Converter*), solo admite valores de tensión situados entre 0V y 1V en este SoC y tiene 10 bits de resolución. Dicho pin en el presente trabajo se utiliza para medir el estado de carga de la batería. Un enlace a la hoja de especificaciones completa de este módulo se encuentra en el *Apéndice B*.

Como puede verse en la *Figura 2.17* la tensión de 3.3V saliente del AMS1117 alimenta al ESP12-E a través del pin VCC, utilizando un condensador para estabilizar la corriente entrante. El pin EN se mantiene en alto mediante una resistencia pull up siempre y cuando el interruptor este en la posición que enciende el sistema.

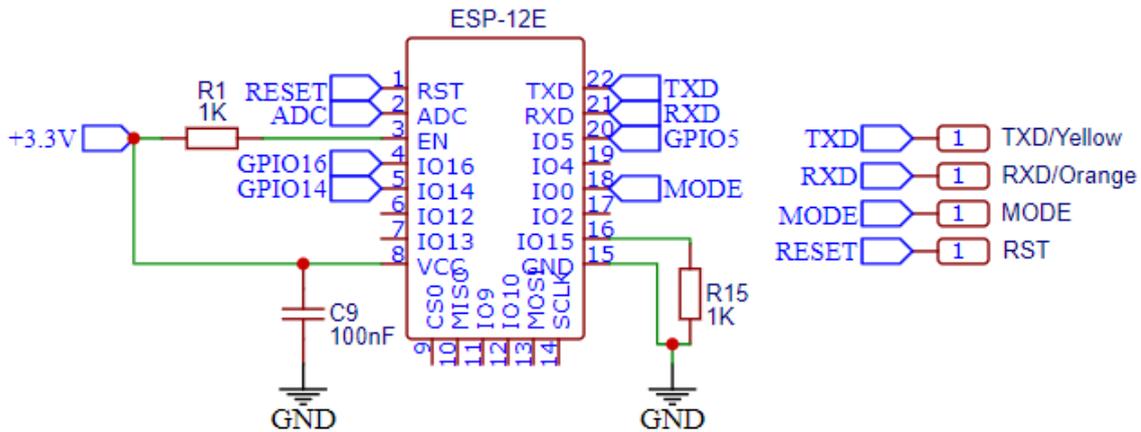


Figura 2.16: Sección del esquemático perteneciente al ESP12E

De los pines accesibles se utilizan 8, 4 para programar el módulo (TXD, RXD, RESET y IO0) y otros 4 para funcionamiento del sistema que se describe posteriormente en este apartado. En el esquemático de la *Figura 2.17* se han utilizado puertos para facilitar la representación de los esquemáticos que aparecerán posteriormente. Así mismo, se muestran en la parte derecha de la imagen los pads correspondientes a la interfaz UART utilizados para la programación del módulo así como el utilizado para determinar el modo de funcionamiento del módulo y el pin de reseteo. La interfaz UART se describe en la *Subsección 2.5* así como el entorno Arduino IDE y el procedimiento para programar el módulo.

En la *Tabla 2.3* se agrupan los puertos del ESP12E utilizados así como una breve descripción que se amplía seguidamente.

Terminal	Uso
ADC	Monitorización del nivel de batería
GPIO16	Mostrar nivel de batería
GPIO14	Mostrar nivel de batería
GPIO5	Controlar el LED de alta potencia

Tabla 2.3: Listado de pines utilizados y su uso

En la *Figura 2.17* se muestra el esquemático correspondiente al circuito que permite, mediante el convertor analógico/digital del ESP12E, monitorizar el estado de carga de la batería. Hay que recordar que, mientras la batería tiene un rango situado entre los 3V y los 4.2V, el pin del ESP12E puede recibir una tensión máxima de 1V. Es por esta razón que se realiza el divisor de tensión mostrado en la figura y regido por la ecuación siguiente:

$$V_{ADC} = \frac{R_{11}}{R_{11} + R_7} \cdot V_{Bat+_{Out}} \quad (2.2)$$

Por lo tanto, teniendo en cuenta el valor de las resistencias utilizadas,  $V_{ADC}$  no puede alcanzar, teóricamente, valores superiores a 0.9V. Asegurándose así la integridad del SoC ESP8266 en todo el rango operativo de la batería.

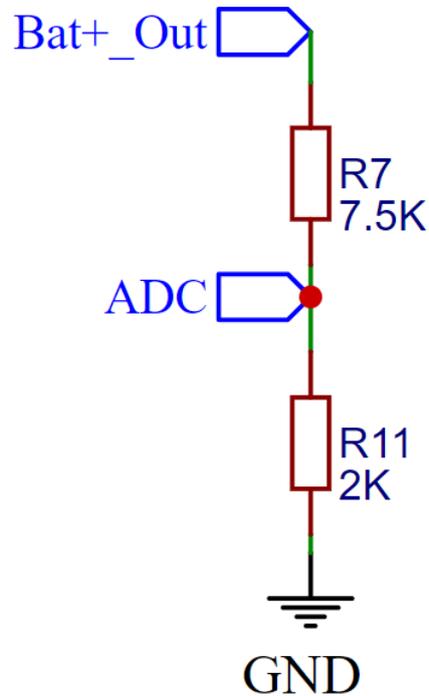


Figura 2.17: Sección del esquemático perteneciente al ADC

Los puertos GPIO 14 y 16 se utilizan para controlar unos leds utilizados para identificar visualmente el estado de carga de la batería. Para ello se realiza el esquemático representado en la *Figura 2.22*

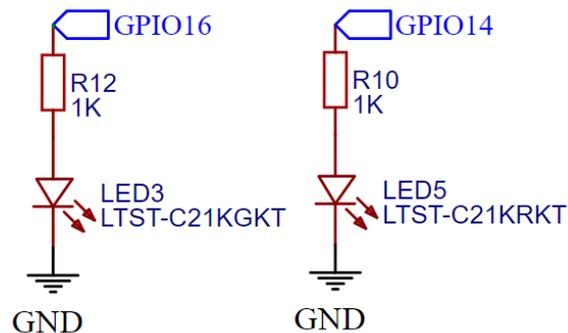


Figura 2.18: Sección del esquemático perteneciente a los leds del estado de la batería

Finalmente, para terminar con la *Sección 2.2*, se presentan los componentes de la parte de la iluminación y su esquemático. Debido a que la máxima corriente que se puede obtener de un GPIO del módulo son 12mA, para administrar una mayor cantidad de corriente se hace uso de un MOSFET de canal N (*Figura 2.19*) accionado mediante el GPIO5.

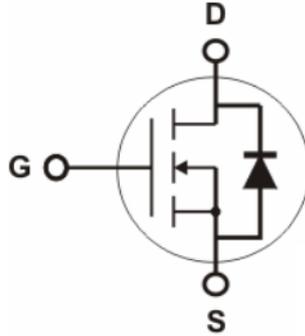


Figura 2.19: MOSFET canal N

Un MOSFET es un componente con tres terminales comúnmente conocidos como drain, gate y source (*Figura 2.19*). Este componente sirve para, mediante la tensión aplicada en la gate, controlar la intensidad que fluye entre drain y source. Dicho de otro modo funciona como un interruptor digital. Concretamente, un MOSFET de canal N permitirá el paso de corriente entre drain y source cuando se aplique una tensión positiva en la compuerta.

El modelo utilizado en el presente trabajo es el AO3400, *Figura 2.20*, fabricado por Shikues, la tensión de Breackdown que puede soportar entre drain y source es de 30V lo que, como se clarificará en el esquemático, supone un valor varias veces superior a los 5V que va a tener que soportar y, según el *Gráfico 2.20b*, teniendo en cuenta que la tensión  $V_{GS} = 3V$  el MOSFET utilizado permite más de 5A de corriente.

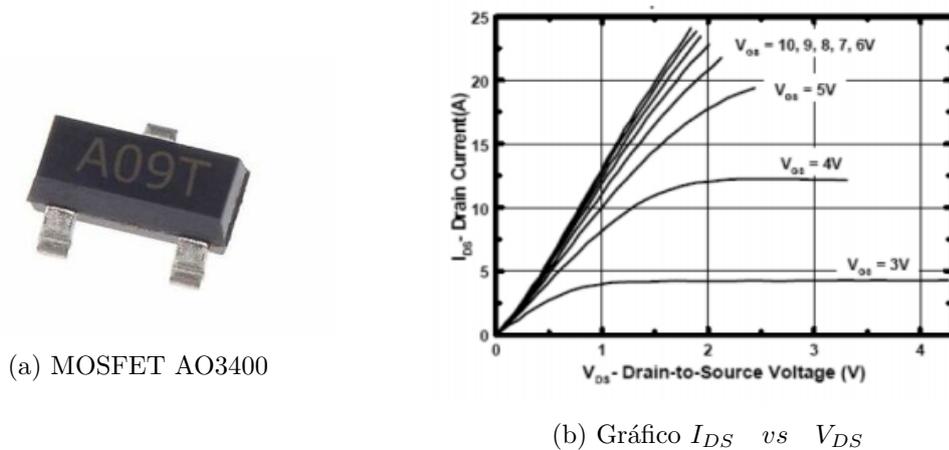


Figura 2.20: MOSFET AO3400 y curva característica

Se detalla ahora el último componente importante del sistema de iluminación, el LED de OSRAM modelo GOLDEN DRAGON con código de fabricante: LY W5SM-HZJZ-46-1.



Figura 2.21: LED Golden Dragon de OSRAM

Para terminar con este apartado se va a detallar el esquemático donde se controla el LED principal. Como se muestra en la *Figura 2.22* el GPIO5 del ESP12E es el encargado de controlar la tensión del terminal gate del MOSFET, mediante la resistencia R6 se asegura que la tensión de gate baje a 0 inmediatamente cuando lo haga el GPIO5.

Por otra parte, para alimentar el LED se utiliza el puerto Out\_Switch que provee una tensión de 5V (ver *Figura 2.12* o *subsección 2.2.1*). Para asegurar la estabilidad de esta sección del circuito se implementa el condensador C8 y mediante la resistencia R5 se regula la intensidad que circula a través del LED.

Teniendo en cuenta que el led tiene un salto de tensión de 2.3V , a través de la resistencia habrá un salto de 2.7V. Por lo tanto utilizando la ley de Ohm [ver 7, cap 2] se obtiene:

$$I_{LED} = \frac{V_{Out\_Switch} - 2,3V}{R5} = 0,54A \quad (2.3)$$

Donde R5 tiene es una resistencia de 5,11 Ohms, como se ve en la *Ecuación 2.3*, la intensidad que atraviesa el LED es de 0.54A. Hay que tener en cuenta que, en estas condiciones, R5 debe ser capaz de soportar 1.45W para no quemarse durante su uso.

Según su hoja técnica, en esta configuración, el LED debería actuar con una intensidad luminosa de 23 cd.

