



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Escuela Técnica Superior de Ingeniería del Diseño

Integración de un autopiloto con sistema de control y
navegación de largo alcance en un UAS de tipo VTOL.

Trabajo Fin de Máster

Máster Universitario en Ingeniería Aeronáutica

AUTOR/A: Claver Marí, Lucas

Tutor/a: Quintanilla García, Israel

CURSO ACADÉMICO: 2021/2022

Resumen

El objetivo de este TFM es implementar tanto autopiloto como sistema de control y navegación de largo alcance en un UAS ("Unmanned Aircraft System") de tipo VTOL ("Vertical Take Off and Landing"). Para la puesta a punto de la aeronave se han empleado simulaciones del tipo "Software in the loop" y "Hardware in the loop", que sirven como pruebas de apoyo para la verificación del buen funcionamiento del autopiloto y sistemas embarcados. A partir de estas se podrán llevar a cabo las pruebas en vuelo.

Resum

L'objectiu d'aquest Treball de Fi de Màster es implementar tant auto pilot com sistema de control de navegació de llarg abast en un UAS ("Unmanned Aircraft System") de tipus VTOL ("Vertical Take Off and Landing"). Per a la posada a punt de l'aeronau s'han emprat simulacions del tipus " Software in the loop " i " Hardware in the loop ", que serveixen com a proves de suport per a la verificació del bon funcionament de l'autopilot i sistemes embarcats. A partir d'aquestes es podran portar a terme les proves en vol.

Abstract

The main aim of this TFM is to implement both an autopilot and a long-range navigation and control system in a VTOL ("Vertical Take Off and Landing") UAS ("Unmanned Aircraft System"). For good performance "Software in the loop" and "Hardware in the loop" simulations have been used. They serve as a support for verifying the proper functioning of the autopilot and embedded systems. From these, the flight tests can be carried out.

Índice general

Resumen	3
Índice general	5
Índice de figuras	7
Índice de tablas.....	9
Lista de símbolos y abreviaturas	11
Memoria	13
1. Objetivos e introducción.....	15
1.1. Objetivos	15
1.2. Introducción	15
2. UAS. Definición y normativa	17
2.1. Definición y clasificación	17
2.2 Normativa actual.....	22
2.2.1. Clases AESA.....	23
2.2.2. Categorías	24
2.3. Integración de UAS con tráfico tripulado.....	25
3. Aeronave y autopiloto	27
3.1. Componentes	28
3.2 Autopiloto.....	31
3.3. PX4.....	32
3.3.1. Controladores MC.....	33
3.3.2. Controladores FW.....	36
3.3.3. Modos de vuelo	38
4. Sistema de control y navegación	40
4.1 Componentes embarcados	40
4.2. Componentes en tierra	41
4.3 Diagrama de conexiones	44
5. Simulaciones y preparación para el vuelo	46
5.1. Software in the loop.....	46
5.1.1. Entorno de desarrollo y simulación.....	47
5.1.2. Configuración de la aeronave en PX4.....	49
5.1.3. Ejecución de la simulación y resultados	56
5.2. Hardware in the loop	61

5.3. Preparación para el vuelo	64
5.3.1. Puesta a punto de todos los componentes	64
6. Conclusiones y posibles futuros trabajos	67
Bibliografía	68
Presupuesto	71
7. Presupuesto del proyecto	73
7.1 Recursos	73
7.2 Precios unitarios	74
7.3 Presupuesto total	76

Índice de figuras

Figura 2.1. Componentes de un UAS.....	18
Figura 2.2. Comparativa de prestaciones entre ala fija, multicoptero y helicóptero	20
Figura 2.3. Quad plane, tail sitter, tiltrotor y tilt wing, respectivamente	21
Figura 2.4. Lillium Jet, proyecto de aerotaxi	22
Figura 2.5. Fases de implementaciones del U-Space.	25
Figura 3.1. Babyshark VTOL 260 de FoxTech.....	27
Figura 3.2. Controladora Cube Orange de Cube Pilot.	29
Figura 3.3. Diagrama de puertos de Cube Orange.	30
Figura 3.4. Dispositivo Here3 montado sobre la parte superior de la aeronave.	30
Figura 3.5. Controladores del modo Multicoptero	34
Figura 3.6. Diagrama de bloques del controlador de posición en MC.	34
Figura 3.7. Diagrama de bloques del controlador de velocidad en MC.....	35
Figura 3.8. Diagrama de bloques del controlador de actitud en MC	35
Figura 3.9. Diagrama de bloques del controlador de velocidad angular en MC.....	36
Figura 3.10. Diagrama de bloques del controlador de actitud en FW	38
Figura 3.11. Diagrama de bloques del controlador de actuaciones VTOL	38
Figura 4.1. Diagrama de conexiones del receptor RXLRS.....	40
Figura 4.2. Transmisores de video embarcado XOSD3B	41
Figura 4.3. Estación de control en tierra GCSD4 V2	42
Figura 4.4. Receptor redundante y antena bidireccional.....	42
Figura 4.5. Antena secundaria 433 MHz.	43
Figura 4.6. Receptor de video RXVID3 de 2.4 GHz y 5.8 GHz.	43
Figura 4.7. Diagrama de conexiones del lado aire del sistema de telemetría redundante XLRS	44
Figura 4.8. Diagrama de conexiones del lado tierra del sistema de telemetría redundante XLRS	45
Figura 5.1. Comunicaciones SITL entre simulador estación de control y PX4.....	47
Figura 5.2. Modelo del vehículo original de gazebo como standard VTOL.....	49
Figura 5.3. Modelo 3D de la aeronave BabyShark en Gazebo.	50
Figura 5.4. Archivo mezclador principal empleado para SITL.	53
Figura 5.5. Archivo CMakeList de la carpeta de mezcladores.....	54

Figura 5.6. Archivo mezclador auxiliar empleado para SITL.....	54
Figura 5.7. Grupo de control de mezcladores para VTOL	55
Figura 5.8. Ventana de configuración de QGroundControl.....	56
Figura 5.9. QGroundControl durante la simulación de un vuelo.	57
Figura 5.10. Vuelo del BabyShark en Gazebo durante una simulación.....	57
Figura 5.11. Ventana de configuración de QGC para activar el joystick virtual.	59
Figura 5.12. Herramienta de QGC para gestión y descarga de logs de vuelo.	59
Figura 5.13. Herramienta de configuración de los controladores PID.	60
Figura 5.14. Angulo de cabeceo durante vuelo en ala fija previo al autotune.	60
Figura 5.15. Angulo de cabeceo durante vuelo en ala fija posterior al autotune.....	61
Figura 5.16. Comunicaciones HITL entre simulador, estación de control y PX4	62
Figura 5.17. Ventana de configuración en QGC para activar la simulación HITL.	62
Figura 5.18. Archivo SDF para simulaciones HITL.....	63
Figura 5.19. Configuración de radio principal en DMD Studio.....	64
Figura 5.20. Objetos e instrumental de vuelo superpuestos al video de la cámara FPV.	65
Figura 5.21. Configuración del enlace entre la estación de control y el sistema de comunicaciones.....	66

Índice de tablas

Tabla 3.1. Dimensiones de la aeronave.....	28
Tabla 5.1. Tabla de comandos SITL según el vehículo a simular	48

Lista de símbolos y abreviaturas

ADS-B	Automatic Dependent Surveillance - Broadcast
ATC	Air Traffic Control
BVLOS	Beyond Visual Line of Sight
EASA	European Aviation Safety Agency
ESC	Electric Speed Controller
FPV	First Person View
FW	Fixed Wing
HITL	Hardware In the Loop
MC	Multicóptero
MTOM	Maximum Take Off Mass
OACI	Organización de Aviación Civil Internacional
PID	Proporcional Integral y Derivativo
QGC	QGroundControl
RPAS	Remotely Piloted Aerial System
SDF	Simulation Description File
SITL	Software In the Loop
TECS	Total Energy Control System
UAM	Urban Air Mobility
UAS	Unmanned Aircraft System
UTM	UAS Traffic Management
VLOS	Visual Line of Sight
VTOL	Vertical Take Off and Landing

Parte I

Memoria

I. Memoria

1. Objetivos e introducción

1.1. Objetivos

El principal objetivo de este trabajo es poner en funcionamiento la aeronave tipo VTOL con un autopiloto y un sistema de control y navegación de largo alcance. Concretamente se trata de un BabyShark 260 y el sistema de control de largo alcance XLRs (Extra Long Range System) de los cuales se hablará en profundidad en los siguientes apartados.

Para ello se va a ir llevando a cabo objetivos secundarios que nos permitirán conseguir nuestro objetivo principal:

- Implementación del autopiloto de software libre en la controladora de vuelo.
- A partir del modelo 3D de la aeronave y de simulaciones de la aerodinámica crear el entorno de simulación de vuelo con autopiloto.
- Realizar simulaciones en dicho entorno del tipo Software in the loop y Hardware in the loop.
- Implementar el sistema de control en la aeronave.
- Llevar a cabo pruebas en vuelo con autopiloto.

Para ello se ha tenido que aprender previamente sobre el funcionamiento del autopiloto, el entorno de simulación, ambos tipos de simulaciones, comprender el funcionamiento de las comunicaciones entre aeronave y estación de control en tierra, así como los protocolos de comunicación entre estos. Todos estos temas se abordarán a lo largo del trabajo.

1.2. Introducción

Este trabajo se ha desarrollado en la UPV concretamente dentro del grupo de investigación WUAS UPV (Working Unmanned Aircraft System UPV). Se ha partido de proyectos que se habían llevado a cabo meses anteriores.

Dado que los equipos y aeronave son costosos se optó en un primer momento por adaptar un UAS de tipo avión convencional en un VTOL añadiendo cuatro rotores con sus respectivas modificaciones estructurales y de cableado con el fin de asemejarse al VTOL que se quiere configurar. Dichas modificaciones se implementaron en un RaptorV2 del fabricante Voltantex RC.

Este proyecto se realizó en [1], donde debido a los cambios introducidos a la aeronave se comprometió ligeramente la capacidad de transicionar establemente entre multirotor y ala fija debido a la introducción de peso y falta de rigidez estructural, dado que la aeronave que se empleó era muy ligera. A pesar de eso, ese proyecto brindó una mejor visión acerca del software del autopiloto y del funcionamiento de ciertos equipos embarcados que también se usarán en este proyecto.

El procedimiento de validación de autopiloto y sistema de control es similar al empleado en la aeronave de prueba inicial, con una primera aproximación a sus características

I. Memoria

aerodinámicas mediante la creación de su modelo 3D y la evaluación de su desempeño aerodinámico realizando simples simulaciones que nos proporcionen información acerca de su comportamiento dinámico. Posteriormente con simulaciones de vuelo que nos permitan configurar el autopiloto y sus controladores para posteriormente probar el hardware del autopiloto y después de las verificaciones oportunas del sistema de comunicaciones, realizar las pruebas de vuelo necesarias.

2. UAS. Definición y normativa

En este apartado se van a introducir el concepto de UAS y todo lo que este engloba, así como los diferentes tipos y clasificaciones según tipología de vuelo, características de las operaciones que lleven a cabo y su riesgo operacional.

2.1. Definición y clasificación

Cuando nos referimos a un UAS (Unmanned Aircraft System), nos referimos a un Sistema de Aeronave No tripulada, de forma que se reciben las ordenes de control de vuelo de manera remota o mediante un piloto automático preconfigurado para realizar la misión que se requiera.

