



INGENIA SE | CURSO 2022-2023

# Informe de Pruebas Final

**Equipo Hell-ix**

**23 de mayo de 2023**



# Índice general

<b>1. Introducción</b>	<b>3</b>
<b>2. Descripción del sistema probado</b>	<b>3</b>
2.1. Hardware	3
2.2. Software	4
2.3. Integración	5
<b>3. Descripción de las políticas de pruebas</b>	<b>5</b>
<b>4. Pruebas de verificación</b>	<b>6</b>
4.1. Diseño de las pruebas	6
4.1.1. Prueba 1	6
4.1.2. Prueba 2	6
4.1.3. Prueba 3	6
4.1.4. Prueba 4	7
4.1.5. Prueba 5	7
4.1.6. Prueba 6	7
4.1.7. Prueba 7	8
4.1.8. Prueba 8	8
4.1.9. Prueba 9	8
4.2. Ejecución y resultados de las pruebas	8
4.2.1. Prueba 1	8
4.2.2. Prueba 2	9
4.2.3. Prueba 3	9
4.2.4. Prueba 4	10
4.2.5. Prueba 5	10
4.2.6. Prueba 6	11
4.2.7. Prueba 7	11
4.2.8. Prueba 8	12
4.2.9. Prueba 9	12
<b>5. Limitaciones identificadas y resultados de la verificación y la validación</b>	<b>13</b>
5.1. Limitaciones identificadas	13
5.1.1. Incompatibilidad del AI Deck	13
5.1.2. Incompatibilidad del LED-ring deck con el Flow deck v2	13
5.2. Conclusiones de verificación	14
5.3. Conclusiones de validación	15

## 1. Introducción

En el presente documento se detallan el sistema probado y los resultados de las pruebas del sistema con el fin de verificar que el sistema cumple con los requisitos y las especificaciones fijadas y validar que puede utilizarse como inicialmente se concibió para cumplir con las necesidades del cliente.

## 2. Descripción del sistema probado

A continuación se realiza una breve descripción de los componentes e interfaces del sistema que han sido testeados. Estos incluyen hardware, software y su integración.

### 2.1. Hardware

El sistema probado consiste en el dron “Crazyflie 2.1”, el cual fue utilizado como plataforma de pruebas. El dron estaba equipado con el “Flow deck v2”, un módulo que proporciona capacidades de detección y seguimiento de movimiento mediante una cámara y un sensor de flujo óptico. Además, se utilizó la antena “Crazyradio 2.4GHz for Crazyflie USB Radio w/ Antenna” para establecer la comunicación inalámbrica entre el dron y el ordenador desde el cual se enviaban los comandos de control. Por último, se utilizó el módulo “AI deck 1.1” para las funcionalidades relacionadas con la visión del dron y el módulo “LED-ring deck” con fines estéticos.

El hardware utilizado en las pruebas fue el siguiente:

- **Dron Crazyflie 2.1:** El dron Crazyflie 2.1 es conocido por su diseño duradero y su facilidad de montaje sin necesidad de soldadura. Además, cuenta con soporte para expansiones de módulos con detección automática, lo que permite ampliar sus capacidades y funcionalidades según las necesidades del usuario.
- **Flow deck v2:** El Flow deck v2, que estaba instalado en el dron, agrega capacidades de detección y seguimiento de movimiento. Este módulo incluye una cámara y un sensor de flujo óptico (PMW3901), que permiten al dron comprender su movimiento en cualquier dirección y medir con precisión la distancia al suelo utilizando un sensor de tiempo de vuelo (VL53L1x). Estas capacidades proporcionan una plataforma de vuelo estable y la posibilidad de programar trayectorias en función de distancias y movimientos específicos.
- **Crazyradio 2.4GHz:** La antena Crazyradio 2.4GHz for Crazyflie USB Radio w/ Antenna se utilizó para establecer la comunicación inalámbrica entre el dron Crazyflie 2.1 y el ordenador desde el cual se enviaban los comandos de control. Esta antena cuenta con un potente amplificador de señal (20 dBm) y una baja latencia, lo que garantiza una transmisión confiable de los comandos de control y una comunicación estable entre el dron y el ordenador.
- **AI Deck 1.1:** El AI Deck es un módulo ultraligero de 30 gramos que permite la computación a bordo del dron. Con el procesador IoT GAP8 y la cámara ULP, el Crazyflie puede capturar y analizar su entorno. La versión 1.1 ofrece capacidades de inteligencia artificial para una navegación autónoma, con conectividad WiFi y control remoto.