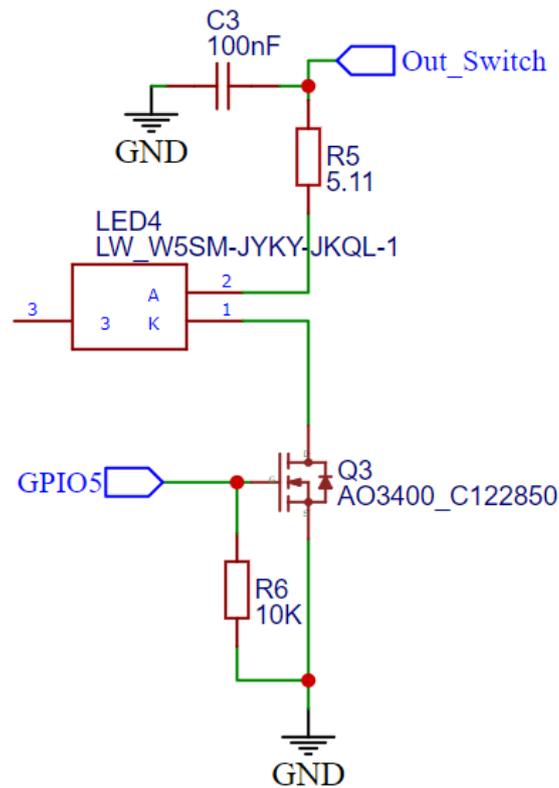


Figura 2.22: Sección del esquemático perteneciente al LED de alta intensidad

Se ha cubierto hasta aquí el diseño en lo referente a todo el hardware y al esquemático del sistema. Tanto para la realización del esquemático ya presentado, como para el diseño de la PCB se ha hecho uso de las herramientas de automatización de diseño electrónico que facilita la página web EasyEda [8].

## 2.3. PCB

Una PCB (*Placa de Circuito Impreso*) es una superficie donde se han realizado pistas en material conductor sobre una base no conductora. Con su uso se consigue conectar eléctricamente y sostener un conjunto de componentes de una forma compacta. De manera general las pistas son de cobre y para la fabricación de la base suele hacerse uso de resinas de fibra de vidrio o plástico entre otros.

Una PCB puede estar constituida por diferentes capas favoreciendo así la miniaturización. Las pistas pueden tener la posibilidad de empezar en una de las capas y a lo largo de su recorrido pasar por otras diferentes haciendo uso de vías, esto resulta de gran utilidad cuando se pretende minimizar al máximo el tamaño de la PCB ya que amplía las posibles rutas que puede recorrer una pista para unir dos terminales. Por otro lado, una herramienta útil que se tiene en el diseño de PCB es el área de cobre, haciendo uso de esta herramienta es posible crear una área en la PCB que interconecte varios terminales sin necesidad de crear pistas de manera individual. Esto resulta de especial interés práctico cuando hay muchos componentes que comparten una conexión, como puede ser la conexión a tierra.

En esta sección se expone el diseño de la PCB y algunas características y definiciones relacionadas con este elemento. Para la realización del diseño se han utilizado las huellas que facilita la web de EasyEda. Una huella es el diseño de los pads de soldadura que se corresponden en su medida y disposición con los de los terminales del componente al que hacen referencia. Así mismo, para reducir el tamaño lo máximo posible, se ha hecho uso de las dos caras de la PCB.

Uno de los parámetros que hay que determinar en el diseño de la PCB es el ancho que van a tener las pistas que conecten los elementos de la placa. Este parámetro tiene una gran importancia y en este trabajo se han seguido las recomendaciones estandarizadas de IPC-2221A [9]. Para la aplicación de dicho estándar se ha utilizado la herramienta proporcionada por la compañía norteamericana Advanced Circuits [10].

Teniendo en cuenta que el consumo máximo que puede alcanzar el ESP8266 son 0.4A, que el TP4056 está configurado para cargar la batería a 0.36A y que el LED cuando se encuentra activo consume 0.35A se ha establecido una anchura de las pistas de 0.45mm. Esta configuración tiene un buen margen de seguridad ya que con estas dimensiones, las pistas deben de permitir una intensidad de hasta 1.34A sin ocasionar una subida de temperatura de las mismas mayor a 10°.

El primer paso en el diseño ha sido la determinación del tamaño de la PCB, se ha limitado el tamaño de la placa a un cuadrado de 50x50mm, manteniendo así una miniaturización razonable para la implementación en drones de hasta 3m de envergadura. Posteriormente, se han colocado tanto la huella del interruptor como la del puerto micro-USB ya que estos elementos debían de estar emplazados en un lado del cuadrado para posibilitar su acceso.

El siguiente paso ha sido el emplazamiento del LED de alta potencia y del módulo ESP12E. Ambos se han situado en la parte central de la placa, cada uno en una de sus caras. El LED se ha localizado en esa posición central para favorecer la emisión de luz, por otra parte, los motivos de colocar centralmente el módulo ESP12E son tanto su tamaño como el alto número de trazas de cobre que se conectan a sus terminales.

Una vez estos dos elementos principales han sido colocados, se ha intentado organizar los restantes componentes en grupos lógicos para así minimizar la longitud de las trazas siempre teniendo en cuenta que la altura de los componentes colocados en la parte frontal de la PCB no tuviese una elevación superior a los 2mm con respecto de la superficie de la misma. Esta última condición debe de satisfacerse para que ningún componente intersekte con la tapa superior de la caja, que tiene una elevación sobre la superficie de la PCB de 2mm, como se verá en la *Sección 2.6*.

Por último, se realizan en las esquinas taladros de 3mm para facilitar la sujeción de la placa a una caja.

Una vez el diseño de la placa está finalizado, *Figura 2.23*, se generan los archivos para el fabricante. Gerber es el formato de archivo estándar que se utiliza para trasladar la información del diseño de una PCB a un fabricante. La información del diseño de una PCB completa está formada por diversos archivos gerber que, en conjunto, contienen los datos que el fabricante necesita para crear la PCB. La fabricación del prototipo de este trabajo se ha encargado a la compañía JLCPCB a través de su página web [11].

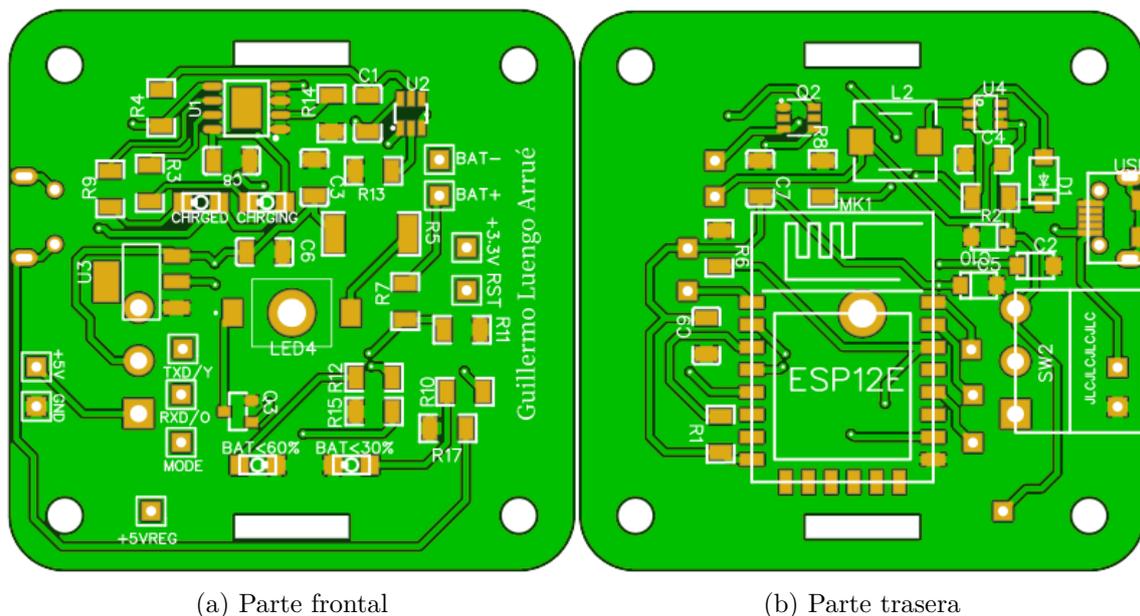
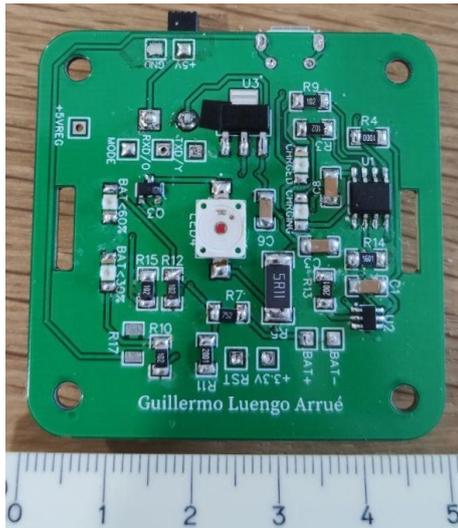


Figura 2.23: Imagen de la PCB (diseño)

Se muestra, a continuación, el resultado obtenido en la fabricación del prototipo con los componentes ya soldados en la placa, *Figura 2.24*. Se ha añadido una regla en las imágenes como validación de la dimensión de la PCB cuadrada ( $50 \times 50 \text{ mm}$ ).



(a) Parte frontal



(b) Parte trasera

Figura 2.24: Imagen de la PCB

Una vez presentado este diseño, hay que mencionar que previamente se desarrolló un diseño distinto que fue descartado debido a no haberse diseñado con el objetivo de ser alojado en una caja contenedora. No obstante, para comprobar el funcionamiento del sistema se hace uso de placas con el diseño descartado. Por lo tanto, el mismo se presenta en el *Apéndice A*.

## 2.4. Protocolos de Comunicación

En la comunicación de datos el principal problema que surge es como llevar la información entre dos puntos diferentes sin errores utilizando para ello una codificación determinada. En los extremos de la comunicación se encuentran los elementos receptores y transmisores de información que deben ser capaces de decodificar y codificar los datos que viajan a través del medio transmisor. Hay que tener en cuenta que el sistema encargado de realizar la transmisión actúa de forma impermeable al contenido que se transmite.

En definitiva, un protocolo de comunicaciones es un conjunto de reglas que habilita, a las entidades que forman parte de un sistema de comunicación, para transmitirse información entre ellas mediante cualquier variación de una magnitud física. Se encarga pues, de estandarizar un conjunto de normas de codificación y decodificación. Evidentemente, para que un protocolo tenga éxito en su aplicación debe de ser aceptado y implantado por los participantes involucrados en la comunicación.

En la presente sección se exponen, brevemente, los diferentes protocolos de comunicación que intervienen en este Trabajo Final de Grado.

### 2.4.1. Comunicación Asíncrona UART

Tal y como se mencionó en la *subsección 2.2.2*, para programar el módulo ESP12E hace falta arrancarlo en un modo específico que permite recibir los datos del programa a través de los pines UART (*Transmisor/Receptor Universal Asíncrono*) del módulo. En este apartado se hace una introducción breve a este protocolo de comunicación.

UART es un protocolo de comunicación en serie creado en 1960 por Gordon Bell. En un protocolo de este tipo los datos se envían uno tras otro y, al ser asíncrono, no requiere de ningún reloj común entre transmisor y receptor para poder establecer una comunicación de forma efectiva. Una de las principales desventajas que presenta este protocolo es el hecho de que únicamente se pueden comunicar dos dispositivos a la vez.

No es un protocolo que permita una transmisión de información muy rápida, su máximo se encuentra en 115200 bits/s, comparado con alternativas de comunicación en serie como son los protocolos I2C y SPI con tasas máximas de uno y dos ordenes de magnitud superiores respectivamente.