El término UAS, es usado de forma global para referirse a todos los elementos y sistemas, además del propio vehículo, que forman parte del vuelo de la aeronave. También se conocen como UAV (Unmanned Aerial vehicles), referido únicamente al vehículo, UAS (Unmanned Aerial System), que hace referencia a los sistemas embarcados, RPAS (Remotely Piloted Aircraft System), que no incluyen vehículos con autonomía completa y por último se les llama comúnmente drones. Un UAS está constituido por diferentes elementos [3]:

- Aeronave con sus sistemas propulsivos y de alimentación.
- Estación de control remoto en tierra. Permite recibir los datos de la aeronave, tanto imágenes en primera persona de la cámara abordo como del estado de sus componentes y sensores. También permite la planificación de misiones, el control manual del vuelo de la aeronave y su carga de pago.
- Operador humano. Aunque su presencia podría ser prescindible en UAS autónomos, que haya alguien supervisando el funcionamiento del vuelo añade seguridad a éste y permite evitar accidentes en caso que algo falle.
- Sistema de navegación. Se incluye en este tanto sensores encargados de determinar posición, orientación, altitud y velocidad, como receptores GPS.
- Enlace de datos. Sistema encargado de transmitir y recibir los datos entre aeronave y estación de control en tierra. Se transmiten los comandos de vuelo que realiza el operador y se recibe la telemetría de los componentes a bordo de la aeronave.
- Carga de pago. Dependiendo de la misión puede que sean radares, cámaras específicas con visión infrarroja o cámaras térmicas. Habitualmente este tipo de cámaras incorporan dos o tres ejes de rotación con pequeñas almohadillas que absorben las vibraciones con el fin de mejorar la calidad de los datos que se recogen.
- Equipos adicionales. Incluyen cualquier equipo necesario tanto para el despegue como para el aterrizaje, ya sean catapultas, cohetes, cuerdas o paracaídas, ya sea por necesidad o por seguridad, en caso de fallo catastrófico o imposibilidad de aterrizaje seguro.

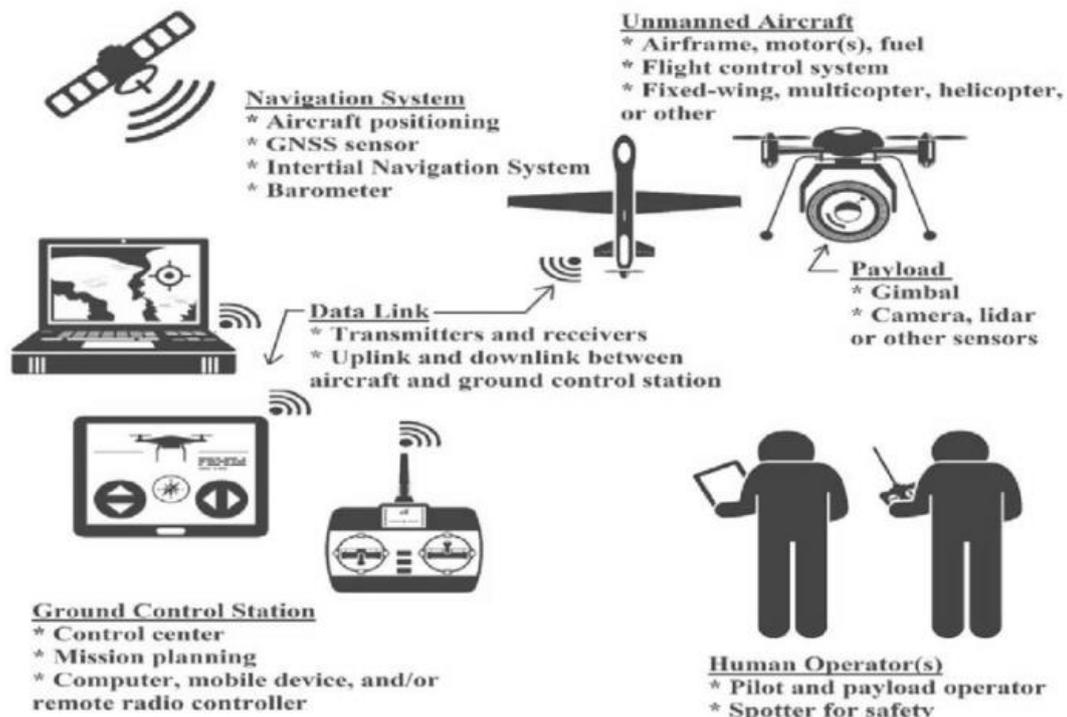


Figura 2.1. Componentes de un UAS. [12]

El sector de los UAS ha crecido exponencialmente en los últimos años a raíz de la gran variedad de aplicaciones que se han explorado. La razón principal es la gran versatilidad que esta tecnología nos ofrece, desde aplicaciones donde se requieren aeronaves pequeñas con gran maniobrabilidad, hasta aeronaves de gran envergadura, todas ellas permitiendo que el piloto no esté a bordo.

Este sector se divide principalmente en dos ámbitos de aplicación, el militar y el civil/comercial. Como sabemos, los desarrollos militares son una rama de la tecnología muy importante en cuanto a aportaciones al ámbito civil, mucha de la tecnología que existe a día de hoy se desarrolló al inicio con fines militares y luego se aprovechó para fines civiles o comerciales.

En el ámbito militar, los UAS realizan tareas de vigilancia, de reconocimiento del terreno, y también se usan como armas, cosa que hemos visto en la historia reciente durante guerras. El uso de UAS es una gran ventaja ya que provee de seguridad a los pilotos al no correr el riesgo de ser interceptado, permite que las aeronaves sean más pequeñas y que vuelen más bajo, con menos ruido y con menor probabilidad de ser detectadas por otros radares. Además en caso que el UAS sea interceptado, es mucho más económico que perder un avión de combate como un Eurofighter, por ejemplo.

En cuanto al ámbito civil y comercial, los UAS se están utilizando en construcción, agricultura, mantenimiento e incluso como herramienta de vigilancia para los cuerpos de seguridad del estado. Las inversiones que empresas realizan en esta tecnología son cada vez mayores y las tareas que desempeñan los UAS son cada vez más numerosas.

Dependiendo del tipo de despegue y de la tipología del ala mediante la cual se obtiene la sustentación, los UAS se pueden dividir de manera muy general según sean de ala fija o rotatoria.

Ala fija

Realiza el despegue y el aterrizaje en horizontal, generando la fuerza de sustentación con el ala. Los motores proporcionan empuje en la dirección de avance y el control de actitud, guiñada cabeceo y alabeo, se realiza mediante las superficies de control.

Las ventajas principales de este tipo de aeronaves son las altas velocidades de vuelo y el gran alcance. En cuanto a transporte de carga, también permiten gran capacidad de carga de pago, aunque se ve reducida debido a que gran parte del peso corresponde al almacenamiento de la energía necesaria para los motores, ya sean baterías o combustible.

Por el contrario, las desventajas principales son la necesidad de una pista de aterrizaje y las limitaciones inherentes a la imposibilidad de realizar vuelo a punto fijo.

Ala rotatoria

Este tipo de aeronaves no necesitan pista de aterrizaje para realizar las maniobras de despegue y aterrizaje, ya que la potencia de los motores es la que genera tanto el empuje como la sustentación mediante la rotación de las palas. El control de actitud se lleva a cabo mediante modificaciones de velocidad en el rotor de las palas, o movimientos relativos de estas, con batimiento, paso y arrastre, según las características de la unión entre palas y rotor.

Las ventajas son que puede volar a punto fijo y la infraestructura reducida para las operaciones de despegue y aterrizaje.

Por el contrario, el alcance de este tipo de aeronaves se ve reducido y la velocidad de vuelo es relativamente baja frente a las aeronaves de ala fija.

Cuando hablamos de UAS de ala rotativa diferenciamos entre del tipo helicóptero y multicóptero. Los multicópteros se diferencian por tener tres o más rotores situados en el mismo plano, a los cuales modifican la potencia para realizar el control de actitud, mientras que los helicópteros utilizan el paso, batimiento y arrastre para dicha función. Los multicolores permiten una gran maniobrabilidad en todas direcciones, aunque comúnmente la carga de pago que pueden llevar es mucho menor.

A continuación, se muestra un gráfico comparativo entre carga, tiempo de vuelo y alcance de estas tres configuraciones, que son las más comunes.

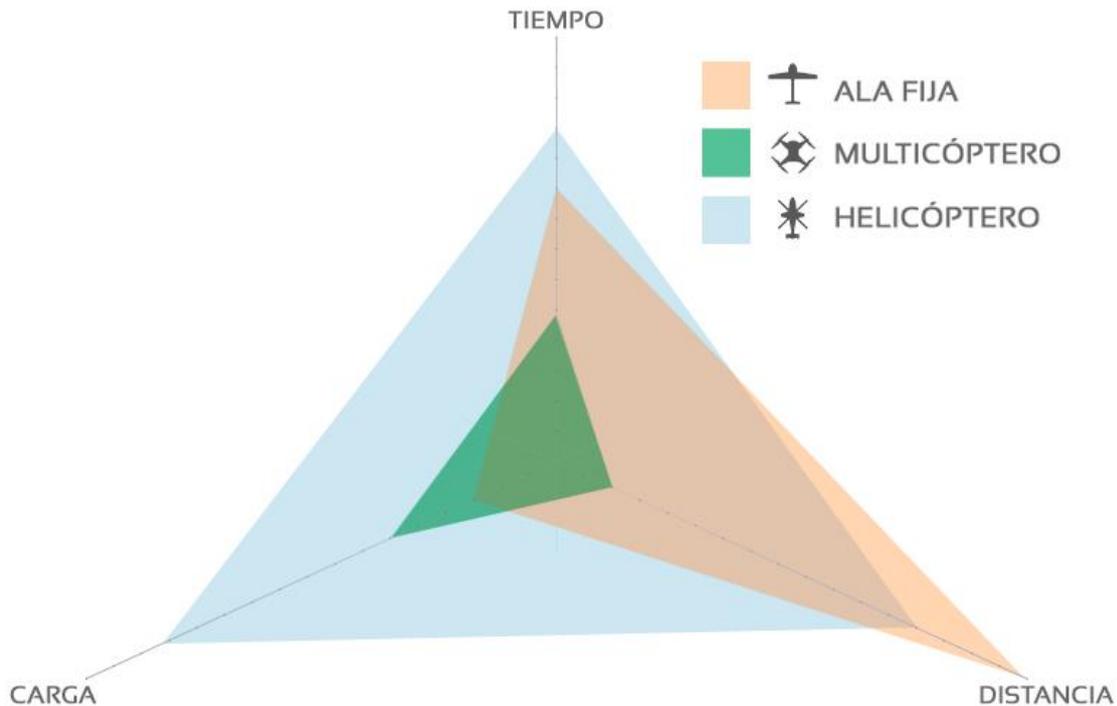


Figura 2.2. Comparativa de prestaciones entre ala fija, multicóptero y helicóptero. [4]

Dadas las prestaciones que los diferentes tipos de aeronaves nos ofrecen, vemos que el ala fija es la mejor opción para desplazamientos más largos y de más rápidos. Sin embargo, dado que su principal problema es el despegue, aterrizaje y el vuelo a punto fijo, surge la posibilidad de fabricar aeronaves capaces de ofrecer las prestaciones del ala fija y las ventajas del ala rotativa, de ahí nace el concepto de VTOL (Vertical Take Off and Landing). Este tipo de aeronaves nos permiten despegar en vertical, transicionar a vuelo con ala fija durante la parte de crucero de la misión, y volver a volar usando los rotores que proporcionan empuje en dirección vertical, tanto para realizar el aterrizaje o si se necesita volar a punto fijo por requerimientos de la misión.