- **LED-ring deck:** El Crazyflie 2.X LED-ring expansion board es un módulo que cuenta con 12 LEDs RGB hacia abajo que permiten crear patrones personalizados controlados desde el ordenador. Además, tiene dos LEDs frontales adicionales que pueden encenderse y apagarse, funcionando como faros.

Estos componentes se montaron siguiendo los manuales de instrucción disponibles en la página oficial del dron <https://www.bitcraze.io/>. Además, se utilizaron los ordenadores personales del equipo Hell-ix, las instalaciones disponibles de la Universidad y los obstáculos prestados por el Centro de Automática y Robótica de la ETSII.

## 2.2. Software

El sistema probado del dron Crazyflie 2.1 incluye el uso de software externo obtenido de Internet, como las librerías, y de desarrollo propio por el equipo de Hell-ix, que se centra en el uso de estas librerías. El software nos permite comunicarnos con el dron, controlarlo, analizar imágenes para la detección de obstáculos y generar trayectorias predefinidas para el dron. A continuación, se describe el software utilizado durante las pruebas:

- **Librerías Crazyflie:** Se emplearon las librerías de Python proporcionadas por el repositorio <https://github.com/bitcraze/crazyflie-lib-python>. Estas librerías permiten establecer la comunicación con el dron y controlarlo con un ordenador externo a través de la antena Crazyradio 2.4GHz.
- **Código de generación de trayectorias:** Se desarrolló un código personalizado apoyado en librerías de MATLAB, alojado en el repositorio "<https://github.com/Ingenia-SE/Hell-ix>", para la generación de trayectorias en función de unos obstáculos predefinidos. El código genera un archivo en formato CSV con las coordenadas que el dron debe seguir durante las pruebas de obstáculos y la prueba freestyle.
- **Algoritmo de visión:** Se desarrolló un código personalizado apoyado en la librería de OpenCV disponible para Python 3.x, alojado en el repositorio "<https://github.com/Ingenia-SE/Hell-ix>", para el análisis de imágenes del dron en tiempo real. Este código recibe una imagen y detecta los obstáculos en ella. Además, modifica la trayectoria previamente generada para centrar su camino en la puerta y evitar colisiones.
- **Software de simulación:** Se utilizó el simulador Webots para realizar las pruebas del dron en un entorno simulado. Se desarrolló un controlador para el dron simulado y se integró con el código de generación de trayectorias.
- **Entorno de desarrollo:** Se utilizó un entorno de desarrollo integrado (IDE), que era Visual Studio Code, para escribir y ejecutar el código de control del dron y las pruebas. Este IDE ofrece herramientas de depuración y facilidades para el desarrollo de software.
- **Grabación de vídeo:** Se utilizó un software de grabación de vídeo para registrar las pruebas realizadas. Este software permitió capturar las acciones del dron y su comportamiento durante las pruebas de manera visual, lo cual facilita el análisis posterior de los resultados obtenidos. A parte de los dispositivos de grabación personales de los miembros del equipo, se pudieron realizar grabaciones de alta calidad gracias a las instalaciones que nos cedieron desde el CAR, que contaban con una sala preparada para el vuelo de drones y cámaras de vídeo para grabar las pruebas.
- **Sistema operativo:** Las pruebas se llevaron a cabo en un entorno con un sistema operativo compatible que soportara las librerías y herramientas utilizadas durante el proceso de prueba. El sistema operativo principal utilizado fue Windows, aunque fue necesario

instalar Linux Ubuntu para ciertas tareas.