Este protocolo de comunicación hace uso exclusivamente de dos cables, que se conectan de forma cruzada a los terminales TX y RX de los componentes. De este modo, un extremo del cable se conecta a TX de un componente y el otro extremo se conecta al terminal RX del otro componente tal y como se muestra en la *Figura 2.25*.

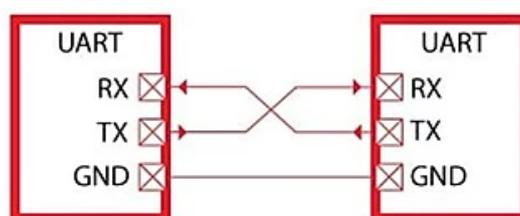


Figura 2.25: Comunicación UART asíncrona

Tal y como se ha mencionado este protocolo no hace uso de una línea de reloj, pero si que es necesario fijar una velocidad de transmisión de bits por segundo, en baudios, para que los dos componentes se puedan comunicar correctamente. Por otro lado, también es requisito indispensable para la correcta comunicación que ambos dispositivos compartan la misma conexión a tierra.

Así pues, una vez el módulo ESP12E se encuentra en modo programación, solo es necesario conectar sus terminales UART (*Figura 2.15*) al ordenador mediante un adaptador USB a UART como el mostrado en la *Figura 2.26*.



Figura 2.26: Cable adaptador comunicación USB a UART

Por último, hace falta un programa que se encargue de compilar el programa y subirlo al módulo. El software Arduino IDE [6] es el utilizado en este Trabajo Final de Grado.

#### 2.4.2. Protocolo UDP

Para cubrir la parte de comunicación inalámbrica entre los módulos ESP12E se hace uso de las capacidades WiFi que el SoC ESP8266 incorpora. Como se ha mencionado anteriormente, uno de los módulos funciona como punto de acceso mientras que el resto se conectan a la red que crea el primero. Para transferir información a través de esta red inalámbrica en este Trabajo Final de Grado se ha optado por el protocolo UDP (*Protocolo de Datagramas de Usuario*).

El protocolo UDP es uno de los protocolos de transporte que se encuentra dentro de la suite TCP/IP. Este protocolo no resulta del todo fiable ya que no tiene un sistema para recuperar o controlar errores y por lo tanto no se utiliza en aplicaciones en las que debe asegurarse que el contenido de la transmisión llega siempre al host de forma correcta.

Pero las siguientes características de este protocolo lo hacen especialmente interesante para la aplicación a este Trabajo Final de Grado:

- Rapidez debido al que no realiza establecimiento de conexión antes de enviar datos
- Permite comunicaciones broadcast/difusión
- Protocolo sencillo

Al no necesitar los paquetes necesarios para realizar el establecimiento de conexión el servidor, en este caso un módulo ESP12E, tiene una menor carga de trabajo. Por otro lado, gracias a las capacidades de broadcast, se puede difundir un mensaje a todos los usuarios conectados a la red. Esta capacidad para difundir un mensaje se utiliza para poder enviar una señal a todos los módulos del sistema de luces anticolidión para que parpadeen de forma sincronizada.

Por otro lado, la pérdida de datos o errores en la transmisión no resultan críticos en la aplicación del sistema ya que al contar con diversos módulos, en el caso eventual de que alguno no reciba la señal de un parpadeo los otros sí lo harán.

Por lo tanto, el protocolo UDP se ha escogido debido a que permite difundir un mensaje a todos los dispositivos conectados a la red y, al mismo tiempo, el uso de este protocolo permite una menor carga de trabajo para el SoC que alternativas como el protocolo TCP.

## 2.5. Software

Una vez ya se han diseñado los circuitos y la PCB, el siguiente paso en el diseño es crear el código para que el sistema cumpla con lo requerido. Como se ha comentado anteriormente, en este sistema de luces anticolidión habrá dos tipos de funcionamiento, se muestran en la *Tabla 2.4*.

Modo 1	Crea red WiFi y envía señal para parpadeo de LED a 60bpm
Modo 2	Unirse a la red y estar a la espera para accionar el LED

Tabla 2.4: Modos de funcionamiento de los módulos del sistema

Cada modo distinto de funcionamiento hace uso de un código diferente. Los códigos escritos que hacen que los módulos cumplan con su funcionamiento se encuentran expuestos en las *Subsecciones 2.5.2* y *2.5.3*. El módulo que funciona con el Modo 1 debe contener el *Código 2.1* y para que el módulo funcione con el Modo 2, se le debe subir el *Código 2.2*.

Tanto para escribir el código como para subirlo a los módulos ESP12E se ha hecho uso del software proporcionado por Arduino en su página web, Arduino IDE. Una breve exposición de dicho software se expone en la *Subsección 2.5.1*.

### 2.5.1. Arduino IDE

El primer paso para la realización del código es la instalación de la plataforma de software Arduino IDE disponible a través de su página web [6]. Este entorno de desarrollo integrado permite la programación de placas compatibles, como es el caso de la ESP12E al estar basada en el SoC compatible ESP8266.

Así mismo, este programa mediante el uso de librerías desarrolladas por la comunidad o por los propios desarrolladores permite una implementación sencilla de diferentes funciones

Al tratarse de una placa compatible pero no ser originalmente una placa Arduino, es necesario realizar un pequeño set-up en el programa Arduino IDE. Es necesario añadir, a través del menú de gestión de placas, la compatibilidad con el SoC ESP8266. Tras este paso en el menú herramientas es posible seleccionar la placa ESP12E. Estos pasos no tienen una excesiva complejidad debido a la gran comunidad de desarrolladores que existe detrás del ecosistema de Arduino.

Tras esta configuración inicial, ya se puede proceder a la escritura del código del programa.

### 2.5.2. Código del punto de acceso

En esta parte se expone el código, con comentarios descriptivos, utilizado para que el módulo ESP12E cree una señal WiFi y, utilizando el protocolo UDP, envíe un carácter cada segundo a través de la red a todos los dispositivos conectados a ella.

Antes de la exposición del código se detallan algunas de las partes más relevantes.

- En las líneas 3 y 5 se incluyen las librerías que permiten utilizar las funciones WiFi del SoC8266 así como el protocolo UDP.
- Entre las líneas 40 y 86 tiene lugar el monitoreo de la batería. Se realiza siempre que se enciende el sistema y la información se ofrece al usuario mediante dos LEDs ubicados en la PCB.
- Entre líneas 124 y 141 uso de la función millis() dentro de condicionante para enviar la señal cada segundo. Alternativas a esta programación, como utilizar función delay(), provocan reinicios debido a bloqueo de funciones ocultas.

```

1 //Inclusión de librerías
2 //Incluye la librería WiFi del SoC ESP8266
3 #include <ESP8266WiFi.h>
4 //Incluye la librería del protocolo de comunicación UDP
5 #include <WiFiUdp.h>
6
7
8 //Configuración de las direcciones de red
9 IPAddress local_IP(192, 168, 4, 1);
10 IPAddress gateway(192, 168, 4, 0);
11 IPAddress subnet(255, 255, 255, 0);
12
13 //Definición de variables
14 const char *ssid = "PAcceso"; //Nombre de la red que va a crear
15
16 unsigned int localPort = 2000;
17 unsigned long Periodo = 1000; //1 segundo
18 unsigned long Milisprevio = 0;
19 int x = 0;
20
21 //Define dirección IP para difusión en la red
22 IPAddress SendIP(192, 168, 4, 255);
23 //Crea objeto wifiudp udp, permite enviar y recibir paquetes por UDP
24 WiFiUDP udp;
25
26 //Bloque correspondiente al conversor analógico/digital
27 const int analogInPin = A0; // ESP8266 Pin analógico ADC0 = A0
28 int sensorValue = 0; // Value read from the pot
29 int outputValue = 0; // Value to output to a PWM pin
30 int countadc=0;
31
32 //-----//
33 //Configuración inicial
34 //-----//
35 void setup() {
36
37 //Pines 16 y 14 controlan LEDs de batería, se definen como salida
38 pinMode(16, OUTPUT); // >60%
39 pinMode(14, OUTPUT); // >30%
40 //Bloque para medir nivel de batería al encender el sistema

```

```

41 //Bucle while que pasa 10 veces para mostrar el nivel de batería (
    countadc=0)
42 while (countadc < 10) {
43 //asigna un valor entre 0 y 1024 a sensorValue según la tensión de la
    batería
44 sensorValue = analogRead(analogInPin);
45 //map a 1024 es 1V en el divisor de tension, que da si entran 4.75V.
    a 4.2 está al maax
46 //La función map remapea un numero dentro de un rango en otro rango
    distinto.
47 //Esto permite transformar el valor obtenido de la lectura del pin
    analógico a el valor
48 //de tensión de la batería un tratamiento más sencillo. Dicho valor
    se almacena en
49 //outputValue
50 outputValue = map(sensorValue, 0, 1024, 0, 475);
51 //Se imprime en la consola el valor de tensión para comprobaciones
52 Serial.print("Tensión = ");
53 Serial.print(outputValue);
54 //Control de los LEDs indicadores de batería según el nivel leído por
    //el ADC
55
56 if (outputValue < 460 && outputValue > 372) {
57 //Con más del 60% de carga se quedan los 2 LEDs fijos
58 digitalWrite(16, HIGH);
59 digitalWrite(14, HIGH);
60 delay(500); //Espera de 0.5 segundos
61 countadc = countadc + 1; //Incremento en el contador
62
63 } else if (outputValue <= 372 && outputValue > 336) {
64 //Entre 60% y 30% de carga parpadea el LED >60%
65 digitalWrite(16, HIGH);
66 delay(500); //Espera de 0.5 segundos
67 countadc = countadc + 1; //Incremento en el contador
68 digitalWrite(16, LOW);
69 delay(500); //Espera de 0.5 segundos
70 } else { //
71 //Menos de un 30% de batería parpadea el LED >30%
72 digitalWrite(14, HIGH);
73 delay(500); //Espera de 0.5 segundos
74 countadc = countadc + 1; //Incremento en el contador
75 digitalWrite(14, LOW);
76 delay(500); //Espera de 0.5 segundos
77 }
78 }
79 //Salida del bucle del monitoreo de batería
80 //Apagado de los LEDs
81 digitalWrite(16, LOW);
82 digitalWrite(14, LOW);
83
84 //—————//
85 //Fin del monitoreo de batería
86 //—————//
87
88 //Establece la velocidad de datos en bits por segundo (baudios) para la
    transmisión de datos en serie
89 Serial.begin(115200);
90 //Establece el modo de funcionamiento. Punto de acceso
91 WiFi.mode(WIFI_AP);
92 //Establece el modo de funcionamiento. Punto de acceso
93 //Imprime, a través del puerto serie, texto en formato ASCII
94 //Usado para comprobación, imprime en puerto serie información de la
    configuración de la red
95 Serial.print("Setting soft-AP configuration ... ");