Al realizar este híbrido entre dos tipologías de aeronave, no solo conseguimos todo lo bueno de ambas, sino que también se tienen desventajas, como una menor autonomía, menor capacidad de carga y menores eficiencias aerodinámicas ya que ambos modos de vuelo interfieren entre ellos, teniendo que cargar, por ejemplo, con los rotores de vuelo a punto fijo cuando volamos con ala fija y viceversa. A raíz de estas desventajas, se han realizado multitud de configuraciones con tal de minimizarlas. Las configuraciones más destacadas son las siguientes.

- Quadplane. Se trata de una configuración de ala fija convencional que puede o no incluir estabilizador horizontal, con la diferencia que se añaden en las alas dos sujeciones para los cuatro rotores que actuarán en vuelo a punto fijo, además de los rotores para vuelo horizontal con ala fija.
- Tailsitter. Presenta una configuración habitual de ala delta, con la capacidad de aterrizar con el morro de la aeronave hacia arriba. Las transiciones entre vuelo vertical y horizontal son complejas, debido al vuelo vertical, en el que el aire incide lateralmente sobre las alas y dificulta el control de posición.

- Tiltrotor/wing. En este caso tenemos una configuración convencional de avión, con la particularidad de que o el ala junto con el motor tiene la capacidad de girar 90° o solo es el rotor el que realiza el giro. También se pueden añadir rotores de cola para mejorar la maniobrabilidad y estabilidad en vuelo a punto fijo. Este tipo de aeronaves permiten un mejor control que los tailsitter, aunque precisan de mayores actuadores, dado que necesitan hacer girar los rotores o ala.



Figura 2.3. Quad plane, tail sitter, tiltrotor y tilt wing, respectivamente. [16,13,14,15]

Actualmente se están llevando a cabo proyectos con este tipo de aeronaves híbridas bajo el proyecto de la EASA, *European Aviation Safety Agency*, llamado UAM, *Urban Air Mobility*. En este proyecto se pretende introducir el transporte de pasajeros y carga en entornos urbanos y desplazamientos cortos, espacio aéreo que requiere una seguridad adicional debido obviamente a las zonas pobladas que se sobrevuelan. Es por eso que desde EASA se está trabajando en crear un marco normativo que regule las operaciones, la navegabilidad y la integración del espacio aéreo, además de fomentar la investigación y el desarrollo tecnológico con el fin de empezar a llevar a cabo este tipo de vuelos durante los próximos años.

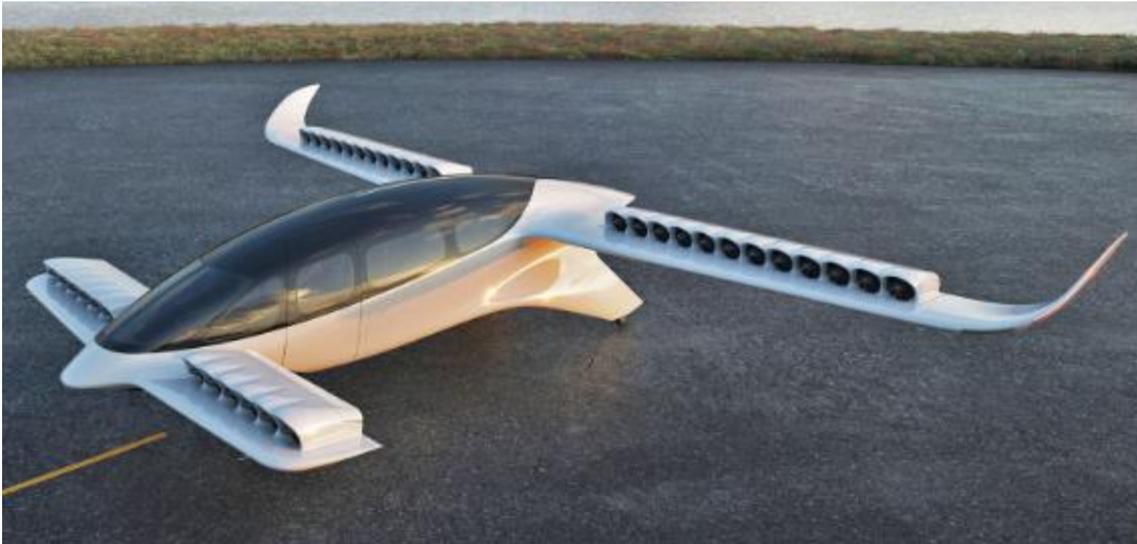


Figura 2.4. Lilium Jet, proyecto de aerotaxi. [17]

Entre los proyectos que se están desarrollando hay varios que cabe destacar el de la empresa alemana Lilium. Se trata de una aeronave VTOL autónoma configurable en 4 o 6 asientos, o para carga de mercancías urgentes, capaz de volar entre 40-200 km de forma totalmente eléctrica. En paralelo otras empresas como Airbus o Vertical Aerospace están desarrollando aeronaves similares, Lilium es solo un ejemplo de los avances que vendrán en esta materia en los próximos años, que necesariamente se acompañarán con novedades normativas y nuevos reglamentos operacionales.

2.2 Normativa actual

Uno de los inconvenientes que el desarrollo se ha ido encontrando a medida que avanzaba era la falta de legislación en ciertos tipos de operaciones que pudieran garantizar una seguridad operacional adecuada, sobre todo en vuelos cerca de núcleos urbanos y de aeródromos. El problema es que la normativa vigente dedicada a vuelos tripulados consta de unas normas y una estructura reglamentaria que se ve muy limitada para incluir las operaciones con UAS debido a su naturaleza evolutiva y complejidad operacional.

Con el fin de garantizar la seguridad, la EASA empezó hace unos años a desarrollar unas reglas comunes para Europa. El enfoque principal es asegurar altos estándares de seguridad, como los logrados en la aviación tripulada. El objetivo es minimizar los riesgos de operaciones y encontrar un equilibrio entre fabricantes y operadores de drones en cuanto a seguridad, privacidad, medio ambiente y contaminación acústica.

Las dos normativas principales en Europa son los Reglamentos EU 2019-947 y EU2019-945. el primero, conocido como reglamento de ejecución describe las categorías de las operaciones y las exigencias para los operadores, mientras que el segundo, conocido como reglamento delegado describe requisitos para las aeronaves.

En cuanto a las normativas estatales, disponemos del Real Decreto 1036/2017 que regula las operaciones, los requisitos y al operador. En el se establecen dos categorías

de operaciones declarativa y autorizada. En la declarativa se llevan a cabo operaciones de riego bajo realizadas fuera de cualquier espacio controlado y siendo el operador el que realice una declaración previa a AESA, mientras que en la autorizada se requiere de un Estudio Aeronáutico de Seguridad (EAS) para poder recibir la autorización. Posteriormente en el Reglamento del Aire, RD 1180/2018, se establecieron requisitos de equipamiento para aeronaves no tripuladas.

2.2.1. Clases AESA

Desde la Agencia Estatal de Seguridad Aérea se distinguen siete clases de UAS según su peso, ya que este condiciona las operaciones relacionadas y por ende la seguridad de las mismas.

C0. Su MTOM es inferior a 250 gramos y su máxima velocidad de 19 m/s. La altura máxima permitida es 120 metros por encima del punto de despegue y tiene que se controlable por el piloto en todo momento.

C1. Su MTOM ha de ser inferior a 900 gramos, limitado a 120m por encima del punto de despegue, ha de ser controlable en todo momento por el piloto. Tiene que llevar luces visibles desde el tierra y aire en operaciones nocturnas e identificación remota.

C2. MTOM inferior a 4 kg incluida carga útil. A los requisitos de las clases C1 y C2 se añaden los siguientes. Enlace de datos protegido contra acceso no autorizado, modo de baja velocidad, 3m/s, seleccionable por el piloto y equipado con función de geoconsciencia.

C3. Mismos requisitos técnicos que los C2 a diferencia de tener un MTOM inferior a 25 kg y una dimensión máxima característica de 3 metros.

C4. Mismos requisitos técnicos que C3 y C2, con MTOM inferior a 25 kg. El piloto debe ser capaz de controlar y maniobrar la aeronave en todo momento de forma segura y no se pueden tener modos de control automático de trayectoria, exceptuando la asistencia a la estabilización del vuelo.

C5. Este tipo de drones son utilizados en categoría específica en escenario estándar 1, llevando a cabo operaciones sobre zonas controladas y en entorno urbano. MTOM inferior a 25kg, no pueden ser aeronaves de ala fija, excepto aeronave cautiva. Equipado con enlace de datos protegido contra acceso no autorizado y con modo de baja velocidad, 5 m/s. Si existe limitación de acceso a determinadas zonas deberá informar al piloto siendo interoperable con el sistema de control de vuelo.

C6. Se utilizan en categoría específica bajo escenario estándar 2, zona terrestre controlada en entorno escasamente poblado. MTOM inferior a 25 kg, velocidad horizontal máxima de 50 m/s. En caso de pérdida del enlace de datos, se dispondrá de método para recuperarlo o para terminar el vuelo de forma segura. Sistema de aviso por baja batería.

2.2.2. Categorías

Abierta

Las operaciones en esta categoría no están sujetas a ninguna autorización operacional ni a ninguna declaración de operaciones a realizar por parte del operador de forma previa al vuelo. Se consideran operaciones de bajo riesgo operacional. La máxima masa de despegue (MTOM, *Maximum Take Off Mass*) es de 25 kg. Además, no puede llevar mercancías peligrosas ni soltarlas durante el vuelo. Se distingue entre tres subcategorías según la zona de vuelo.

- Categoría A1. Se puede sobrevolar personas, aunque se evita hacerlo sobre personas ajenas a la operación y en ningún caso sobre aglomeraciones de gente.
- Categoría A2. Se vuelan cerca de personas, siempre a una distancia de al menos 30 metros respecto de personas no participantes en la operación
- Categoría A3. Se vuela en todo momento lejos de personas no participantes, a una distancia horizontal mínima de 150 metros de zonas residenciales, comerciales, industriales o recreativas.

Específica

Se trata de operaciones que requieran una autorización de la autoridad competente según el artículo 12, una autorización recibida acorde al artículo 16 o bajo circunstancias definidas en el artículo 5(5), una declaración hecha por el operador de drones. Son operaciones de riesgo operacional medio por lo que se debe realizar una evaluación de riesgos y las medidas de evasión de dichos riesgos. Existen dos escenarios predefinidos en lo que solo sería necesaria la declaración por parte del operador.

- **Escenario estándar 1, STS-01.** Son operaciones en zonas terrestres controladas en entornos urbanos dentro del alcance visual del piloto, VLOS (Visual Line Of Sight). Se llevan a cabo con UAS de identificación de clase C5, como se ha comentado en el apartado anterior 2.2.1.
- **Escenario estándar 2, STS-02.** Son operaciones en una zona terrestre controlada en un entorno poco poblado, en este caso más allá del alcance visual del piloto, BVLOS (Beyond Visual Line Of Sight), y con observadores del espacio aéreo. Se llevan a cabo con UAS que dispongan de identificación de clase C6.

Certificada

Operaciones en esta categoría requerirán la certificación del UAS de conformidad con el Reglamento delegado (UE) 2019/945 y la certificación del operador, y en su caso, la del piloto. Se consideran operaciones de alto riesgo operacional ya que se sobrevuelan concentraciones de personas con un UAS con una dimensión característica de 3m o más. Estas operaciones pueden conllevar transporte de personas o mercancías. Para realizar estas operaciones es necesaria la certificación del UAS por AESA y del operado de UAS, que necesitará una licencia por parte del piloto a distancia.