## 2.3. Integración

La integración del sistema probado del dron Crazyflie 2.1 se llevó a cabo mediante la combinación y funcionamiento conjunto de los componentes de hardware y software mencionados anteriormente.

El dron Crazyflie 2.1, equipado con el Flow deck v2 y la antena Crazyradio 2.4GHz, se utilizó como la plataforma principal durante las pruebas. Para establecer la comunicación inalámbrica entre el dron y el ordenador, se utilizó la antena Crazyradio 2.4GHz y las librerías Crazyflie de Python. Estas librerías proporcionaron los medios para establecer y mantener una conexión confiable y estable con el dron.

El código personalizado desarrollado para la generación de trayectorias en simulación, almacenado en el repositorio "<https://github.com/Ingenia-SE/Hell-ix>", permitió definir obstáculos predefinidos en un entorno simulado y generar un archivo CSV con las coordenadas para las pruebas de obstáculos y freestyle. Este código se integró con las librerías Crazyflie y se ejecutó desde el mismo entorno de desarrollo utilizado para el control del dron.

Durante la integración surgieron numerosas dificultades que se fueron solventando a medida que avanzaba el proyecto. Sin embargo, dos incompatibilidades entre los módulos de hardware del dron no se pudieron resolver. Estas incompatibilidades se detallan en el capítulo 5.1 Limitaciones identificadas.

## 3. Descripción de las políticas de pruebas

Debido a la imposibilidad de realizar pruebas de manera exhaustiva en el sistema completo, se define a continuación el alcance de las pruebas que se han considerado.

- **Cobertura de funcionalidades esenciales:** Se probaron las funcionalidades esenciales del sistema. Esto incluye las acciones básicas y fundamentales que el dron debe ser capaz de realizar, como el despegue, el aterrizaje, el vuelo estacionario y el control de la altitud. Estas funcionalidades fueron sometidas a pruebas rigurosas para verificar su correcto funcionamiento.
- **Pruebas de comandos y entradas:** Se verificó la respuesta del sistema a diferentes comandos y entradas. Se evaluaron tanto los comandos válidos como los inválidos para asegurar un manejo adecuado de situaciones de entrada inesperadas o incorrectas. Esto incluye la prueba de movimientos direccionales, cambios de velocidad, cambios de altitud, entre otros comandos relevantes para el control del dron.
- **Pruebas de detección y seguimiento de movimiento:** Dado que el sistema cuenta con capacidades de detección y seguimiento de movimiento, se realizaron pruebas específicas para evaluar su precisión y fiabilidad. Se diseñaron escenarios de movimiento en los que el dron debía seguir trayectorias predefinidas y se evaluó su capacidad para mantenerse en ruta y ajustar su posición según los estímulos externos.
- **Pruebas de integración de hardware y software:** Dado que el sistema está compuesto por hardware (dron, módulos) y software (librerías, código de control), se llevaron a cabo pruebas de integración para verificar la correcta interacción y comunicación entre estos componentes. Se evaluaron aspectos como la detección de hardware, la comunicación

inalámbrica, la transmisión de comandos y la respuesta del sistema en conjunto.

- **Pruebas de rendimiento y estabilidad:** Se realizaron pruebas para evaluar el rendimiento y la estabilidad del sistema en diferentes escenarios y condiciones. Se evaluaron aspectos como la duración de la batería, la estabilidad en vuelo y la capacidad de recuperación ante situaciones inesperadas.

Estas políticas de pruebas se diseñaron para garantizar que el sistema cumpla con los requisitos establecidos y que pueda realizar las pruebas de la competición de drones.

## 4. Pruebas de verificación

Las pruebas de verificación tienen por objeto verificar el cumplimiento de los requisitos del sistema. Es decir, verificar que el sistema se comporta de manera adecuada en los casos de uso que reflejan su uso esperado. A continuación se detalla el diseño de estas pruebas, su preparación y ejecución y los resultados obtenidos.