```

```

96  Serial.println(WiFi.softAPConfig(local_IP, gateway, subnet) ? "Ready" :
    "Failed!");
97  Serial.print("Setting soft-AP ... ");
98  //Espera hasta que se haya creado el punto de acceso, imprime puntos
    cada 0.1s
99  while (!WiFi.softAP(ssid))
100 {
101     Serial.println(".");
102     delay(100);
103 }
104 //Una vez creado el punto de acceso, imprime los parametros del mismo a
    través del puerto serie
105 Serial.print("Iniciado AP ");
106 Serial.println(ssid);
107 Serial.print("IP address:\t");
108 Serial.println(WiFi.softAPIP());
109 //Inicializa el protocolo UDP
110 Serial.println("Starting UDP");
111 udp.begin(localPort);
112 Serial.print("Local port: ");
113 Serial.println(udp.localPort());
114 }
115
116 //-----//
117 //Programa principal
118 //-----//
119
120 void loop() {
121     //millis() Devuelve el número de milisegundos desde que la placa
        Arduino empezó a ejecutar
122     //Compara tiempo actual-tiempo previo con el periodo de 1 segundo
123     //Si el incremento de tiempo es mayor a 1 segundo entra dentro del
        condicionante
124     if (millis() - Milisprevio > Periodo) {
125         Milisprevio = millis();//almacena el tiempo en Milisprevio
126         if (x == 0) {
127             x = 1;
128             //Envío de string a todos los host conectados, puerto objetivo 2000
129             udp.beginPacket(SendIP, 2000);
130             udp.write("1");
131             udp.endPacket();
132             //Termina el envío
133         }
134         //Envío de string a todos los host conectados, puerto objetivo 2000
135         else {
136             x = 0;
137             udp.beginPacket(SendIP, 2000);
138             udp.write("0");
139             udp.endPacket();
140         }
141     }
142 }

```

Código 2.1: Código del módulo configurado como punto de acceso

### 2.5.3. Código de las estaciones

Siguiendo con la descripción del código, en esta subsección se presenta, con comentarios descriptivos, el programa que permite al módulo ESP12E conectarse al módulo que funciona como punto de acceso. Así mismo, el módulo conectado está a la escucha de mensajes transmitidos bajo protocolo UDP para la activación del LED.

Al igual que se hizo en la subsección anterior, antes de proceder con el código utilizado, se exponen algunos de los puntos más relevantes del mismo.

- En las líneas 3 y 5 se incluyen las librerías que permiten utilizar las funciones WiFi del SoC8266 así como el protocolo UDP.
- Entre las líneas 38 y 87 tiene lugar el monitoreo de la batería. Se realiza siempre que se enciende el sistema y la información se ofrece al usuario mediante dos LEDs ubicados en la PCB.
- El pin 5 controla el MOSFET encargado de accionar el LED OSRAM, se define como salida en la línea 90
- El programa principal, líneas 123 a 148, es un loop que chequea la llegada de paquetes bajo protocolo UDP. En caso de entrada enciende el LED.
- Entre líneas 142 y 147 uso de la función millis() dentro de condicionante para apagar el LED pasados 0.15 segundos. Alternativas a esta programación, como utilizar función delay(), provocan reinicios debido a bloqueo de funciones ocultas.

```

1 //Inclusión de librerías
2 //Incluye la librería WiFi del SoC ESP8266
3 #include <ESP8266WiFi.h>
4 //Incluye la librería del protocolo de comunicación UDP
5 #include <WiFiUdp.h>
6
7
8 //Configuración de las direcciones de red
9 IPAddress staticIP(192, 168, 4, 8); //Cada estación debe tener una IP
   distinta
10 IPAddress gateway(192, 168, 4, 0);
11 IPAddress subnet(255, 255, 255, 0);
12
13 //Definición de variables
14 const char *ssid = "PAcceso"; //Nombre del punto de acceso al que se va a
   conectar
15
16 unsigned int localPort = 2000; //Puerto a la escucha para recibir
   paquetes UDP
17 char incomingPacket[255]; //Buffer para almacenar los paquetes entrantes
18
19 //Bloque correspondiente al conversor analógico/digital
20 const int analogInPin = A0; // ESP8266 Pin analógico ADC0 = A0
21 int sensorValue = 0; // Value read from the pot
22 int outputValue = 0; // Value to output to a PWM pin
23 int countadc = 0;
24
25 //Crea objeto wifiudp udp, permite enviar y recibir paquetes por UDP
26 WiFiUDP udp;
27

```

```

28 char packetBuffer[9]; //Donde se obtienen los datos UDP
29 unsigned long Milisprevio = 0;
30 #define LED_BUILTIN 2
31
32
33 //-----//
34 //Configuración inicial
35 //-----//
36 void setup() {
37
38 //Pines 16 y 14 controlan LEDs de batería, se definen como salida
39 pinMode(16, OUTPUT); // >60%
40 pinMode(14, OUTPUT); // >30%
41 //Bloque para medir nivel de batería al encender el sistema
42 //Bucle while que pasa 10 veces para mostrar el nivel de batería (
  countadc=0)
43 while (countadc < 10) {
44 //asigna un valor entre 0 y 1024 a sensorValue según la tensión de la
  batería
45 sensorValue = analogRead(analogInPin);
46 //map a 1024 es 1V en el divisor de tension, que da si entran 4.75V.
  a 4.2 está al maax
47 //La función map remapea un numero dentro de un rango en otro rango
  distinto.
48 //Esto permite transformar el valor obtenido de la lectura del pin
  analógico a el valor
49 //de tensión de la batería un tratamiento más sencillo. Dicho valor
  se almacena en
50 //outputValue
51 outputValue = map(sensorValue, 0, 1024, 0, 475);
52 //Se imprime en la consola el valor de tensión para comprobaciones
53 Serial.print("Tensión = ");
54 Serial.print(outputValue);
55 //Control de los LEDs indicadores de batería según el nivel leído por
  //el ADC
56
57 if (outputValue < 460 && outputValue > 372) {
58 //Con más del 60% de carga se quedan los 2 LEDs fijos
59 digitalWrite(16, HIGH);
60 digitalWrite(14, HIGH);
61 delay(500); //Espera de 0.5 segundos
62 countadc = countadc + 1;//Incremento en el contador
63
64 } else if (outputValue <= 372 && outputValue > 336) {
65 //Entre 60% y 30% de carga parpadea el LED >60%
66 digitalWrite(16, HIGH);
67 delay(500); //Espera de 0.5 segundos
68 countadc = countadc + 1;//Incremento en el contador
69 digitalWrite(16, LOW);
70 delay(500); //Espera de 0.5 segundos
71 } else { //
72 //Menos de un 30% de batería parpadea el LED >30%
73 digitalWrite(14, HIGH);
74 delay(500); //Espera de 0.5 segundos
75 countadc = countadc + 1;//Incremento en el contador
76 digitalWrite(14, LOW);
77 delay(500); //Espera de 0.5 segundos
78 }
79 }
80 //Salida del bucle del monitoreo de batería
81 //Apagado de los LEDs
82 digitalWrite(16, LOW);
83 digitalWrite(14, LOW);
84

```

```

85 //-----//
86 //Fin del monitoreo de batería
87 //-----//
88
89 //Pin 5 controla el LED Osram, se define como salida
90 pinMode(5, OUTPUT); // el osram
91 //Establece la velocidad de datos en bits por segundo (baudios) para la
    transmisión de datos en serie
92 Serial.begin(115200);
93 Serial.println();
94 //Configuración de los parámetros del módulo en la red
95 WiFi.config(staticIP , gateway, subnet);
96 //Inicia la conexión con el punto de acceso
97 WiFi.begin(ssid);
98 //Bucle que imprime en el monitor serie el momento de conexión
99 Serial.println("");
100 //Bucle de espera hasta que se ha conectado y da los datos de
101 //la conexión
102 while (WiFi.status() != WL_CONNECTED) {
103     delay(500);
104     Serial.print(".");
105 }
106 Serial.println("");
107 Serial.print("Connected to ");
108 Serial.println(ssid);
109 Serial.print("IP address: ");
110 Serial.println(WiFi.localIP());
111 //Inicia el protocolo UDP
112 Serial.println("Starting UDP");
113 udp.begin(localPort);
114 Serial.print("Local port: ");
115 Serial.println(udp.localPort());
116
117 }
118
119 //-----//
120 //Programa principal
121 //-----//
122
123 void loop() {
124     //Chequeo para comprobar si hay un paquete UDP y almacena su tamaño en
        packetSize
125     int packetSize = udp.parsePacket();
126     //Si se ha recibido un paquete UDP, entonces encender el LED
127     if (packetSize) {
128         //Lectura de datos UDP en el buffer especificado
129         int len = udp.read(incomingPacket, 255);
130         if (len > 0)
131         {
132             incomingPacket[len] = 0;
133         }
134         //Independientemente del string recibido, enciende el LED
135         //Se evita la posibilidad de un error debido a los fallos posibles
        dados por el protocolo UDP
136         if (strcmp(incomingPacket, "1") == 0) {
137             digitalWrite(5, HIGH);
138         }
139         else {
140             digitalWrite(5, HIGH);
141         }
142         //Bucle de espera de 0.15 segundos
143         Milisprevio = millis();
144         while (millis() - Milisprevio < 150) {

```

```
145     }  
146     //Apagado del LED  
147     digitalWrite(5, LOW); //El 5 es es osram  
148 }  
149 }
```

Código 2.2: Código del módulo configurado como estación

## 2.6. Diseño de la caja

En esta sección se presenta el proceso de diseño de la caja contenedora donde se aloja tanto la batería que alimenta al sistema como la PCB que contiene todos los componentes con sus conexiones. Para la realización de este componente se ha utilizado el software de diseño asistido por computadora (*CAD*) Fusion 360®[12].

Un software CAD permite la creación de modelos 3D, mediante el uso de ordenadores. Esta herramienta es utilizada por ingenieros para aumentar su productividad, mejora la calidad del producto y asimismo dota de una gran versatilidad a todo el proceso de diseño. La parametrización que brinda el uso de este tipo de software otorga al diseño de una gran precisión así como abre la puerta a posibles revisiones de diseño donde es posible cambiar alguno de los parámetros sin afectar al resto.

Así mismo, realizar el diseño en un software CAD posibilita exportar el mismo a través de archivos generados por el programa para crear copias de seguridad o compartir el diseño. Del mismo modo, se tiene la opción de realizar la impresión del diseño mediante el uso de una impresora 3D. Esto es un gran avance ya que permite la fabricación de un prototipo con una gran precisión cuya obtención no sería posible por otros medios.

Como se ha comentado, para realizar el diseño se ha utilizado el software Fusion 360 haciendo uso de una licencia de estudiante. Este software permite crear diseños CAD 3D con una única herramienta de diseño de productos integrada, basada en nube y fácil de usar.

El diseño de la caja se ha estructurado en dos partes, una tapa superior y la caja contenedora inferior. En primera instancia, se pretendía, una vez finalizado el diseño, realizar la impresión del mismo utilizando una impresora 3D.

El modelo utilizado para la impresión es el modelo Micro, *Figura 2.27*, ofrecido por el fabricante M3D [13].



Figura 2.27: Impresora M3D Micro

No obstante, viendo que los resultados obtenidos en la impresión de la tapa superior no satisfacían las necesidades del proyecto, se optó por utilizar metacrilato para la mencionada parte. Los resultados obtenidos en la impresión de la parte inferior cumplieron las expectativas sin problema.

### 2.6.1. Tapa superior

En esta sección se va a realizar una guía paso a paso para obtener el diseño de la tapa superior.