2.3. Integración de UAS con tráfico tripulado

Al igual que en la aviación tripulada existen proveedores de servicios de gestión de tránsito aéreo (ATM, *Air Traffic Management*) y navegación aérea, lo que se pretende desde la OACI (Organización de Aviación Civil Internacional) es crear un sistema similar que permita la integración de ambos tipos de vuelos, tripulados y no tripulados, dando los servicios oportunos a estos últimos. A esto se le llama UTM (UAS Traffic Management) y proveerá a este tipo de aeronaves de una serie de servicios que permitan mantener la seguridad operacional y minimizar la disrupción del sistema de aviación existente.

Este nuevo servicio que se pretende implementar no excluirá las prestaciones que ya se proporcionan ahora con el servicio de ATM, el principal objetivo es la interoperabilidad. El primer paso consta de integrar UTM y ATM proporcionando consciencia situacional a ambos tipos de aeronaves. Los requerimientos técnicos y operacionales permitirán crear un ecosistema donde operadores y proveedores de servicio se encarguen de la coordinación, ejecución y gestión de las operaciones.

A partir de este concepto, la comisión europea propone una serie de servicios UTM para integrar a los UAS en el cielo europeo, este proyecto europeo se denomina U-Space. Se trata de un marco habilitador para poder llevar a cabo cualquier tipo de misión, incluso en espacios aéreos congestionados. El programa de desarrollo e integración de U-Space sigue 4 fases que empezaron en 2019 [5]:

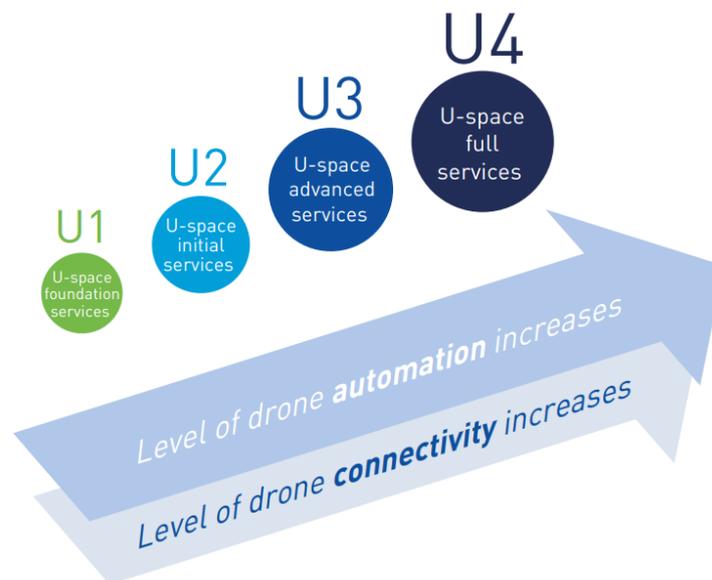


Figura 2.5. Fases de implementaciones del U-Space. [5]

- Fase 1. 2019. Servicios fundamentales del U-Space, a destacar tanto el registro como la identificación electrónica como el sistema de geofencing.
- Fase 2. 2022. Servicios iniciales para gestión de vuelos. Se crea un sistema de planificación de vuelos, aprobación de permisos, seguimiento e interacción convencional con ATC (*Air Traffic Control*).

I. Memoria

- Fase 3. 2027. Servicios avanzados dando apoyo a operaciones más complejas, tanto en entornos urbanos como en zonas densas de tráfico. Se incluyen automatismos para resolución de conflictos entre aeronaves además de funcionalidades de detección y evasión de obstáculos.
- Fase 4. 2030. Servicio completo. Altos niveles en procesos de automatización y comunicación donde se integran completamente aeronaves tripuladas y UAS.

3. Aeronave y autopiloto

La aeronave empleada para este proyecto es un BabyShark VTOL 260 de FoxTech, de fabricación china. Consta de una configuración de ala fija convencional, con la estructura de los cuatro motores para el VTOL integradas en las alas y que se extiende hasta los estabilizadores traseros. A partir de la normativa establecida en el apartado anterior, la aeronave es de clase C3, ya que supera los 4 kg de MTOM de la clase C2, pero no supera los 3 metros de longitud característica máxima que se impone en la C3. En cuanto a la categoría, se englobaría dentro de abierta A3, siempre que se cumplan los requisitos de vuelo del entorno establecidos en ella.



Figura 3.1. BabyShark VTOL 260 de FoxTech. [18]

Esta aeronave consta de ciertas particularidades, como son el motor de empuje vertical situado en la parte trasera del fuselaje, que es más corto de lo habitual y no integra los estabilizadores. Estos se unen a la parte posterior de la estructura de los motores VTOL formando 45 °, lo que se conoce como forma de A. De este modo se integran estabilizadores horizontal y vertical en dos superficies solamente. Cabe destacar la importancia de la sección horizontal que une ambas superficies inclinadas, que provee a la aeronave de sustentación extra, ayudando a compensar los momentos aerodinámicos generados en el ala principal.

En cuanto al ala principal, solamente cuenta con alerones en la parte final para realizar los giros, ya que disponer de flaps, slats o frenos aerodinámicos sería innecesario debido a la naturaleza de despegue y aterrizaje de esta aeronave. Dispone de una cámara gimbal de dos ejes que se añade en la parte inferior delantera que permite recopilar imágenes, independientemente de la cámara FPV, que es mucho más pequeña, y se fija en la parte frontal superior.

La incorporación de esta cámara gimbal y de baterías algo más grandes para aumentar la autonomía, hacen que el centro de masas de la aeronave se modifique, aunque siempre manteniéndolo bajo las especificaciones del fabricante, que son entre los 14 y

I. Memoria

16 cm desde el borde de ataque del ala en su unión con el fuselaje. Las dimensiones más relevantes de la aeronave se muestran en la siguiente tabla.

Envergadura	2.5 m
Longitud	1.44 m
Superficie alar	0.54 m ²
Cuerda en la raíz	0.26 m
Aspect ratio ala principal	11.57
Masa	8 kg

Tabla 3.1. Dimensiones de la aeronave

3.1. Componentes

La planta propulsiva de la aeronave consta de cuatro motores para el vuelo en modo multirotor y un motor de cola para el vuelo como ala fija. Los motores VTOL son cuatro MN505-S KV260 de T-Motor, con hélices de 16x5.5 pulgadas y ESCs Flame 60A HV de T-Motor. Sus características principales son [9]: $K_V = 260$, $V_{max} = 47.16 V$, $\eta_{max} = 0.86$, $T_{nom} = 73.57 N$, $D = 0.406 m$. El motor de cola es un ECO 4120C V2 de Dualsky con hélice de 15x8 pulgadas y ESC FlyFun 120A V5 de Hobbywing. Sus características principales son [10]: $K_V = 500$, $V_{max} = 21 V$, $\eta_{max} = 0.88$, $T_{nom} = 59.1857 N$, $D = 0.381 m$.

El sistema de alimentación necesario para que funcionen estos motores consta de cuatro baterías divididas en dos grupos, dos baterías LiPo en serie 6S de 5000mAh para los motores VTOL y dos baterías Li-Ion en paralelo 6S de 16000mAh, para el vuelo en ala fija que incrementan notablemente la autonomía. Claramente las baterías de los 4 motores verticales son mucho más pequeñas y con mucha menos autonomía que las otras, esto se debe a que el vuelo en modo multirotor solo se va a utilizar para el despegue hasta una altura determinada y para el aterrizaje. Si fuera necesario volar más alto, este ascenso se realizaría en vuelo en ala fija.

La placa controladora de vuelo utilizada es un Cube Orange de CubePilot que se monta en una placa portadora como se ve en la figura [3.2].



Figura 3.2. Controladora Cube Orange de Cube Pilot. [19]

Las principales características de esta controladora de vuelo son:

Procesador

- Procesador STM32H743ZI de 32 bits ARM Cortex M7.
- RAM de 1MB y Flash de 2MB.
- Procesador adicional STM32F103 para redundancia.

Sensores

- 2 barómetros MS5611 de 10 cm de resolución en altitud.
- 3 IMU diferentes con acelerómetro y giroscopio, una de ellas además incorpora magnetómetro.

Alimentación

- Redundancia de entradas de alimentación con conmutación automática frente al fallo.
- Protección ante niveles bajos de batería y frente a sobre corrientes.

Puertos

- 14 salidas PWM (Pulse Wide Modulation). 8 principales y protegidas ante fallos y 6 auxiliares. Utilizadas para servos y ESC.
- 2 puertos CAN.
- 2 puertos GPS.
- Antena ADS-B.
- Puerto I2C.
- Micro USB.
- Ranura para tarjeta SD donde se almacenan logs de vuelo.
- Entrada de radio RC IN.
- 2 puertos de salida de telemetría para conectar con la estación de control en tierra.
- Puerto convertidor de analógico a digital.

I. Memoria

Dimensiones

- Placa portadora y cube Orange: 94.5x44x17.3 mm.
- Masa de 73 gramos.

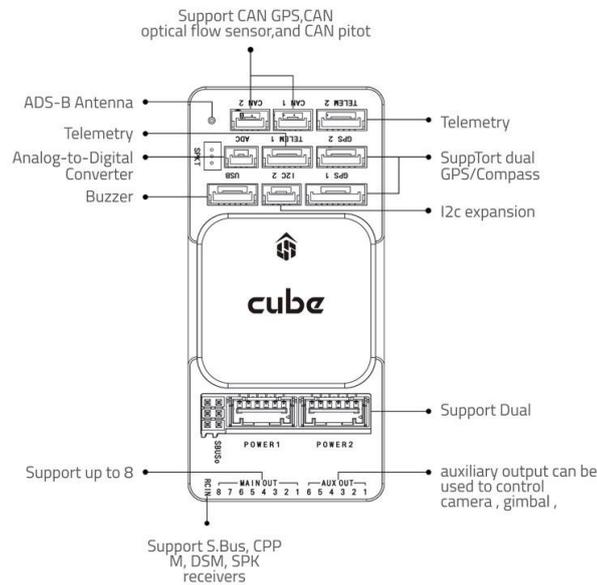


Figura 3.3. Diagrama de puertos de Cube Orange. [20]

Esta controladora de vuelo también dispone de función ADS-B, aunque no se va a utilizar debido a la incompatibilidad con el firmware del autopiloto.

En cuanto a la unidad de posicionamiento, se emplea el Here3, también de CubePilot. Se trata de un sistema GNSS de alta precisión, pudiendo llegar a pocos centímetros, compatible con GPS, GLONASS y BeiDou. Cuenta con un procesador STM32F302 y una unidad IMU ICM20948. Este dispositivo se colocará en la parte superior de la aeronave para mejor recepción de señal y el protocolo de comunicación empleado es CAN.



Figura 3.4. Dispositivo Here3 montado sobre la parte superior de la aeronave.

También cuenta con un tubo de Pitot que se conecta al puerto IC2 de la placa controladora, al tratarse de un VTOL que va a volar en ala fija es imprescindible conocer la velocidad aerodinámica de la aeronave en todo momento.

Las conexiones de los servos responsables de las superficies de control y de los ESC de los motores con la placa controladora se realizan a partir del código del autopiloto. Dependiendo del airframe utilizado se establecen unas salidas predeterminadas para la placa controladora donde cada salida corresponde con un servo o un ESC concreto.

3.2 Autopiloto

El autopiloto, combinando el software y el hardware, se encarga de generar las señales e instrucciones y enviarlas a los distintos elementos de control de la aeronave con el fin de automatizar ciertas operaciones de acuerdo a lo que el piloto ha planificado previamente. Actualmente, estas operaciones llevadas a cabo de forma autónoma han de estar supervisadas en todo momento por el piloto, siendo este capaz de tomar el control de la aeronave en cualquier momento, ya sea porque así lo desea el piloto o por seguridad debido a algún tipo de fallo, ya sea del autopiloto o de otro sistema.