### 4.1. Diseño de las pruebas

#### 4.1.1. Prueba 1

- Nombre: Prueba de vuelo controlado
- Descripción: Esta prueba tiene como objetivo verificar si el dron es capaz de volar de manera controlada.
- Requisitos a verificar:
  - El dron debe ser capaz de mantener una altitud específica dentro de un rango de tolerancia de  $\pm 10$  cm.
  - El dron debe mantener un patrón de vuelo estable con desviaciones mínimas de inclinación, balanceo y guiñada de la trayectoria planificada, dentro de un rango de tolerancia de  $\pm 5$  grados.
  - El dron debe ser capaz de seguir una trayectoria de vuelo predefinida con una desviación mínima, dentro de un rango de tolerancia de  $\pm 20$  cm.

#### 4.1.2. Prueba 2

- Nombre: Prueba de capacidad de carga
- Descripción: Esta prueba tiene como objetivo verificar si el dron es capaz de levantar su propio peso más los componentes montados.
- Requisitos a verificar:
  - El dron debe ser capaz de proporcionar la potencia necesaria a los motores para levantar su propio peso más los componentes montados.

#### 4.1.3. Prueba 3

- Nombre: Prueba de control manual del GCS

- Descripción: Esta prueba tiene como objetivo verificar si el Sistema de Control en Tierra (GCS) puede manipular el dron manualmente en caso de amenazas de seguridad o eventos inesperados.
- Requisitos a verificar:
  - El GCS debe ser capaz de controlar el movimiento del dron en diferentes direcciones, incluyendo adelante, atrás, izquierda, derecha, arriba y abajo.
  - El GCS debe ser capaz de ajustar la velocidad y altitud del dron en respuesta a las señales del control remoto.

#### 4.1.4. Prueba 4

- Nombre: Prueba de control automático del GCS
- Descripción: Esta prueba tiene como objetivo verificar si el GCS puede controlar el movimiento del dron de forma automática.
- Requisitos a verificar:
  - El dron debe estar equipado con un sistema de control automático que pueda ser accedido y controlado de forma remota a través de un dispositivo de control remoto.

#### 4.1.5. Prueba 5

- Nombre: Prueba de retroalimentación del estado del dron
- Descripción: Esta prueba tiene como objetivo verificar si el GCS puede proporcionar información en tiempo real sobre el estado del dron.
- Requisitos a verificar:
  - El GCS debe ser capaz de mostrar información sobre la ubicación, altitud, vida útil de la batería y posibles errores o fallos del dron.
  - La retroalimentación del estado del dron debe ser fácilmente accesible y claramente visible en el dispositivo de control remoto.

#### 4.1.6. Prueba 6

- Nombre: Prueba de procedimientos de despegue y aterrizaje automáticos
- Descripción: Esta prueba tiene como objetivo verificar si el dron está diseñado con procedimientos automáticos de despegue y aterrizaje.
- Requisitos a verificar:
  - El procedimiento de despegue automático debe permitir que el dron despegue desde una posición estacionaria sin requerir la intervención manual del operador.
  - El procedimiento de aterrizaje automático debe permitir que el dron aterrice de manera segura y controlada sin requerir la intervención manual del operador.

#### 4.1.7. Prueba 7

- Nombre: Prueba de vuelo estacionario
- Descripción: Esta prueba tiene como objetivo verificar si la batería del dron dura el tiempo necesario para realizar la competición.
- Requisitos a verificar:
  - La batería instalada en el dron debe ser lo suficientemente grande como para asegurar que se complete la competición.

#### 4.1.8. Prueba 8

- Nombre: Prueba de trayectoria avanzada
- Descripción: Esta prueba tiene como objetivo verificar si el dron es capaz de seguir una trayectoria compleja y alcanzar una posición determinada.
- Requisitos a verificar:
  - El dron debe ser capaz de seguir una posición objetivo.

#### 4.1.9. Prueba 9

- Nombre: Prueba de análisis de imagen
- Descripción: Esta prueba tiene como objetivo verificar si el algoritmo de procesamiento de imagen es capaz de reconocer obstáculos.
- Requisitos a verificar:
  - El software debe ser capaz de procesar imágenes.