El primer paso es realizar un sketch de 50x50 mm, tras lo cual se realiza una extrusión de 2mm y un fillet a las 4 esquinas de 5 mm.

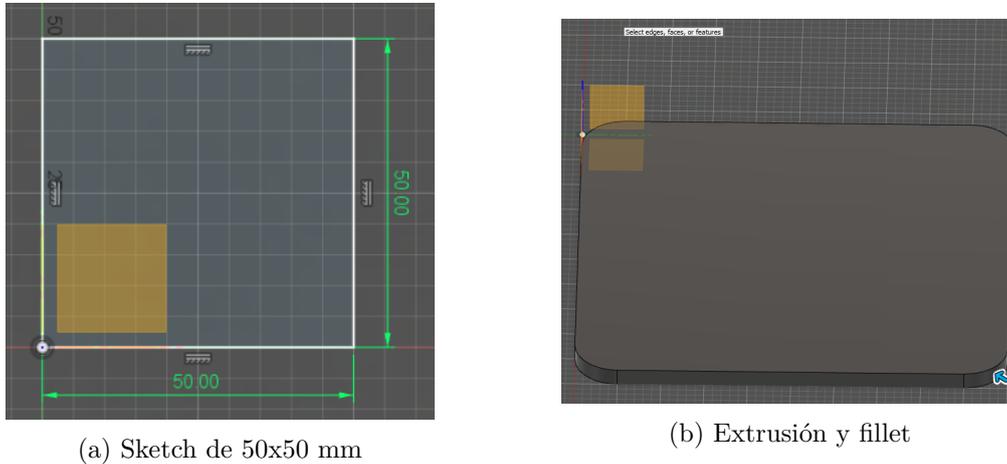


Figura 2.28: Paso 1 del diseño de la tapa superior

El siguiente paso es la realización de los agujeros *Figura 2.29*. Estos se encuentran a 5 mm de las esquinas y tienen un diámetro de 3mm, para realizarlos se hace uso de la herramienta hole.

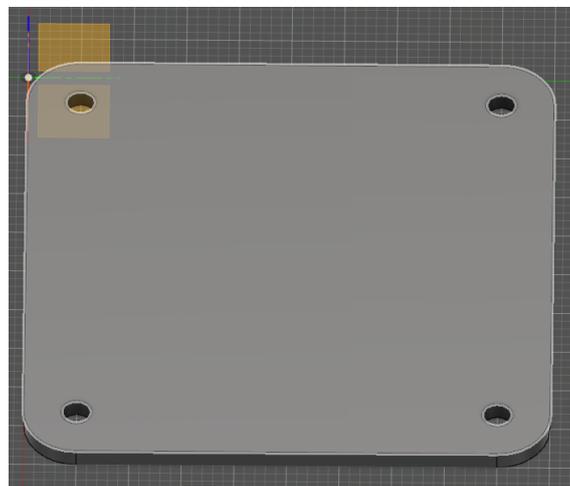


Figura 2.29: Agujeros tapa superior

El resultado de estas operaciones se exporta como archivo STL para realizar la impresión. Esta pieza se utiliza como guía para cortar una de las piezas de metacrilato que compone la tapa superior, *Figura 2.30*.

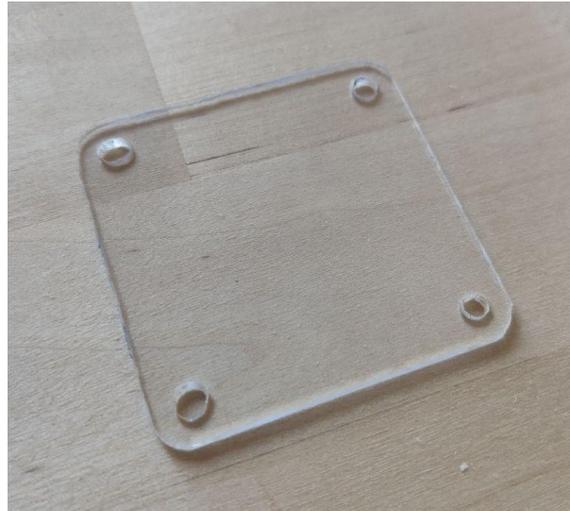


Figura 2.30: Pieza de metacrilato superior

Partiendo de la pieza de la *Figura 2.29* se realiza el sketch que se muestra en detalle en la *Figura 2.31* sobre su cara superior.

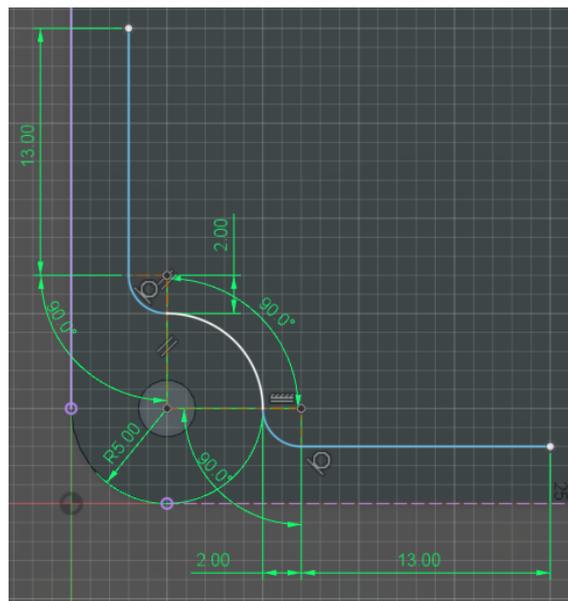
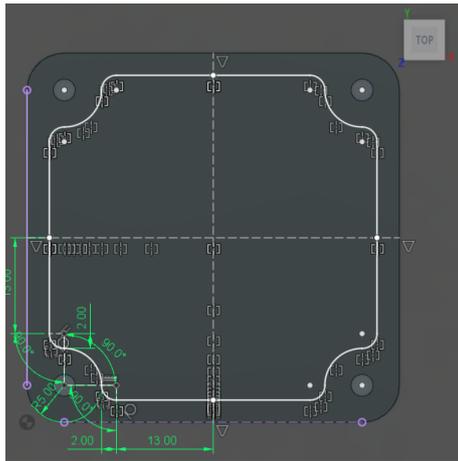
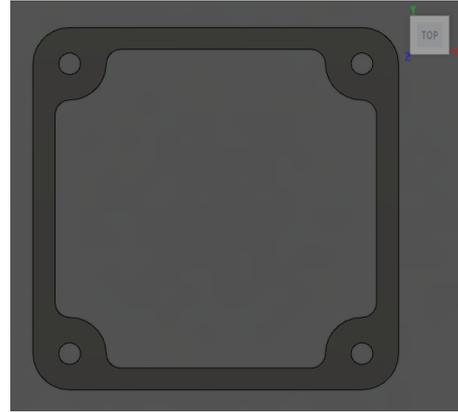


Figura 2.31: Sketch 2n parte

Tal y como se muestra en la *Figura 2.32a*, utilizando el sketch anterior como punto de partida, y haciendo uso de la herramienta de espejo, se replica sobre toda la superficie. Y se utiliza para realizar un agujero pasante en la pieza, *Figura 2.32b*.



(a) Sketch en espejo sobre toda la superficie



(b) Pieza con el corte

Figura 2.32: Paso 1 del diseño de la tapa superior

La pieza resultante se utiliza de guía para cortar otra lámina de metacrilato, cuyo resultado se muestra en la *Figura 2.33*

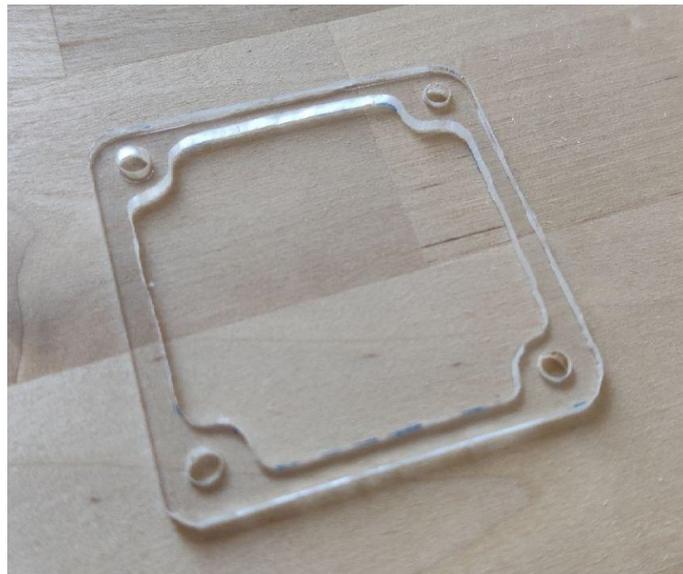


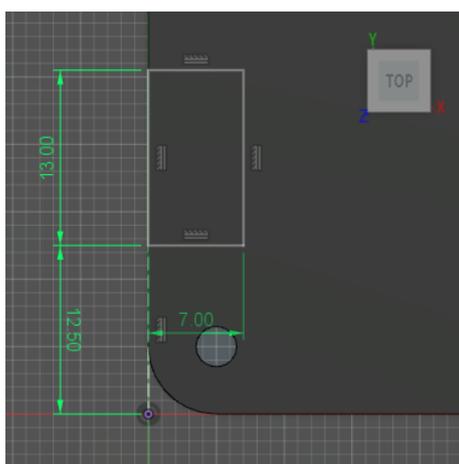
Figura 2.33: Pieza de metacrilato inferior

La tapa superior se forma ensamblando las dos piezas de metacrilato, una sobre la otra con una altura total de 4mm y un hueco interior con una altura de 2mm. De esta forma se asegura que una vez montada, hay espacio para los componentes emplazados en la parte superior de la PCB.

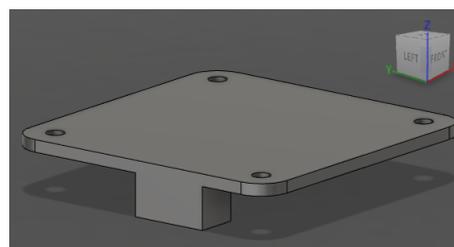
### 2.6.2. Caja contenedora inferior

Para la realización de la parte inferior, primero se ha diseñado un modelo de la PCB con el interruptor. El proceso de realización se detalla a continuación.

Se parte de la pieza de la *Figura 2.29*, y se dibuja el sketch de la *Figura 2.34a* sobre la parte inferior de la misma, las medidas se han obtenido midiendo el interruptor con un pie de rey. Se extruye 7 mm desde la superficie inferior, como se ve en la *Figura 2.34b* obteniéndose así un modelo básico de la PCB.



(a) Sketch sobre superficie inferior



(b) Pieza con extrusión de 7mm

Figura 2.34: Modelo de la PCB con el interruptor

Seguidamente, se empieza con el diseño de la pieza inferior. Primero se extruye el sketch de la *Figura 2.28a* hacia abajo una distancia de 15 mm y realizando fillets de 5mm en las cuatro esquinas. También se practican los agujeros siguiendo el mismo procedimiento que el utilizado para alcanzar la *Figura 2.29*. Se obtiene la pieza de la *Figura 2.35*.