También proporciona seguridad cuando el enlace de datos entre el piloto y la aeronave se corta o no funciona correctamente, ya que permite activar un modo de seguridad que podemos configurar para que la aeronave aterrice o vuelva al punto de despegue o cualquier otro de forma totalmente autónoma.

Es capaz de realizar todo tipo de operaciones, despegues, aterrizajes, seguir puntos de una ruta, además de realizar tareas específicas durante estas operaciones como seguimiento de objetivos o captura de imágenes. Permite llevar a cabo todas estas misiones además de realizar el control aerodinámico. Para ello ha de ser capaz de conocer exactamente la actitud y la posición de la aeronave, lo que consigue mediante sensores inerciales IMU, sensores infrarrojos o GPS. Las medidas de estos sensores se utilizan en las diferentes estrategias de control de cada variable a controlar con el fin de conseguir los valores adecuados para seguir la trayectoria marcada por el piloto.

También es fundamental en el equilibrado del vuelo, ya que, aunque estemos en modo manual, u otros modos que no sigan una trayectoria especificada por el piloto de forma autónoma, el autopiloto es capaz de estabilizar el vehículo y que este realice maniobras suaves y controladas en todo momento, además de contrarrestar los efectos del viento y de ráfagas que podrían comprometer dicha estabilidad.

Toda esta información del autopiloto se almacena en el firmware, que permite controlar los circuitos electrónicos y enviar las señales correspondientes a los controladores. Según el tipo de aeronave y de hardware de autopiloto, se necesita un firmware específico que sea compatible con los elementos externos y que permita un buen funcionamiento. Existen diferentes tipos de firmwares de pilotos automáticos, puede que sean comerciales, ya sean de empresas o de tipo software libre, o que sea privados tanto de empresas como de los países para uso en UAS militares. Existen diferentes empresas que comercializan sus drones con su hardware y firmware específicos como DJI o Embention, que comercializan componentes para drones, tanto autopilotos como estaciones de control o cámaras. Entre los firmwares libres destacan dos proyectos,

I. Memoria

ArduPilot y PX4, que cuentan con extensa documentación y que van mejorando y adaptándose a las nuevas aeronaves incluyendo cada vez más funcionalidades.

En este trabajo se ha optado por utilizar uno de estos firmwares de código abierto. Las principales ventajas, a parte de su precio es que permiten adaptarse a una gran cantidad de hardware y componentes del UAS, ya que se va adaptando e incluyendo cada vez más controladoras de vuelo, por ejemplo, sin importar el fabricante. Esto hace que sea un gran activo para el público en general y para el ámbito educativo.

ArduPilot se lanzó años antes que PX4, por lo que al inicio solían implementar nuevas mejoras y versiones de sus autopilotos más rápidamente que PX4, además de realizar pruebas en vuelo de todas sus versiones. Existen multitud de foros donde encontrar respuestas a los problemas que van surgiendo a medida que se configura el autopiloto. En estos foros se puede encontrar también información directamente relacionada con el software principal con el que se realiza la planificación del vuelo y configuración del autopiloto y demás sistemas embarcados, MissionPlanner. La realidad es que la comunidad de PX4 ha tenido un gran crecimiento a lo largo de los años y ambos autopilotos, exceptuando algunas funciones específicas, son capaces de llevar a cabo las mismas tareas con la gran mayoría de drones que existen. Aunque PX4, sí que dispone de una función para añadir un receptor ADS-B, este ha de ser externo para poder configurarlo y al estar incluido en el Cube Orange no se puede utilizar, cosa que si podría hacerse en ArduPilot. La realidad es que esta función a pesar de ser interesante, no es de gran utilidad por el momento ya que no se va a volar en un entorno en el que esta funcionalidad sea necesaria para evitar accidentes.

Sin embargo, se ha elegido el firmware de PX4 debido a la mayor flexibilidad en el código que permite, con lo que se podrá hacer mejores simulaciones de vuelo. Además, aunque en ArduPilot también se podría realizar, PX4 cuenta con un *airframe* (estructura de aeronave de referencia) específico para la aeronave a la cual se lo vamos a implementar, con los controladores y mezcladores específicos para modelar la cola en forma de A, que es su mayor diferencia frente a un avión VTOL convencional o *Quadplane*. El software de planificación de vuelo y configuración del autopiloto que se va a usar es el QGroundControl, como no podía ser de otra manera, que ofrece una interfaz gráfica a la que es fácil adaptarse. Además, el manual de PX4 cuenta con todo tipo de instrucciones que se pueden consultar donde aparecen todas las funciones disponibles tanto del firmware como de su interfaz QGroundControl.

3.3. PX4

Se trata de un firmware de *Dronecode Foundation*, una fundación sin ánimo de lucro donde se incluyen proyectos de desarrollo de código abierto para el sector de los drones, ya sean autopilotos como PX4, softwares de planificación de vuelos como QGroundControl, hardware tipo las series de placas Pixhawk o protocolos de comunicaciones específicos como MAVLink. Antes de continuar con PX4, se va a explicar brevemente que es y en que consiste este protocolo de comunicación.

MAVLink es un protocolo muy ligero y de mensajes para comunicaciones con drones y entre componentes a bordo de estos. Es de código abierto LGPL, que garantiza la libertad de compartir y modificar el software asegurando que sea libre para todos los usuarios. Formula una regla de transmisión y recepción para la información que se transmite con frecuencia entre estaciones de control y aeronave. Estas reglas se definen con parámetros en forma de bibliotecas de mensajes. Mediante este protocolo se comunican hasta 255 sistemas concurrentes, ya sean aeronaves, estaciones de control, antenas o cualquier otro tipo de hardware.

El software de PX4 cuenta con una gran variedad de aeronaves de referencias, conocidas como *airframes*. Estos marcos de referencia disponen de características específicas que nos ayudan a configurar la aeronave, como son el tipo de aeronave, si es de ala fija, multirotor o VTOL, el número de motores que tiene o la disposición de éstos en la aeronave. A medida que el software ha ido evolucionando se han ido incorporando cada vez más *airframes*, cada vez más específicos, hasta el punto de llegar a contemplar prácticamente todas las configuraciones posibles de drones.

El piloto automático se encarga del control de servos y ESC para mantener la actitud del dron, marcada por el piloto de forma directa, o indirecta mediante la planificación del vuelo. Para ello el autopiloto emplea controladores PID, basados en la corrección de las variables de estado, y corrigiendo los errores entre las medidas o estimadas y los valores objetivo. No todos los controladores del autopiloto son PID, también hay otros que son P, PI, o PD, además de combinaciones de estos. Dado que los VTOL tiene dos modos de vuelo completamente diferentes, multirotor (MC) y ala fija (FW), estos controladores habrá que ajustarlos para los dos tipos de vuelo, y posteriormente reajustar la transición entre estos, donde se combinan las acciones entre ambos modos de vuelo. A continuación, se va a explicar cómo funcionan los controladores en ambos modos de vuelo.

3.3.1. Controladores MC

Las actuaciones de este modo de vuelo consisten únicamente en los cuatro motores, que variarán la velocidad de giro para ajustar la actitud de la aeronave a la deseada. El mezclador es el que se encarga de transformar las actuaciones requeridas por el autopiloto en las señales que envía a los ESC para que cambien su velocidad de giro. En la figura [3.5] se muestra un diagrama de bloques donde se modelan todos los controladores de este modo de vuelo.

I. Memoria

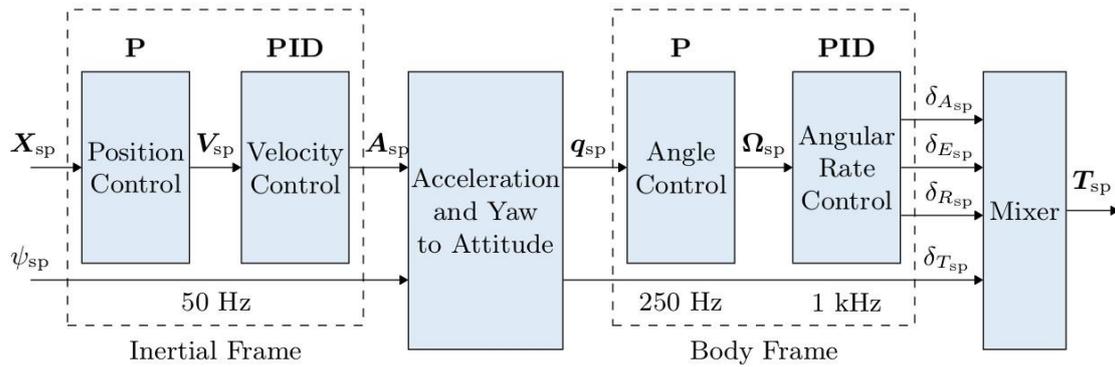


Figura 3.5. Controladores del modo Multicóptero.[6]

Para definir las variables estimadas o medidas, se utilizan los valores de los sensores embarcados pasándoles un filtro Kalman extendido para procesar las mediciones y proporcionar finalmente los valores estimados.

Controladores de posición y velocidad

El controlador de posición es de tipo proporcional, multiplicando la variación de posición directamente y obteniendo la velocidad necesaria, siempre dentro de los límites impuestos por el usuario, ya sea de forma arbitraria o por limitaciones de la aeronave. Este controlador solo se usa cuando queremos volar a punto fijo, ya que, si queremos volar a una velocidad determinada, ya disponemos de esta para entrar directamente al controlador de velocidad. Los diagramas de bloques de estos dos controles son los siguientes.

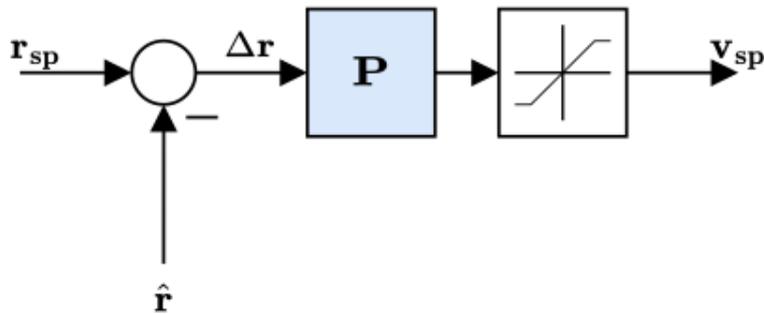


Figura 3.6. Diagrama de bloques del controlador de posición en MC. [6]

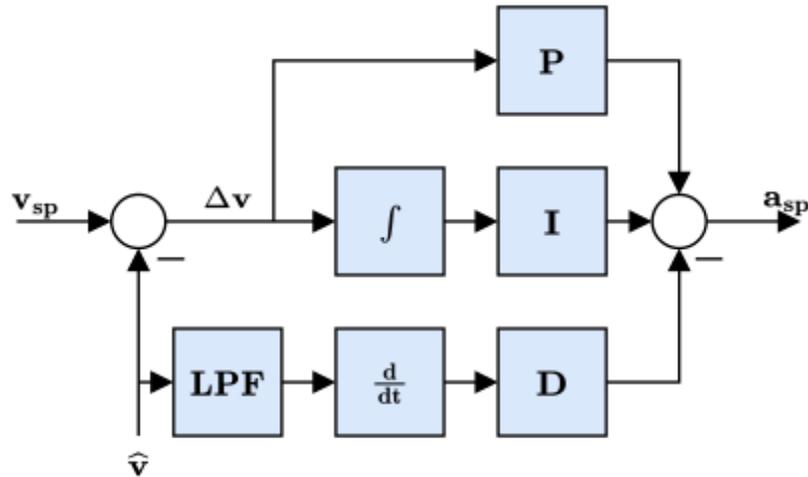


Figura 3.7. Diagrama de bloques del controlador de velocidad en MC. [6]

Siendo las variables sp , *setpoint*, las correspondientes a valor deseado, mientras la V y \hat{v} las variables estimadas. LPF hace referencia a un filtro de paso bajo que se utiliza en la parte derivativa con el fin de reducir el ruido.