### 4.2. Ejecución y resultados de las pruebas

#### 4.2.1. Prueba 1

##### **Procedimiento de la prueba:**

1. Verificar que el dron esté en condiciones de vuelo y configurado correctamente.
2. Establecer una altitud específica como objetivo de vuelo
3. Observar el vuelo del dron y registrar cualquier desviación de altitud dentro del rango de tolerancia especificado.
4. Establecer una trayectoria de vuelo predefinida con puntos de referencia.
5. Observar el vuelo del dron y registrar cualquier desviación de la trayectoria dentro del rango de tolerancia especificado.
6. Realizar maniobras de cabeceo, alabeo y guiñada y observar las desviaciones de vuelo registradas.



7. Verificar que las desviaciones se mantengan dentro del rango de tolerancia establecido.

**Resultados esperados:**

El dron mantiene una altitud dentro del rango de tolerancia de  $\pm 10$  cm. El dron mantiene un patrón de vuelo estable con desviaciones de cabeceo, alabeo y guiñada dentro del rango de tolerancia de  $\pm 5$  grados. El dron sigue la trayectoria predefinida con desviaciones mínimas dentro del rango de tolerancia de  $\pm 20$  cm.

**Resultados obtenidos:**

El dron ha sido capaz de mantener una altitud, una trayectoria y un patrón de vuelo estable de acuerdo con las tolerancias permitidas y establecidas con anterioridad.

### 4.2.2. Prueba 2

**Procedimiento de la prueba:**

1. Cargar el dron con el peso máximo especificado, incluyendo los componentes montados.
2. Verificar que el dron esté en condiciones de vuelo y configurado correctamente.
3. Hacer despegar el dron y observar si puede mantenerse en vuelo de manera estable.
4. Registrar cualquier dificultad o inestabilidad en el vuelo.
5. Aterrizar el dron de manera segura y controlada.

**Resultados esperados:**

El dron puede levantar su propio peso más los componentes montados sin dificultad y mantener un vuelo estable.

**Resultados obtenidos:**

El dron ha sido capaz de levantar su propio peso más el de los componentes montados y la batería sin que ello suponga una dificultad añadida a la hora de realizar el vuelo en buenas condiciones.

### 4.2.3. Prueba 3

**Procedimiento de la prueba:**

1. Configurar el GCS correctamente y establecer la conexión con el dron.
2. Utilizar el GCS para enviar comandos de control manual al dron.
3. Probar el control del movimiento del dron en diferentes direcciones: adelante, atrás, izquierda, derecha, arriba y abajo.
4. Ajustar la velocidad y altitud del dron utilizando el GCS en respuesta a las señales del control remoto.

**Resultados esperados:**

El GCS puede controlar el movimiento del dron en diferentes direcciones y ajustar la velocidad y altitud del dron según sea necesario.

**Resultados obtenidos:**

El dron ha respondido de manera adecuada a las instrucciones de control enviadas por el GCS. Estas instrucciones estaban relacionadas con el despegue y aterrizaje, movimiento simples y trayectorias más complejas.

#### 4.2.4. Prueba 4

**Procedimiento de la prueba:**

1. Verificar que el dron esté equipado con un sistema de control automático.
2. Configurar el GCS para controlar el dron de forma automática.
3. Establecer una trayectoria de vuelo predefinida para el dron.
4. Iniciar el control automático del dron a través del GCS y observar su comportamiento durante el vuelo.
5. Registrar cualquier desviación de la trayectoria predefinida.

**Resultados esperados:**

El dron puede ser controlado de forma automática a través del GCS y sigue la trayectoria predefinida con desviaciones mínimas.

**Resultados obtenidos:**

El dron puede ser controlado de manera automática y sigue la trayectoria predefinida con desviaciones mínimas.