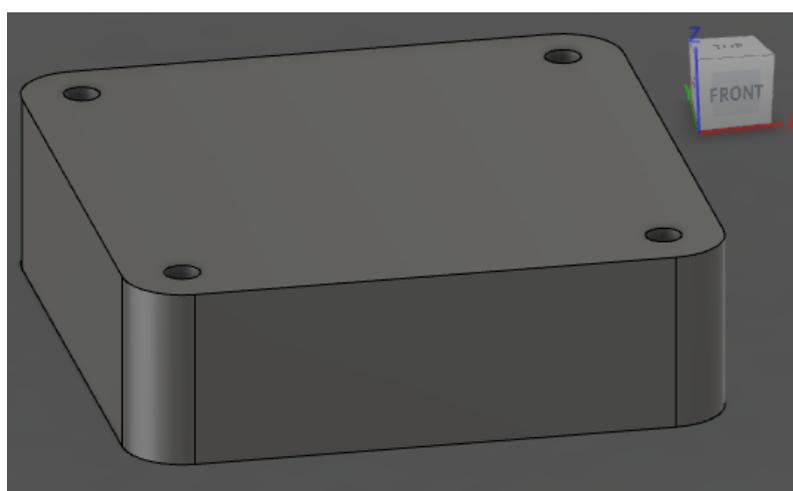


Figura 2.35: Bloque inferior

El siguiente paso consiste en realizar sobre la superficie superior del bloque el sketch de la *Figura 2.32a* para realizar un corte en el bloque de una profundidad de 12mm,

obteniéndose el bloque de la *Figura 2.36*.

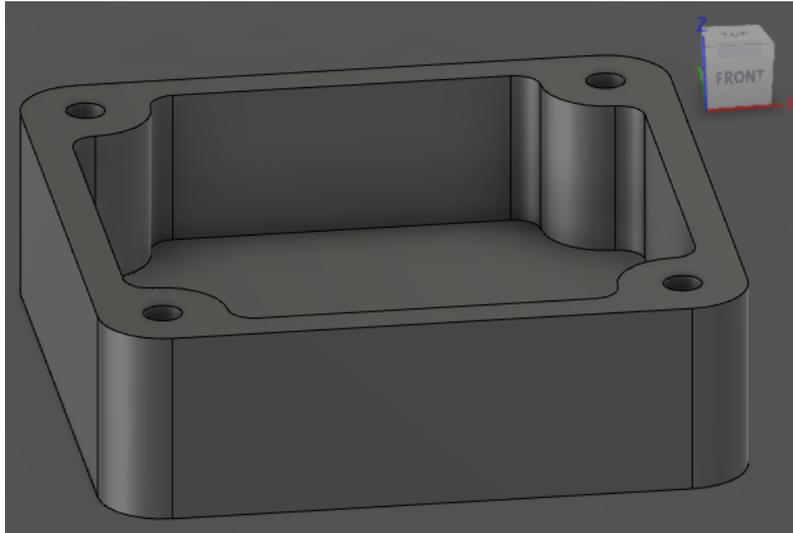


Figura 2.36: Bloque inferior hueco

Ahora, utilizando el modelo de la PCB diseñado al inicio de esta parte y utilizando la herramienta booleana de combinado se elimina la parte del interruptor que intersecta con la caja. De este modo se obtiene el bloque de la *Figura 2.37*.

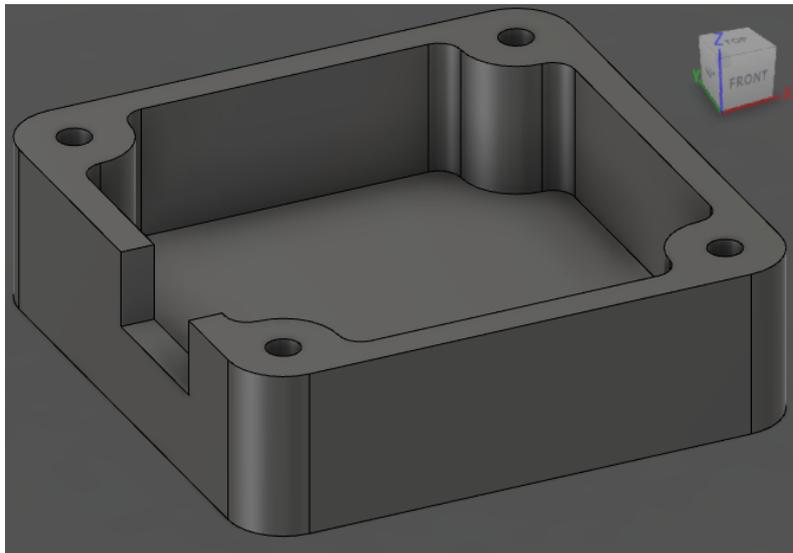
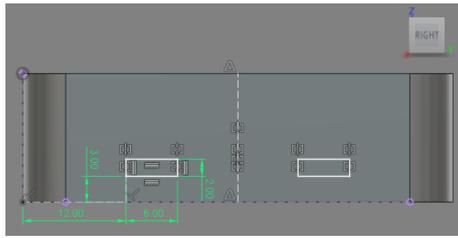
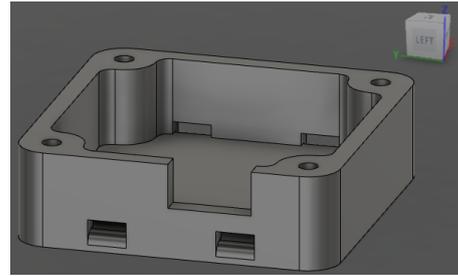


Figura 2.37: Bloque inferior hueco

El último paso consiste en realizar un par de agujeros pasantes que permitan la sujeción de la caja al drone mediante el uso de bridas. Para realizar la operación se realiza el sketch de la *Figura 2.38a* que se utiliza para hacer un corte a lo largo de toda la pieza, obteniendo la pieza final que puede verse en la *Figura 2.38b*.



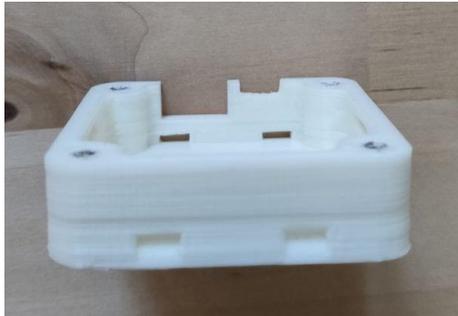
(a) Sketch agujeros pasantes



(b) Parte inferior finalizada

Figura 2.38: Bloque final

Con el diseño finalizado se exporta el cuerpo en formato STL para poder realizar su impresión. El resultado de la misma se observa en la *Figura 2.39*, hay que mencionar que se han realizado algunas muescas con una amoladora para subsanar algunos detalles.



(a) Imagen parte inferior



(b) Imagen parte inferior

Figura 2.39: Caja impresa en 3D

## 3. Validación

En este capítulo se procede a analizar la validez del sistema. Para ello se ha subdividido en dos secciones.

En la primera se comprueba la integridad de la estructura del módulo en sí mismo. Su resistencia, sus acabados y la capacidad de asirse adecuadamente a un dron de manera efectiva así como de no desengancharse a lo largo de la operación.

En la segunda sección se evalúa la visibilidad de la luz del sistema. Para ello, se sitúa el módulo en un punto y se graba desde otro punto conociéndose la distancia de separación entre ambos.

### 3.1. Estructura

El primer paso para la validación del sistema consiste en ensamblar todas las partes que conforman el módulo y comprobar el funcionamiento. Así pues, en la *Figura 3.1*, se muestran la tapa superior, la PCB con sus componentes y batería y la tapa inferior ensamblados.

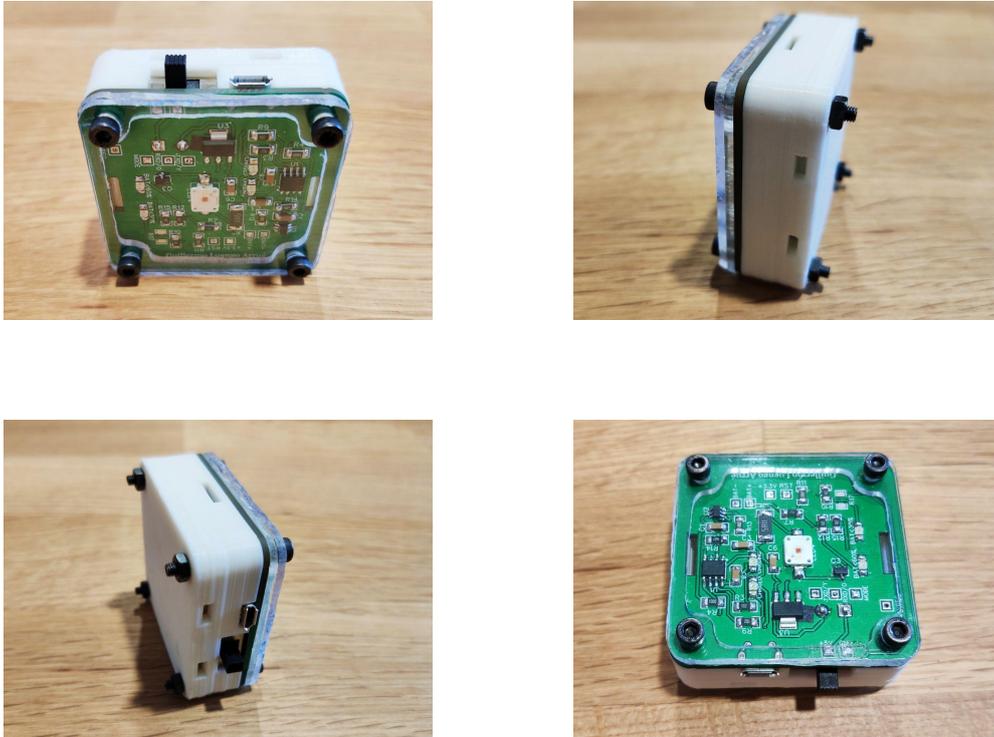


Figura 3.1: Módulo ensamblado

Es esperable que alguno de los módulos del sistema se encuentre en la parte inferior del drone. Para evitar impactos sobre la superficie de la tapa superior, los tornillos sobresalen 3mm de la superficie, de este modo, aterrizar sobre el módulo no causará daños relevantes sobre el mismo.

Para comprobar el sistema de sujeción, se ha montado el módulo en un pequeño drone de carreras, y se ha realizado un vuelo (*Figura 3.2*). En este pequeño vuelo se ha podido certificar visualmente que, gracias al diseño de la tapa superior en metacrilato la luz es visible en un ángulo de 180°. También se ha comprobado que el drone puede despegar y aterrizar sobre el módulo con seguridad ya que las cabezas de los tornillos soportan las fuerzas originadas en los impactos.



Figura 3.2: Módulo montado en drone

## 3.2. Visibilidad

En este apartado se evalúa la visibilidad del sistema a diferentes distancias. Ya que no se tiene acceso a un dron, se utilizan distancias horizontales.

Para su medición se ha hecho uso de Google Maps, herramienta bastante fiable para medir distancias sobre un mapa, aunque no tiene en cuenta las variaciones en altitud. Para la parte visual se ha realizado una grabación y un recorte en los fotogramas en los que los módulos emitían luz.

Pese a los limitados recursos para plasmar la demostración del funcionamiento, ya que los vídeos no se han realizado con cámaras profesionales, se considera que los resultados obtenidos son buenos y suficientes como para plasmar el correcto funcionamiento del sistema.