Controlador de actitud

El controlador de actitud hace uso de cuaterniones, realizando operaciones con estos, cosa que no se va a detallar ya que se trata de algo complejo que no atañe a este trabajo y que lo implementaron en PX4 a partir de [7]. Con la aceleración y ángulo de guiñada deseados el bloque proporciona la conversión de aceleración a empuje y de empuje a actitud. Finalmente, el controlador es de tipo proporcional como se muestra en el diagrama de la figura [3.8].

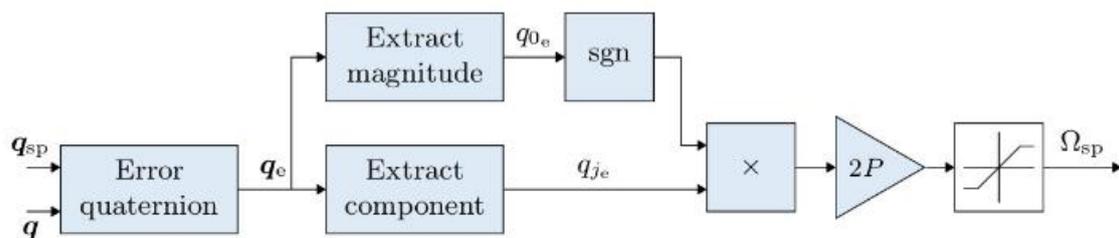


Figura 3.8. Diagrama de bloques del controlador de actitud en MC. [6]

Controlador de velocidad angular

El caso de la velocidad angular es idéntico al de velocidad, exceptuando que se trata de un K-PID. La diferencia reside en que han de tener el valor unidad asignado a la K o a la P. Matemáticamente es equivalente, aunque la ventaja es que permiten reajustar los valores manteniendo la I y la D. En la figura [3.9] se ve el diagrama de bloques de este controlador.

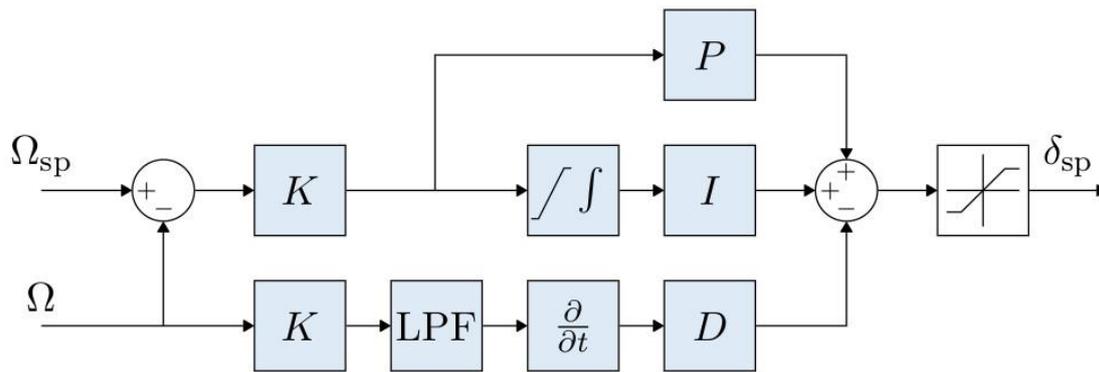


Figura 3.9. Diagrama de bloques del controlador de velocidad angular en MC. [6]

3.3.2. Controladores FW

Los controladores de este modo son ligeramente diferentes y se dividen en dos tipos, el TECS (*Total Energy Control System*) y el control de actitud.

TECS

Se trata de código implementado mediante una librería que se utiliza para el modo de control de posición de ala fija. Para realizar el control se basa en la energía total de la aeronave, la suma de energía cinética y potencial. Este control tiene dos partes una dedicada al control de la energía total, mediante el control del empuje y otra encargada de mantener el balance entre cinética y potencial mediante el ángulo de cabeceo.

Se recibe como entrada la velocidad aerodinámica y la altitud deseada y tiene como salida el porcentaje de acelerador o *throttle* y el ángulo de cabeceo deseado o necesario para conseguir las variables de entrada. Las ecuaciones energéticas que llevan a la obtención de las variables de salida se resumen a continuación.

$$E_T = \frac{1}{2} m V_T^2 + m g h \quad (3.1)$$

$$\dot{E}_T = m V_T \dot{V}_T + m g \dot{h} \quad (3.2)$$

$$E_T = \frac{\dot{E}_T}{m g V_T} = \frac{\dot{V}_T}{g} + \frac{\dot{h}}{V_T} = \frac{\dot{V}_T}{g} + \sin \gamma \quad (3.3)$$

Simplificando el segundo término de la ecuación 3.3, donde la velocidad vertical dividida la horizontal es el seno de la trayectoria y asumiendo bajos ángulos de cabeceo, el seno del ángulo es igual al ángulo.

$$T - D = m g \left(\frac{\dot{V}_T}{g} + \sin \gamma \right) = m g \left(\frac{\dot{V}_T}{g} + \gamma \right) \quad (3.4)$$

$$\Delta T = mg \left(\frac{\dot{V}_T}{g} + \gamma \right) \quad (3.5)$$

Estas primeras ecuaciones corresponden al bucle de control de empuje, mientras que las mostradas a continuación el control del ángulo de cabeceo.

$$\dot{B}_T = mg\dot{h} - mV_T\dot{V}_T \quad (3.6)$$

$$\dot{B} = \gamma - \frac{\dot{V}_T}{g} \quad (3.7)$$

Con las variables deseadas de salida calculadas, estas pasan a un controlador PID habitual. La complejidad del cálculo es notoria ya que ambas salidas afectan de distinta forma a ambas entradas, el aumento de cabeceo hará disminuir la velocidad y aumentar la altura, mientras que mayor empuje resultará en un aumento de la velocidad y también de la fuerza de sustentación. Es por esto que es tan importante tener buenas mediciones de velocidad y de actitud, especialmente cabeceo, ya que pueden comprometer el control de la aeronave si fallan.

Control de actitud

De forma similar al controlador en multirotor, dispone de un bucle externo que calcula las variaciones que posteriormente entran a otro bucle calculando así las aceleraciones que se transmitirán a los mezcladores y de ahí a los servos. El ángulo de cabeceo deseado viene directamente del control TECS, mientras que los de guiñada y alabeo provienen de una ley de navegación que PX4 llama L1.

Dado que el control de superficies es más efectivo a mayor velocidad y menos a menor, se utiliza el valor de velocidad de crucero para escalar el control de las superficies. Se utiliza un factor *feedforward* que sirve para compensar la amortiguación por fenómenos aerodinámicos. Por último, la guiñada se controla a partir del alabeo y cabeceo para obtener giros coordinados en todo momento que minimicen las aceleraciones laterales. El algoritmo que controla los giros coordinados se basa en:

$$\dot{\Psi}_{sp} = \frac{g}{V_T} \tan \phi_{sp} \cos \theta_{sp} \quad (3.8)$$

En la siguiente figura se muestra el diagrama de bloques del control de actitud.

I. Memoria

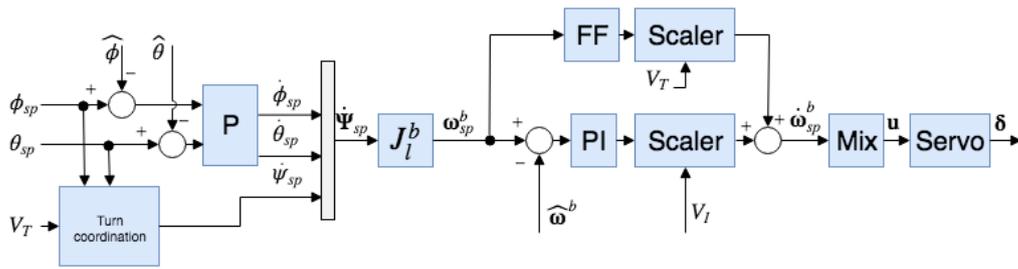


Figura 3.10. Diagrama de bloques del controlador de actitud en FW. [6]

Todos estos controladores descritos en los últimos dos subpuntos se entremezclan cuando se realiza la transición entre ambos modos de vuelo, FW y MC, formando el siguiente diagrama de bloques donde en cierta medida actúan ambos controles.

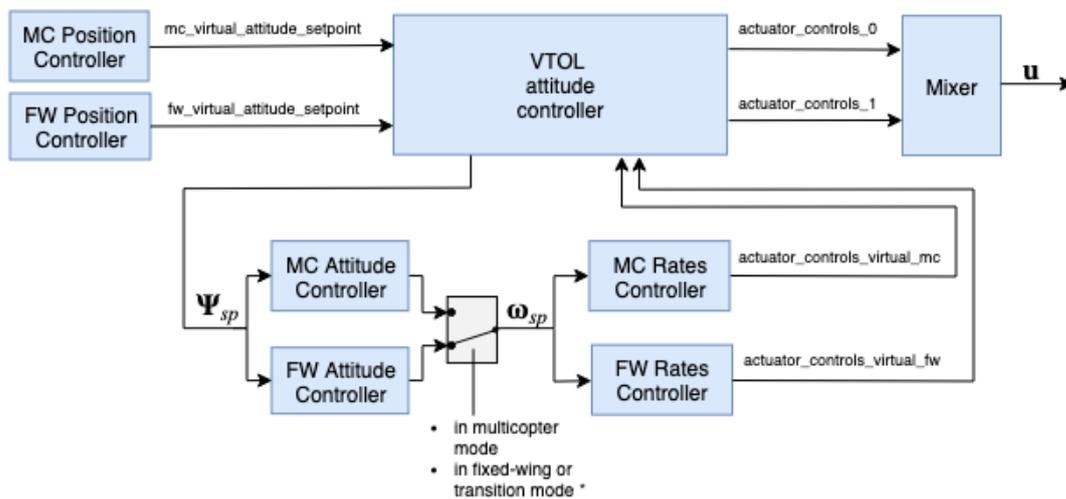


Figura 3.11. Diagrama de bloques del controlador de actuaciones VTOL. [6]

3.3.3. Modos de vuelo

PX4 dispone de numerosos modos de vuelo cuando se trata de un VTOL ya que debe considerar ambos modos de operación, vuelo en multirrotor y en ala fija. Aunque hay algunos podría pensarse que son iguales, las acciones de la aeronave son muy diferentes, diferenciaremos entre modos manuales y automáticos.

Modos manuales

- Position: Cuando se sueltan los mandos o *joysticks*, en MC se frena y mantiene la posición y altitud actual compensando los vientos y otras fuerzas, mientras que en FW la aeronave vuela en línea recta respecto a tierra (compensando las fuerzas del viento) manteniendo la altitud.
- Altitude: Cuando los controles se quedan centrados, en MC se frena y mantiene la altitud, aunque sin compensar la acción del viento, mientras que en FW, volará en línea recta manteniendo la altitud, también sin compensar la acción del viento.

- **Stabilized:** Al soltar los *joysticks* en MC mantiene posición, aunque se necesita controlar el empuje para mantener altitud no corrigiendo los efectos del viento; en FW vuela en línea recta y manteniendo altitud, pero no corrige cambios en orientación y altitud por acción del viento.
- **Manual:** En MC, la aeronave se comporta igual que en *stabilized*; en FW el piloto controla por completo la aeronave con los mandos, lo que resulta complejo, aunque cuando se realizan virajes y se vuelve a centrar el *joystick* la aeronave vuelve a volar en línea recta.