#### 4.2.5. Prueba 5

**Procedimiento de la prueba:**

1. Verificar que el GCS esté correctamente conectado al dron y recibiendo datos en tiempo real.
2. Observar la interfaz del GCS y verificar la disponibilidad de información sobre la ubicación, altitud, vida útil de la batería y posibles errores o fallos del dron.
3. Realizar acciones específicas con el dron y observar cómo se refleja la información en el GCS.
4. Verificar que la retroalimentación del estado del dron sea claramente visible y fácilmente accesible en el dispositivo de control remoto.

**Resultados esperados:**

El GCS muestra información en tiempo real sobre la ubicación, altitud, vida útil de la batería y posibles errores o fallos del dron. La retroalimentación del estado del dron es claramente visible y fácilmente accesible en el dispositivo de control remoto.

#### **Resultados obtenidos:**

El dron ha pasado con éxito la prueba de retroalimentación del estado del dron. A través del software cfclient se puede ver la información enviada por el dron relativa a las coordenadas, porcentaje de batería, nivel de potencia y potencia individual de cada motor.

#### **4.2.6. Prueba 6**

##### **Procedimiento de la prueba:**

1. Configurar el dron en una posición estacionaria para el despegue.
2. Iniciar el procedimiento de despegue automático desde el GCS y observar el despegue del dron.
3. Verificar que el dron despegue de manera segura y sin requerir intervención manual del operador.
4. Configurar el dron en una posición de vuelo estable para el aterrizaje.
5. Iniciar el procedimiento de aterrizaje automático desde el GCS y observar el aterrizaje del dron.
6. Verificar que el dron aterrice de manera segura y controlada sin requerir intervención manual del operador.

##### **Resultados esperados:**

El dron realiza el despegue automático de manera segura y sin intervención manual. También aterriza de manera segura y controlada sin requerir intervención manual.

##### **Resultados obtenidos:**

El dron es capaz de despegar y aterrizar sin intervención manual y de manera segura para la gente de alrededor y sin peligrar la integridad del propio dron por posibles impactos contra el suelo.

#### **4.2.7. Prueba 7**

##### **Procedimiento de la prueba:**

1. Configurar el dron en una posición estacionaria para el despegue.
2. Iniciar el procedimiento de despegue automático desde el GCS y observar el despegue del dron.
3. Verificar que el dron despegue de manera segura y sin requerir intervención manual del operador.
4. Mantener el dron en vuelo estacionario.
5. Cronometrar el tiempo en que el dron está volando.
6. Verificar que el dron aterrice de manera segura y controlada una vez se ha cumplido el tiempo necesario para la competición.

#### **Resultados esperados:**

El dron aguanta en vuelo estacionario 4 min, tiempo máximo que, según la normativa de la competición, debe durar las pruebas.

#### **Resultados obtenidos:**

El dron ha sido capaz de aguantar en condiciones de vuelo estacionario durante 4 minutos, que es el máximo permitido por las normas de la competición. Además, ha sobrado batería, lo cual permite aumentar la potencia y, por tanto, disminuir el tiempo de completado de las pruebas de la competición y así tener más posibilidades de ganar.

### **4.2.8. Prueba 8**

#### **Procedimiento de la prueba:**

1. Crear una trayectoria con curvas de suficiente complejidad.
2. Configurar el dron en una posición estacionaria para el despegue, en la que tenga espacio suficiente para realizar una trayectoria compleja.
3. Iniciar el procedimiento de despegue automático desde el GCS y observar el despegue del dron.
4. Verificar que el dron despegue de manera segura y sin requerir intervención manual del operador.
5. Observar y grabar el vuelo del dron para analizar si cumple con los requisitos.
6. Verificar que el dron aterrice de manera segura y controlada una vez se ha completado la trayectoria.

#### **Resultados esperados:**

Las curvas creadas deben ser enviadas por GCS y recibidas por el dron. El dron debe ser capaz de seguir una trayectoria avanzada enviada a través de GCS.

#### **Resultados obtenidos:**

El dron ha realizado varias trayectorias avanzadas, una con forma de "8", otra con forma de estrella y otra con forma de espiral, y volver a la posición fijada por el GCS. Las trayectorias fueron generadas y enviadas correctamente.