El sistema ha resultado bastante fiable, pues como se expone a lo largo de este capítulo, haciendo uso de dos módulos emisores de luz y uno para el envío de la señal de sincronización es posible distinguir a simple vista la luz que emiten a una distancia máxima de 1 km.

Así mismo, la sincronización de los módulos es satisfactoria y provoca el efecto en el observador de estar viendo un único punto de luz.

Los emplazamientos de las *Subsecciones 3.2.1* y *3.2.2* se sitúan en el casco urbano de Benaguacil.

### 3.2.1. Distancia: 242m

La primera comprobación se realiza situándose el emisor a una distancia de 242m con respecto del observador (*Figura 3.3a*).

La localización exacta del lugar de la validación se muestra en la *Tabla 3.1*.

Coordenadas y Altitud		
Punto 1	39°35'40.7"N 0°35'51.3"W	174 m
Punto 2	39°35'45.8"N 0°35'43.7"W	140 m

Tabla 3.1: Situación de los puntos

Entre los dos puntos existe una diferencia en altitud de 34 m, esto implica que la distancia real en línea recta es de algo más de 244 m. Esta variación se considera despreciable y se asume que la distancia es la obtenida a través de Google Maps.

Aunque el emplazamiento no está en completa oscuridad la iluminación es bastante baja, contando únicamente con una serie de farolas de baja intensidad. Como se puede ver en la *Figura 3.3b* la luz amarilla del sistema se puede distinguir con facilidad en el cielo oscuro a 242 m.



Figura 3.3: Visibilidad a 242m

### 3.2.2. Distancia: 1km

La segunda comprobación se realiza situándose el emisor a una distancia de 1 km con respecto del observador (*Figura 3.4a*).

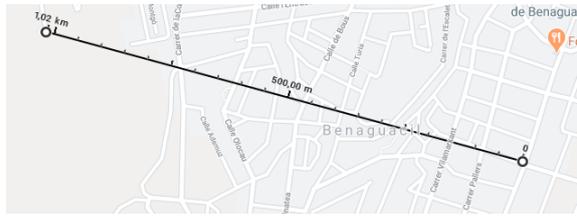
La localización exacta del lugar de la validación se muestra en la *Tabla 3.2*.

Coordenadas y Altitud		
Punto 1	39°35'34.1"N 0°35'51.0"W	219 m
Punto 2	39°35'45.8"N 0°35'43.7"W	111 m

Tabla 3.2: Situación de los puntos

Entre los dos puntos existe una diferencia en altitud de 108 m, esto implica que la distancia real en línea recta es de algo más de 1.005 km. Esta variación, de nuevo, se considera despreciable y se asume que la distancia es la obtenida a través de Google Maps.

En esta ocasión, el emplazamiento es peor que el de la *Subsección 3.2.1* ya que se encuentra más iluminado (*Figura 3.4b*). No obstante, el sistema ofrece una luminosidad suficiente como para ser visto en estas condiciones a 1 km de distancia, tal y como se ve en la figura.



(a) Localización



(b) Visibilidad

Figura 3.4: Visibilidad a 1km

## 4. Conclusión

En el presente Trabajo Final de Grado, se ha elaborado de forma efectiva un sistema de luces anticolidión independiente con capacidades de comunicación inalámbrica, utilizando como base el módulo ESP12E y el LED Golden Dragon de OSRAM.

El proyecto se ha realizado de forma satisfactoria cumpliendo con los objetivos esperados de obtener un sistema capaz de proporcionar luces anticolidión que parpadeen de forma simultánea, con una buena adaptabilidad ya que los módulos del mismo pueden crecer hasta un total de 8. Así mismo, los módulos son más económicos que los ofrecidos en el mercado y, a su vez, para anclarlo al dron únicamente es necesario el uso de bridas obviando así compras de sistemas de sujeción adicionales e innecesarios.

En cuanto a las limitaciones, hay que recordar que en el sistema aquí diseñado uno de los módulos crea una red WiFi a la cual se conectan el resto de los módulos. Es por esto que en el hipotético caso de encontrarse dos usuarios haciendo uso de este sistema en el mismo espacio, los módulos deberían hacer uso de redes WiFi con distinto nombre. Problema que es, por otro lado, fácilmente resoluble si se posee el presente trabajo y unos conocimientos medios de programación.

Por otro lado, como proyección a este trabajo caben posibles ampliaciones. Una mayor miniaturización del mismo es posible si se opta por escoger componentes de un menor tamaño. El tamaño estándar de los componentes usados aquí es el 1206, para facilitar las tareas de soldadura pero sería posible soldar a mano componentes de menor tamaño que se traducirían en una reducción del dispositivo.

Así mismo, puede optarse por realizar una modificación de uno de los módulos para utilizarlo individualmente en un dron de pequeño tamaño, como las opciones ofertadas en el mercado, vistas en la *Subsección 1.3*.

También, como revisión, se plantea la resolución de la limitación arriba planteada, mediante un interruptor que seleccione y cree distintas redes a crear/conectar dependiendo de la posición o mediante la implementación de un código que permita escanear las redes que existen y actuar en consecuencia.

Para terminar, valorando la realización del presente Trabajo Final de Grado como herramienta de aprendizaje, puede decirse que en su realización se han adquirido y afianzado diversos conocimientos tanto teóricos como prácticos en el ámbito de la electrónica, de la informática y de la investigación. Ha complementado en gran medida a la formación adquirida a lo largo del Grado en Ingeniería Aeroespacial y, también, ha servido como acercamiento al mundo de la fabricación de prototipos de un modo que no había sido posible a lo largo del Grado.

## 5. Presupuestos

Como parte final del trabajo, se presentan a continuación una aproximación de los costes asociados al mismo.

Los costes se han dividido en dos secciones. Primero se presentan los costes materiales, estos costes son aquellos relacionados a los componentes que forman parte del sistema, así como el coste derivado de la fabricación del prototipo de la PCB y de la impresión 3D de la caja. Por otro lado, se presentan los costes relacionados con la mano de obra: búsqueda de componentes, diseño de circuitos, diseño de PCB, tiempo de soldadura, realización de código...

Así, el conjunto de lo obtenido en las dos secciones ofrece una aproximación al presupuesto real del proyecto desarrollado.

Hay que tener en cuenta que el coste asociado al proyecto se ve afectado por distintos factores. Como por ejemplo, la disponibilidad de una impresora 3D para poder realizar la impresión de la caja sin encargarla a una empresa externa, o la fluctuación del precio en el mercado de los componentes electrónicos utilizados. Del mismo modo, la situación actual de emergencia provocada por la COVID-19 ha encarecido el precio de los gastos de envío.

## 5.1. Materiales

Para la evaluación del coste en materiales relacionado con el prototipo del presente trabajo se adjuntan, como justificantes, las facturas proporcionadas por los suministradores. En caso de los componentes estos han sido adquiridos a la compañía LCSC [14]. Por otro lado, la fabricación de la PCB ha sido encargada a la compañía JLCPCB [11].

Al precio total de los componentes hay que añadir 4.66\$ de pago por el envío. Para la conversión de dólares a euros se ha utilizado el siguiente factor de conversión: 1\$ = 0,89€

Por otra parte, no se añaden gastos que podrían entrar en esta sección, entre estos caben la estación de soldadura, el estaño, el multímetro o la fuente de alimentación utilizada para alimentar el sistema durante las pruebas. Tampoco se computa el coste del LED OSRAM, ya que fue proporcionado por la universidad y, actualmente, se encuentra descatalogado.

Así pues, en la *Tabla 5.1* se encuentra resumido el total del coste de la parte material del proyecto.

Producto [con envío]	Precio [€]
Componentes	23.83
PCB	10.63
Batería LiPo 350mAh	6.5
Tornillería	5.8
Total	46.76

Tabla 5.1: Coste total en materiales



Holding company:  
**SHENZHEN LICHUANG E-COMMERCE CO.,LTD**  
 Subsidiary Company:  
**SHENZHEN LCSC ELECTRONICS TECHNOLOGY CO., LTD.**  
**LCSC ELECTRONICS TECHNOLOGY (HK) LIMITED**  
 VAT: 91440300MA5EDMX26A  
 FB, Tianjian Building, No. 7 Shangbao Road,  
 Futian District, Shenzhen China 518304  
 +86-755-83210457

Invoice No.	Invoice Date
lcsc-202002105FTZ	02/06/2020
Lcsc Order No.	Order Date
202002105FTZ	10/02/2020
Purchase Order No.	Air Waybill No.
	PQ56GB07313915501461808

Shipping Method	Terms of Delivery	Payment Method	Terms of Payment
Correos de Espana	DAP	stripe	100% Prepaid

Bill to	Ship to
ATTN:GuillermoLuengo C/ de la m [REDACTED] [REDACTED] Valencia [REDACTED] Spain 6079 [REDACTED]	ATTN:GuillermoLuengo C/ de la m [REDACTED] [REDACTED] Valencia [REDACTED] Spain 6079 [REDACTED]

LN	LCSC P/N	MFR. P/N	Customer #	DESC.	Order QTY.	Shipped QTY.	Unit Price (USD)	EXT. (USD)
1	C1469	1206W4J0102T5E		1K ±5% 1/4W ±100ppm/°C 1206 Chip Resistor - Surface Mount RoHS	100	100	0.00176	0.18
2	C1489	1206W4J0103T5E		10K ±5% 1/4W ±100ppm/°C 1206 Chip Resistor - Surface Mount RoHS	100	100	0.001758	0.18
3	C1846	1206B103K500NT		10nF ±10% 50V X7R 1206 Multilayer Ceramic Capacitors MLCC - SMD/SMT RoHS	20	20	0.015937	0.32
4	C7182	TAJA226K006RNJ		22uF ±10% 6.3V 3 Ω @ 100kHz -5 5°C ~ +125°C CASE-A_3216 Tantalum Capacitors RoHS	20	20	0.085527	1.71
5	C8678	SS34		40V 3A 550mV @ 3A SMA(DO-214A C) Schottky Barrier Diodes (SBD) RoHS	20	20	0.026461	0.53
6	C10418	C10418		USB - Micro B Receptacle USB 2.5 SMD USB Connectors RoHS	10	10	0.057314	0.57
7	C16581	TP4056-42-ESOP8		ESOP-8 PMIC - Battery Management RoHS	10	10	0.122727	1.23
8	C17901	1206W4F1000T5E		100 ±1% 1/4W ±200ppm/°C 1206 Chip Resistor - Surface Mount RoHS	100	100	0.002369	0.24
9	C17944	1206W4F2001T5E		2K ±1% 1/4W ±100ppm/°C 1206 Chip Resistor - Surface Mount RoHS	100	100	0.002314	0.23