Modos automáticos

- **Hold:** En MC se mantiene posición y altitud compensando vientos; en FW la aeronave realiza círculos alrededor de la posición actual. Esto permite realizar pausas durante las misiones.
- **Return:** Realiza la misma acción en ambas configuraciones, vuelve a una altura predeterminada a la posición marcada como casa y allí cambia a modo hold si no se ha establecido el aterrizaje posteriormente en la planificación de la misión.
- **Mission:** Misma acción para ambas configuraciones, sigue la misión establecida en QGroundControl. Si se tocan los *joysticks* pasa a modo position y se pueden realizar pausas activando el modo hold.
- **Take-off:** Este tipo de aeronaves, VTOL, solo permite realizar despegues en modo MC, por lo que cuando el vehículo está listo y armado se procede al despegue hasta la altitud predefinida por el usuario pasando a modo position al finalizarlo.
- **Land:** Al igual que con el despegue, solo se puede aterrizar en modo MC por lo que, si volamos en FW, se hace transicionar a MC primero y después aterrizar, desarmándose al finalizar la maniobra.
- **Off-board:** Permite mandar instrucciones al vehículo mediante MAVLink a través de la consola de la estación de control en tierra haciendo posible la modificación de parámetros.

4. Sistema de control y navegación

El gran atractivo de este proyecto es su sistema de telemetría de largo alcance, capaz de controlar el UAS hasta a 200 km. Se trata de un sistema de la empresa española DMD (Digital Micro Device) que cuenta con transmisores y receptores embarcados y en tierra, que canalizan toda la información hacia una estación de control en tierra donde poder controlar el vuelo.

Cabe destacar la redundancia de los sistemas, tanto embarcados como en tierra que permiten aumentar la seguridad operacional de las misiones, algo muy importante cuando realizamos vuelos de largo alcance.

4.1 Componentes embarcados

Los receptores RXLRS

Tienen un alcance de hasta 200 km y se comunican con la estación de tierra mediante protocolo MAVLink, lo que permite comunicar al autopiloto con la estación de control en tierra, esto se hace mediante los puertos TELEM1 y RC IN de la controladora Cube Orange. El puerto TELEM 1 se encarga de transmitir la información al receptor y este la lleva hasta tierra, mientras que el RC IN se conecta al puerto SPPM del RXLRS. Dicho puerto es una salida de los pulsos de todos los servos, permitiendo la conexión con el autopiloto y aumentando el número de canales disponibles. En vez de salir los pulsos de los servos por cada conector de canal, todos los pulsos salen por un único conector, permitiendo reducir el cableado. Se utilizan dos RXLRS uno trabaja a 868 MHz (Principal) y el otro a 433 MHz (secundario). Esto permite que el secundario corrija los posibles errores recibidos por el principal, lo que resulta en un aumento de la sensibilidad y, por ende, el alcance y seguridad, ya que si uno de los dos fallara el sistema podría continuar funcionando con uno solo. Estos receptores se alimentan a 5 voltios y 4 amperios, provenientes de un transformador DC/DC, sin embargo, solo uno de ellos está conectado a esta fuente de alimentación, el otro recibe la alimentación y la información mediante el cable mediante un RCBus que conecta ambos receptores entre ellos.

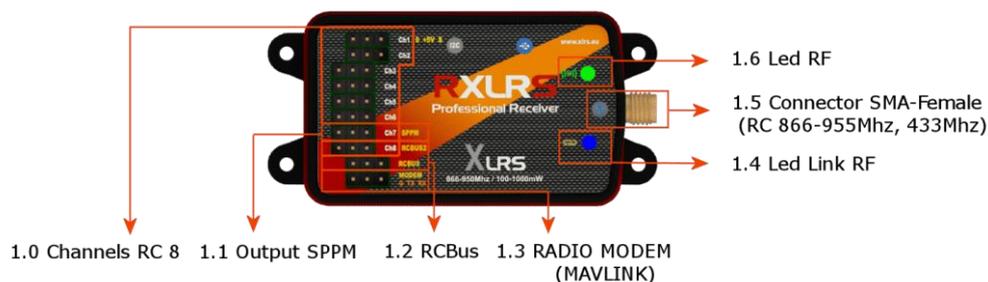


Figura 4.1. Diagrama de conexiones del receptor RXLRS. [11]

Transmisores de video XOSD3B

Los transmisores OSD (*On Screen Display*) permiten enviar las imágenes de la cámara FPV (*First Person View*), además de las imágenes muestran una serie de iconos que ayudan al piloto a conocer el estado de la aeronave en todo momento sin tener que apartar lo ojos de la pantalla. Los transmisores de video, uno de 1.2 GHz y el otro de 2.4 GHz, emiten la información hasta los receptores en tierra, posteriormente estos receptores envían la información a 5.8 GHz al monitor integrado en la estación de control en tierra, o también se pueden conectar directamente los receptores con el monitor por cable, lo que reduce el ruido de la imagen y aumenta la nitidez de esta. La información sale de la controladora de vuelo por el puerto TELEM 2, que se conecta a ambos transmisores, estando conectados entre sí mediante RCBus y uno de ellos al receptor RXLRS mediante RCBus también, con lo que todos los sistemas embarcados quedan conectados entre ellos. Ambos transmisores de video se alimentan mediante un transformador DC/DC de 8 voltios.



Figura 4.2. Transmisores de video embarcado XOSD3B. [11]

4.2. Componentes en tierra

Estación de control GCSD4 V2

La estación de control en tierra se encarga de recibir todos los datos de la aeronave y ser la interfaz entre el piloto y ésta con el fin de controlarla en modo manual o planificar misiones que impliquen el uso del piloto automático. La estación cuenta con los mandos necesarios para pilotar, con una pantalla integrada donde se reciben las imágenes de la cámara FPV o de un gimbal, y además cuenta con una pequeña pantalla en la parte inferior donde viene preinstalado el Windows 10. Es en esta pantalla donde se instalan los softwares de planificación de vuelo oportunos, además de otro programa específico, del sistema de comunicaciones, DMD Studio, donde se pueden modificar ciertos parámetros en cuanto a la transmisión de datos. El monitor integrado tiene puertos de conexión de cable y un receptor de 5,8 GHz según como queramos recibir la señal del receptor XVID. Se conecta mediante un cable Ethernet a la antena inteligente redundante SMRBTS del cual hablaremos a continuación.



Figura 4.3. Estación de control en tierra GCSD4 V2. [11]

Antena inteligente redundante SMRBTS

Este transmisor permite comunicar los receptores de radiocontrol y telemetría embarcados en la aeronave con la estación de control en tierra. Esta montado sobre una antena bidireccional BQ89 de 10-12 dBi, mejorando el rendimiento, alcance y la seguridad en la estación de control. Se trata de un sistema redundante por lo que permite la transmisión en 868 MHz y 433 MHz, como los embarcados. El transmisor de 868 MHz está en la propia antena bidireccional, mientras que el de 433 MHz emplea una segunda antena tipo moxon conectada mediante un cable de longitud menor a los 0,5 metros, despreciando así las pérdidas del cable. Mediane el radiomodem interno permite el control de radio, MAVLink y protocolos transparentes.



Figura 4.4. Receptor redundante y antena bidireccional. [11]



Figura 4.5. Antena secundaria 433 MHz.

Receptor RXVID3

El sistema cuenta con dos receptores RXVID3 de video analógicos, uno que recibe la señal 1,2 GHz y el otro a 2,4 GHz. Esto permite seleccionar el receptor con mejor recepción de señal y pudiendo incorporar una antena omnidireccional para vuelos de menor distancia y una direccional cuando nos alejemos más del punto de control. Cada uno de estos cuenta con dos antenas, una para recibir a la frecuencia correspondiente y otra para transmitir lo recibido a la estación de control a 5,8 GHz, concretamente al monitor integrado. Además, cuentan con salidas RCBus para conectarse directamente al monitor, evitando la transmisión de datos por radiofrecuencia y mejorando la calidad de las imágenes.

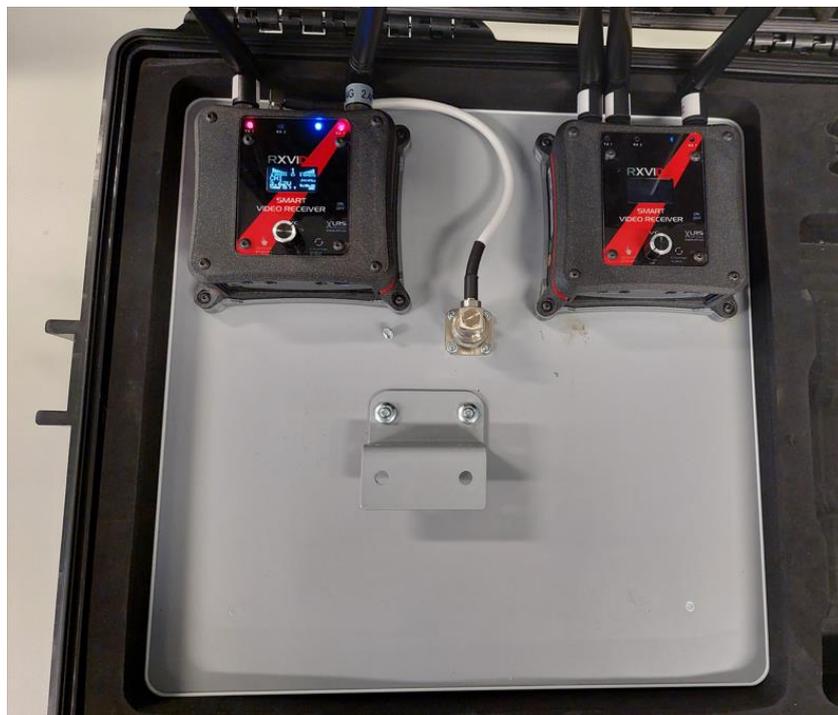


Figura 4.6. Receptor de video RXVID3 de 2.4 GHz y 5.8 GHz.

4.3 Diagrama de conexiones

A continuación, se muestran dos figuras donde se ve cómo se conectan todos los elementos del sistema de comunicaciones de largo alcance tanto en el lado aire como en tierra.