### **4.2.9. Prueba 9**

#### **Procedimiento de la prueba:**

1. Grabar un vídeo de obstáculos mediante una cámara que no va incorporada en el dron.
2. Verificar que la resolución del vídeo es la adecuada para un buen procesamiento de imagen.
3. Comprobar que el código de análisis de imagen realiza de manera correcta la identificación de obstáculos.

#### **Resultados esperados:**

El código es capaz de identificar los obstáculos mediante un algoritmo de procesamiento de imagen.

#### **Resultados obtenidos:**

El código implementado ha sido capaz de reconocer los obstáculos que observaba a través del vídeo grabado específicamente para esta prueba.

## **5. Limitaciones identificadas y resultados de la verificación y la validación**

A continuación, se resumen las conclusiones extraídas de las diferentes pruebas realizadas. Se exponen las limitaciones detectadas en el sistema y se concluye si el sistema cumple con los requisitos y especificaciones (verificación) y si satisface las necesidades del cliente (validación).

### **5.1. Limitaciones identificadas**

En este apartado se describen las limitaciones principales de nuestro sistema que se identificaron durante la fase de pruebas del dron.

#### **5.1.1. Incompatibilidad del AI Deck**

La limitación más importante identificada durante las pruebas es la imposibilidad de conectar el dron con el módulo AI Deck 1.1. Al seguir las instrucciones disponibles en la página oficial, nos encontramos con un problema irremediable en el cual el firmware del módulo no era inmediatamente compatible con el dron.

Se intentó resolver este problema mediante una búsqueda y prueba exhaustiva de los manuales y foros disponibles en Internet. Además, pedimos ayuda al Grupo de Investigación en Visión por Computador y Robótica Aérea de la UPM (CVAR-UPM), que después de muchas pruebas tampoco pudieron solucionar el problema.

El fallo principal identificado fue la necesidad de un dispositivo específico llamado "JTAG programmer/debugger", necesario para poder reprogramar el firmware del módulo con una versión no defectuosa. Este dispositivo no estaba en nuestra posesión ni tampoco lo tenía el CVAR-UPM. El precio de este dispositivo en Internet rondaba los 50€ y, tomando en cuenta la experiencia de compra del propio dron, se decidió que no sería posible su adquisición y uso a tiempo.

#### **5.1.2. Incompatibilidad del LED-ring deck con el Flow deck v2**

La segunda limitación fue la imposibilidad de utilizar los módulos LED-ring deck y Flow deck v2 simultáneamente con el dron. Su uso es incompatible por el propio diseño de los módulos, que deben montarse en el mismo hueco sin posibilidad de ampliar los pines de conexión.

Se buscó una solución a este problema en Internet. La única solución propuesta implicaba desoldar una resistencia del LED-ring deck y varias maniobras más de diversa complejidad, pero los resultados podrían afectar negativamente al Flow deck v2. Se decidió no intentarlo por riesgo de dañar el módulo Flow deck v2 y por la poca relevancia del LED-ring deck.

## 5.2. Conclusiones de verificación

En la siguiente tabla se pueden ver resumidos los requisitos aplicables al dron y al sistema de control del dron. La tabla se estructura en una primera columna con el código de requisito, cuya explicación detallada puede consultarse en la última versión disponible del documento “System Requirements Specifications (Final)”, una columna central que indica si se ha cumplido el requisito o no y una última columna con comentarios útiles.