10	C26047	1206W4J0752T5E	7.5K ±5% 1/4W ±100ppm/°C 1206 Chip Resistor - Surface Mount RoHS	50	50	0.0034	0.17
11	C27442	SMRH74-220MT	22uH ±20% SMD Power Inductors RoHS	8	8	0.163636	1.31
12	C32254	F58205	2 N Channel(Double) 20V 6A 1.2V @ 250uA 28 mΩ @ 4A,4.5V 1W SOT-23 -6 MOSFET RoHS	10	10	0.104545	1.05
13	C82736	FS312F-G	SOT-23-6 Battery Protection ICs RoHS	10	10	0.09721	0.97
14	C84817	MT3608	Step-Up Positive Adjustable 1.12MHz SOT-23-6 DC-DC Converters RoHS	10	10	0.079535	0.8
15	C90146	CL31A226KOHNNNE	22uF ±10% 16V X5R 1206 Multilayer Ceramic Capacitors MLCC - SMD/SMT RoHS	10	10	0.044346	0.44
16	C122850	AO3400	N Channel 30V 5.8A 1.5V @ 250uA 35 mΩ @ 5.8A,10V SOT-23(SOT-23-3) MOSFET RoHS	10	10	0.040628	0.41
17	C147258	CR2512F5R1E04S	5.11 ±1% 1W ±400ppm/°C 2512 Chip Resistor - Surface Mount RoHS	20	20	0.026803	0.54
18	C175223	CRH2512F2R20E04Z	2.2 ±1% 2W ±400ppm/°C 2512 Chip Resistor - Surface Mount RoHS	10	10	0.050489	0.5
19	C229755	AC1206FR-0766R5L	66.5 ±1% 1/4W ±100ppm/°C 1206 Chip Resistor - Surface Mount RoHS	50	50	0.007761	0.39
20	C284938	LTST-C21KGKT	Green 567.5~576.5nm SMD,3.2x1.6x1.1mm Light Emitting Diodes (LED) RoHS	15	15	0.063412	0.95
21	C284939	LTST-C21KRKT	Red 631nm SMD,3.2x1.6x1.1mm Light Emitting Diodes (LED) RoHS	20	20	0.063412	1.27
22	C319015	SS-12D06L5	SPDT 3A 250V On-On DIP Toggle Switches RoHS	9	9	0.272727	2.45
23	C347222	AMS1117-3.3	Fixed 10V 3.3V 1A, SOT-223 Low Dropout Regulators(LDO) RoHS	10	10	0.034944	0.35
24	C398257	C1206X104K050T	100nF ±10% 50V X7R 1206 Multilayer Ceramic Capacitors MLCC - SMD/SMT RoHS	50	50	0.013112	0.66
Merchandise		Freight		Total Invoiced		Amount Paid	
US\$17.45		US\$4.66		US\$22.11		USD US\$22.11	
Reference Currency				Currency Reference Rate			
EUR				0.91355			

THIS ORDER IS SUBJECT TO ALL TERMS AND CONDITIONS DISPLAYED AT: LCSC.COM

Factura de los componentes


**JiaLiChuang (HongKong) Co., Limited**

JLCPCB  
 CSDONG3LOU FA HUO BU TONG FU YU GONG YE  
 YUAN PING DE FU DE GANG.LONG GANG DIST.  
 SHENZHEN 518000  
 The People's Republic of China

**Telephone:** +86 755 23919769  
 support@jlcpcb.com  
 https://jlcpcb.com

**Invoice No:** 2703159A20200603012431  
**Invoice Date:** 05/04/2020  
**Batch No:** W202004050058709

**Total Value:**  
**€10.63**

**Ship To:**

Guillermo Luengo  
 C/ de la n  
 Valencia  
 SPAIN  
  
 guillel90@gmail.com  
 6075  
 VAT No:

**Billing Address:**

Guillermo Luengo  
 C/ de la n  
 Valencia  
 SPAIN  
  
 guillel90@gmail.com  
 6075

Description of Goods	HS Code	Order Number.	Qty.	Unit Value	Total Value
PCB Samples	8534009000	Y2	10	€0.4470	€4.47
<b>SUB-TOTAL</b>					€4.47
<b>FREIGHT COST</b>					€6.16
<b>INVOICE TOTAL</b>					<b>€10.63</b>

Attachment

Remarks

Factura de la PCB

## 5.2. Mano de obra

En cuanto al coste derivado de la mano de obra aquí se tiene en cuenta únicamente el tiempo invertido por el alumno en la realización del sistema desde la idea hasta la concepción del prototipo final, pasando por las fases de: investigación, búsqueda de componentes, aprender lenguaje de programación, comprobación de errores en circuitos, etcétera.

Aunque las horas no se han contabilizado rigurosamente desde el inicio del trabajo se puede realizar una aproximación bastante fiable en cuanto al tiempo invertido en el mismo.

El precio de la hora de un graduado en Ingeniería Aeroespacial se ha estimado asumiendo un salario mensual de 2500 €, resultando en 15.625 €/hora. Por tanto, en la *Tabla 5.2* se muestra el importe debido a este apartado.

Personal	Precio por horas [€/h]	Horas [h]	Coste [€]
Ingeniero Aeroespacial	15.625	250	3906.25

Tabla 5.2: Coste en personal

### 5.3. Coste total

Se ha mostrado el desglose de la estimación de los costes tanto materiales como personales atribuidos a la realización del proyecto. El último paso corresponde, pues, en ver el coste total del conjunto así como en la adición de un 21 % en impuestos derivados del Impuesto al Valor Agregado *IVA* que se aplica al precio del Ingeniero Aeroespacial en cuestión.

El coste total se muestra en la *Tabla 5.3*.

Concepto	Base [€]	Importe [€]
Material		23.83
Mano de obra		3906.25
IVA	3906.25	820.31
Total		4750.39

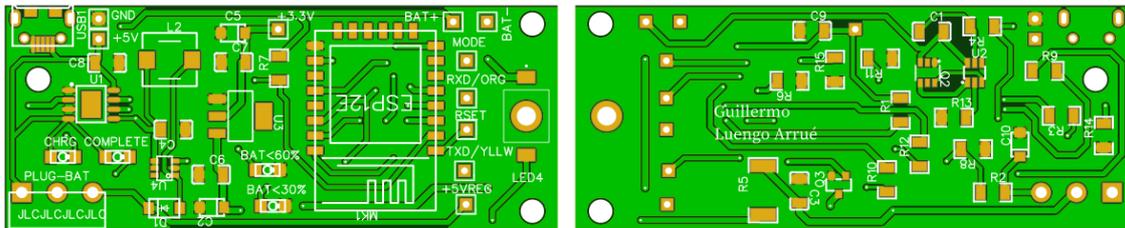
Tabla 5.3: Coste total

# A. PCB V0

En este apéndice se presenta un diseño de PCB que, si bien fue desarrollado completamente y es funcional, no ha sido implementado en el sistema por tener un diseño en planta que no permite implementarlo en una caja de soporte de forma óptima.

No obstante, debido al pequeño tamaño y no ser necesario que el sistema soporte impactos, es posible hacer uso de este diseño envolviendo tanto la PCB como la batería en algún tipo de envoltura plástica.

Hay que remarcar que la única diferencia entre esta placa con respecto de la presentada en el cuerpo del trabajo radica en las dimensiones de la misma y, obviamente, la distribución de los componentes. Esta PCB tiene unas dimensiones de 73x30 mm.

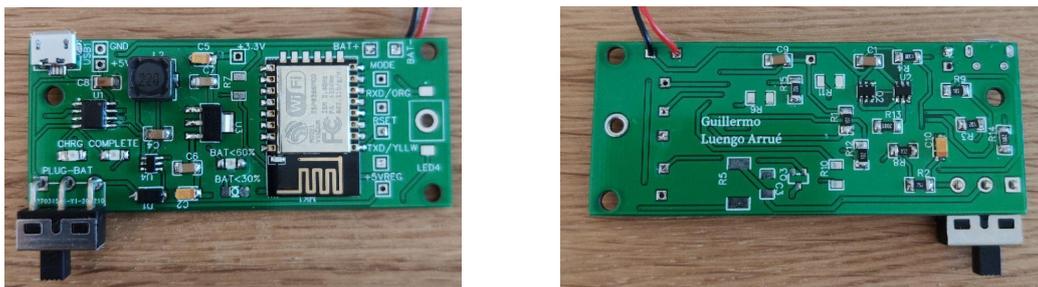


(a) Parte frontal

(b) Parte trasera

Figura A.1: Imagen de la PCB descartada (diseño)

Así mismo se adjunta imagen de una placa con los elementos indispensables para su funcionamiento soldados.



(a) Parte frontal

(b) Parte trasera

Figura A.2: Imagen de la PCB descartada

## B. Fichas de datos técnicos

En este apartado se recopilan los enlaces a los ficheros de datos técnicos de los respectivos componentes utilizados en el proyecto.

Cargador de batería TP4056

IC protección de batería FS312F-G

Mosfet dual canal N FS8205

Convertidor de voltaje MT3608

Regulador de voltaje AMS1117

Modulo WiFi ESP12-E

LED OSRAM Mod: LY W5SM-HZJZ-46-1

Mosfet canal N AO3400

System on Chip ESP8266

LED Osram GOLDEN DRAGON



# Bibliografía

- [1] España. *Real Decreto 1036/2017, de 15 de diciembre, por el que se regula la utilización civil de las aeronaves pilotadas por control remoto*. 2017.
- [2] AESA. *Medios Aceptables de Cumplimiento, apéndice O*. 2019.
- [3] LUMECUBE. *STROBE - ANTI-COLLISION LIGHTING FOR DRONES*. [Web; accedido el 17-05-2020]. 2020. URL: <https://lumecube.com/collections/best-sellers/products/strobe#>.
- [4] litra. *LitraTorch 2.0™ Drone Edition*. [Web; accedido el 17-05-2020]. 2020. URL: <https://litra.com/collections/drones/products/litratorch-drone-edition>.
- [5] ulanzi. *LED on DJI Drone Strobe Light*. [Web; accedido el 17-05-2020]. 2020. URL: <https://www.ulanzi.com/products/ulanzi-dr-01-drone-strobe-light>.
- [6] Arduino. *Arduino - Software*. [Web; accedido el 18-07-2019]. 2020. URL: <https://www.arduino.cc/en/Main/Software>.
- [7] Tony R Kuphaldt. “Lessons In Electric Circuits, Volume I-DC”. En: *Vol. Fifth Edition. Open Book Project* (2006).
- [8] EasyEda. *EasyEDA – Simulador de circuitos y diseño de circuitos impresos online*. [Web; accedido el 20-05-2020]. 2020. URL: <https://easyeda.com/>.
- [9] IPC. *2221A: Generic Standard on Printed Board Design*. 1.<sup>a</sup> ed. IPC, 2003.
- [10] Advanced Circuits. *Printed Circuit Board Trace Width Tool | Advanced Circuits*. [Web; accedido el 17-02-2020]. 2020. URL: <https://www.4pcb.com/trace-width-calculator.html>.
- [11] JLCPCB. *PCB Prototype & PCB Fabrication Manufacturer*. [Web; accedido el 19-03-2020]. 2020. URL: <https://jlcpcb.com/>.
- [12] Datech Solutions. *Autodesk Fusion 360 | La nueva herramienta de CAD*. [Web; accedido el 7-04-2020]. 2020. URL: <https://www.datech.es/productos-autodesk/autodesk-fusion-360/>.
- [13] Datech Solutions. *M3D: 3D Printers, 3D Printing Filaments, 3D Printer Parts*. [Web; accedido el 21-04-2020]. 2020. URL: <https://store.printm3d.com/>.
- [14] LCSC. *LCSC Electronic Components Online | LCSC.com*. [Web; accedido el 9-01-2020]. 2020. URL: <https://lcsc.com/>.