Código de requisito	Resultado	Comentarios
Drone/Flight/SyRS/SY-001	Pasa	Prueba 1
Drone/Motors/SyRS/SY-001.1	Pasa	Prueba 2
Drone/Networking/SyRS/SY-001.2	No pasa	Código no implementado
Drone/Networking/SyRS/SY-001.3	Pasa	Prueba 4
Drone/Onboard Computer/SyRS/SY-001.4	Pasa	Prueba 5
Drone/Onboard Computer/SyRS/SY-001.5	Pasa	Prueba 6
Drone/Onboard Computer/SyRS/SY-001.6	Pasa	Prueba 1
Drone/Flight/SyRS/SY-002	Pasa	Incluido en cfclient
Drone/Battery/SyRS/SY-003	Pasa	Prueba 7
Drone/Battery/SyRS/SY-003.1	Pasa	3 baterías disponibles
Drone/Onboard Computer/SyRS/SY-004	No pasa	AI deck no funcional
Drone/Vision System/SyRS/SY-004.1	No pasa	AI deck no funcional
Drone/Vision System/SyRS/SY-004.1.1	No pasa	AI deck no funcional
Drone/Flight/SyRS/SY-004.1.2	No pasa	AI deck no funcional
Drone/Flight/SyRS/SY-004.2	Pasa	Prueba 8
Drone/Flight/SyRS/SY-004.3	Pasa	Prueba 8
Software/Source Code/SRS/SO-001	Pasa	A través de GitHub
Networking/Communications/SRS/SO-002	Pasa	Prueba 4
Networking/Communications/SRS/SO-002.1	No pasa	Código no implementado
Networking/Communications/SRS/SO-002.2	No pasa	Código no implementado
Networking/Communications/SRS/SO-002.3	Pasa	Prueba 8
Networking/Communications/SRS/SO-002.4	No pasa	AI deck no funcional
Central Computer/Vision/SRS/SO-003	Pasa	Prueba 9
Central Computer/Vision/SRS/SO-004	Pasa	Incluido en cfclient
Central Computer/Vision/SRS/SO-005	Pasa	Prueba 8
Software/Programming Language/SRS/SO-005.1	Pasa	Lenguaje Phyton 3.x

Interpretando la tabla, podemos destacar que la mayoría de los requisitos que el dron y el sistema de control no pasan se debe al fallo de conexión del AI deck 1.1. Además, no se ha implementado el código necesario para detectar un fallo de comunicación con el sistema de control y aterrizar de manera segura en este caso.

En la siguiente tabla se pueden ver también los requisitos aplicables a la propia competición. En este caso, el procedimiento de verificación consiste en consultar la normativa de la competición redactada en el documento “OpsCon (Final)”.

Código de requisito	Resultado
Competition/Deadline/URS/ST-001	No pasa
Competition/Environment/URS/ST-002	Pasa
Competition/Environment/URS/ST-002.1	Pasa
Competition/Environment/URS/ST-003	Pasa
Competition/Environment/URS/ST-003.1	Pasa
Competition/Environment/URS/ST-003.2	No pasa
Competition/Environment/URS/ST-003.3	Pasa
Competition/Environment/URS/ST-003.4	Pasa
Competition/Environment/URS/ST-003.5	Pasa

Se puede observar que la prueba de la competición relacionada con el requisito .../ST-003.2 no fue implementada en la normativa de la competición. Además el dron no es capaz de competir en la fecha indicada en el requisito .../ST-001.

### 5.3. Conclusiones de validación

En resumen, se han identificado limitaciones significativas durante las pruebas del sistema. La primera limitación se refiere a la incompatibilidad del AI Deck 1.1, debido a problemas de firmware y la falta de un dispositivo específico para reprogramarlo. A pesar de los esfuerzos realizados para resolver este problema, no se pudo encontrar una solución viable. La segunda limitación es irrelevante de cara a la validación.

En cuanto a la verificación del sistema, se han realizado pruebas en base a los requisitos establecidos. A través de la tabla que resume los resultados, se observa que hay requisitos que no se cumplen debido a la limitación del AI Deck 1.1. Además, no se ha implementado el código necesario para detectar fallos de comunicación y aterrizar de manera segura. En cuanto a los requisitos de la competición, también se identifica que el dron no está preparado para las fechas establecidas y una prueba no implementada.

En conclusión, debido a las limitaciones y los requisitos no cumplidos, se determina que el sistema no puede considerarse validado en su estado actual, dado que no cumple con la necesidad del cliente de la organización de una competición de drones autónomos.