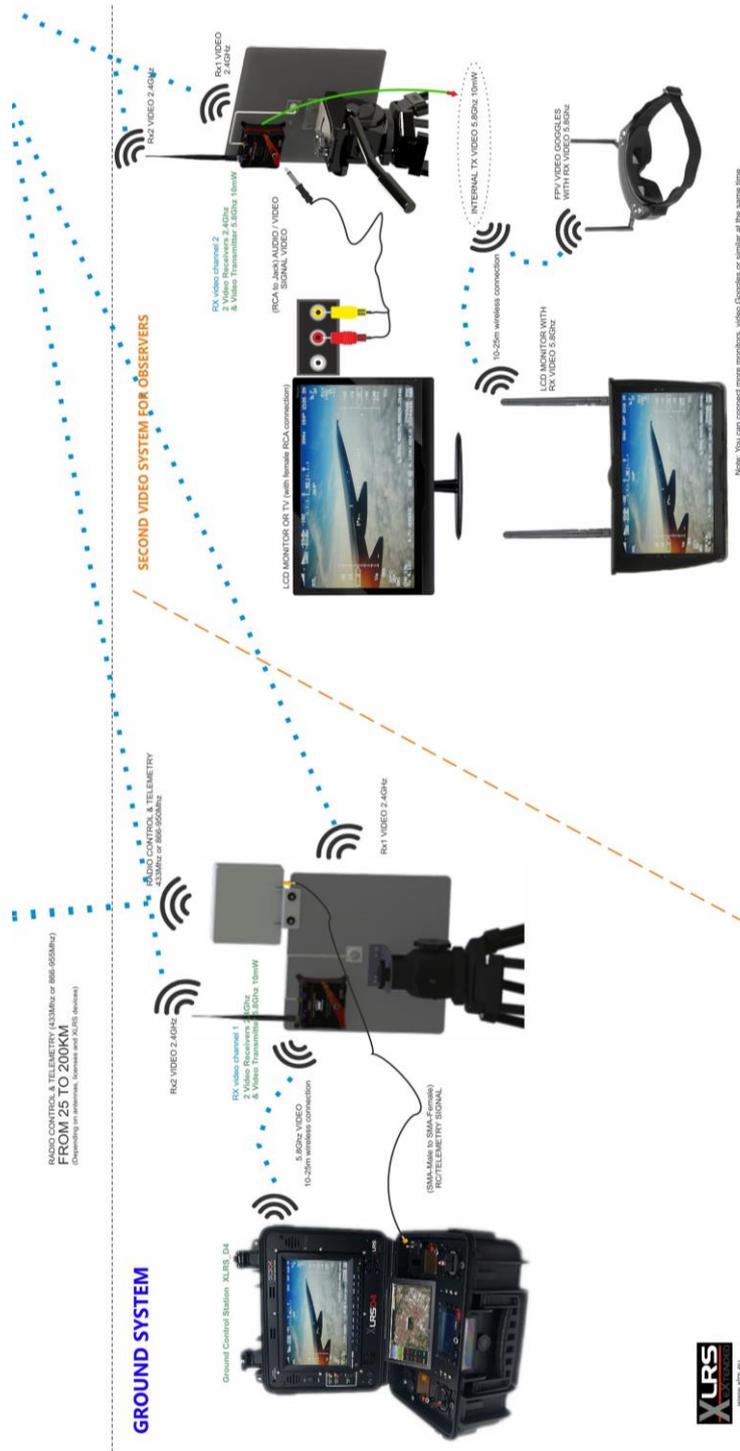


Figura 4.7. Diagrama de conexiones del lado aire del sistema de telemetría redundante XLRs. [11]

4. Sistema de control y navegación

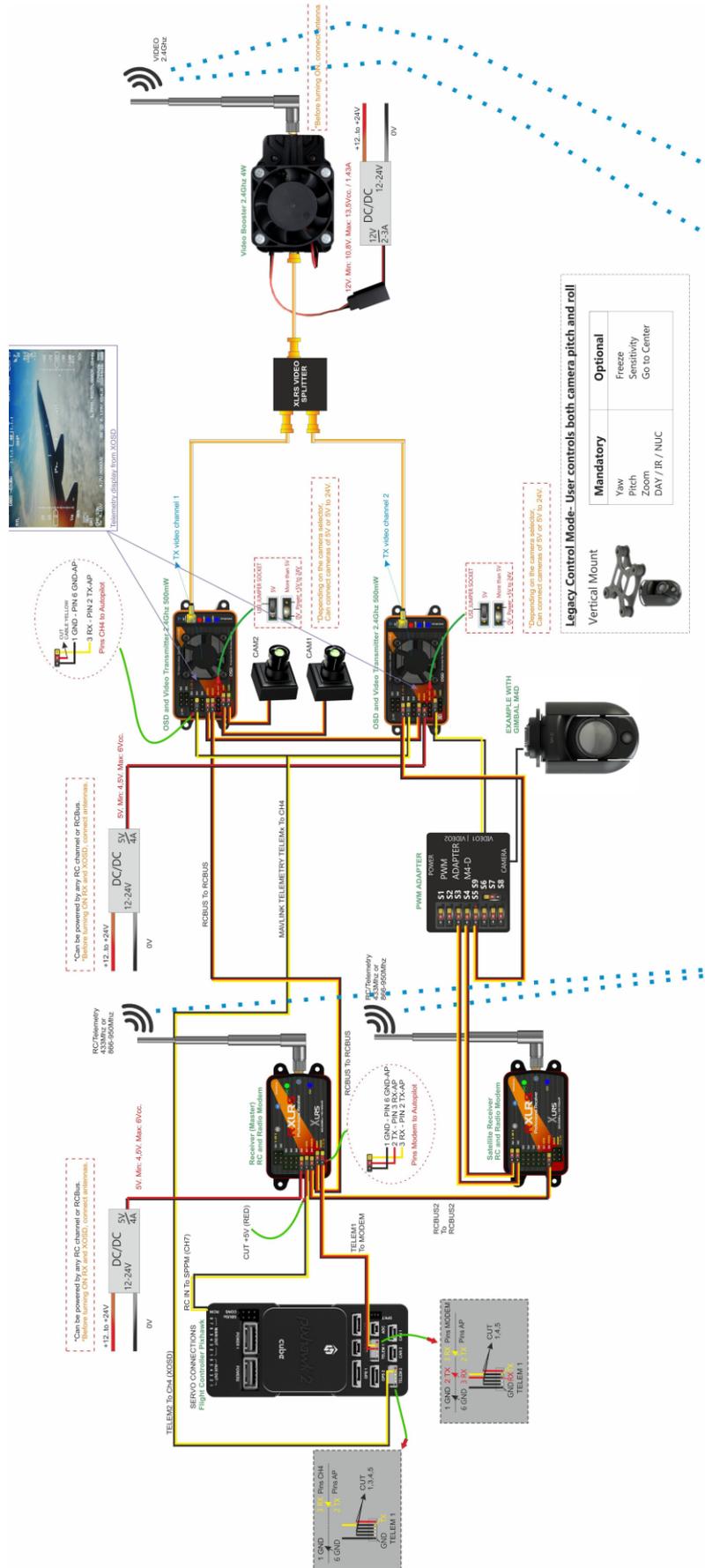


Figura 4.8. Diagrama de conexiones del lado tierra del sistema de telemetría redundante XLRs. [11]

5. Simulaciones y preparación para el vuelo

Una vez se tienen todos los componentes y todos los sistemas necesarios para el vuelo montados queda comprobar que el autopiloto funciona correctamente, lo cual es ciertamente peligroso, ya que de no ser así podríamos poner en peligro la aeronave y cualquier cosa que este en el entorno del vuelo.

Por ello se ha decidido hacer simulaciones de vuelo donde poder probar el autopiloto y la dinámica de la aeronave, incluyendo el proceso característico de este tipo de aeronave, las transiciones entre vuelo en multirotor y vuelo en ala fija.

Estas simulaciones serán de dos tipos, primero SITL (*Software In The Loop*) que consiste en un entorno de simulación donde la aeronave, junto con sus dinámicas de movimiento, es controlada por el software del autopiloto, que se instala en el ordenador. Y luego HITL (*Hardware In The Loop*), donde en esta ocasión es el propio firmware del autopiloto de la propia aeronave el que controla los movimientos de esta, mientras que las fuerzas aerodinámicas, de los motores y demás, continúan simulándose por el programa de simulación.

Para realizar las simulaciones se tuvo que generar un modelo 3D de la aeronave. Las herramientas de SITL disponen de modelos matemáticos para simular las fuerzas aerodinámicas, inerciales, empujes de los motores, etc. Por eso se realizó un estudio con el modelado 3D de la aeronave, para poder conocer coeficientes aerodinámicos necesarios e implementarlos en las simulaciones. También se cargó la aeronave con las baterías que van a utilizar para que el peso de la aeronave y su dentro de masas estén correctamente en las simulaciones. Esto se llevó a cabo en [1].

5.1. Software in the loop

Este tipo de simulaciones representan una gran forma de comprobar el funcionamiento de nuestro autopiloto sin tener que correr el riesgo con la aeronave real, además que permite adaptarse al pilotaje de la aeronave, ya que podemos conectar un mando con joysticks al ordenador, incluso los de una videoconsola.

Cabe destacar que para llevar a cabo todas las simulaciones se ha utilizado el sistema operativo Linux, instalado en un SSD externo. Aunque PX4 también soporta Windows 10, recomiendan usar Linux. La herramienta de simulación 3D con la cual se va a trabajar será Gazebo, ya que a pesar de que se pueden usar otras como jMAVSim o AirSim, Gazebo es la única compatible con aeronaves de tipo VTOL y la recomendada para las simulaciones con PX4. Gazebo solo se puede instalar en Linux o MacOS, lo que también fue una restricción para tener que emplear Linux.

Gazebo

Se trata de un entorno de simulación 3D para robots autónomos con grandes funcionalidades en pruebas de evasión de objetos, aunque en nuestra simulación no nos centraremos en esto ya que nuestra aeronave no lleva sensores de este tipo.

Las misiones se ejecutan desde el programa de estación de control en tierra (QGroundControl) y se conecta a PX4 mediante un puerto UDP. Esto permite ver la ejecución del vuelo en Gazebo mientras lo controlamos desde QGroundControl.

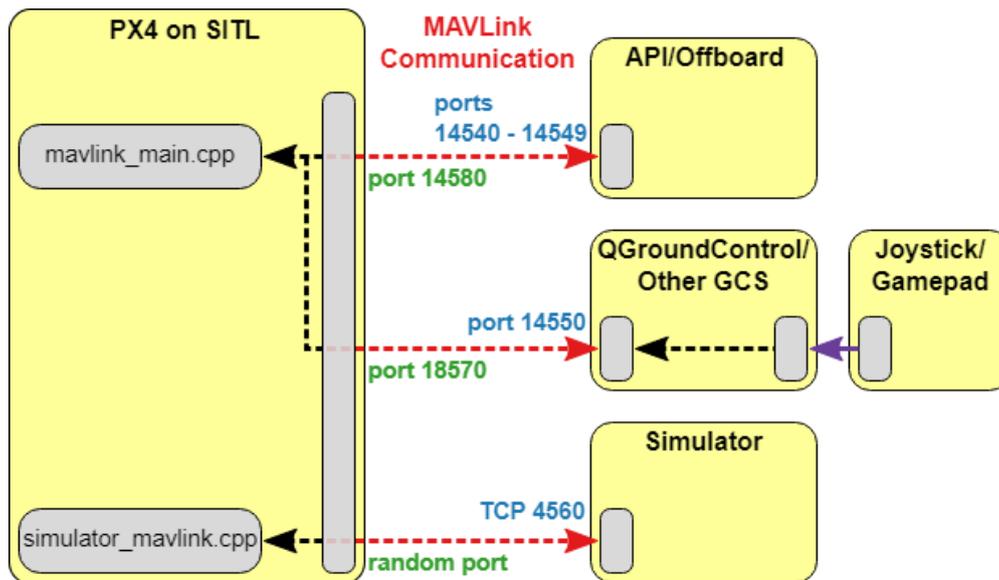


Figura 5.1. Comunicaciones SITL entre simulador estación de control y PX4. [21]

Gazebo se encarga de realizar la simulación de los sensores y el entorno en el que vuela, que se puede modificar, aunque para estas pruebas eso es irrelevante. La información de los sensores se transmite vía MAVLink de Gazebo a PX4 por el puerto TCP 4560, mientras que también se comunica con la estación de control simultáneamente, mediante el puerto UDP 14550, como se muestra en el diagrama de comunicaciones de la figura [5.1]. También dispone de archivos que simulan los efectos aerodinámicos y los rotores, relacionándolos con la geometría asignando ciertas propiedades a cada parte de ésta, lo que se detallará más adelante.

5.1.1. Entorno de desarrollo y simulación

En primer lugar, se ha de configurar el entorno de desarrollo de PX4. Para ello trabajaremos con la consola de Linux, mediante la cual introduciendo todos los comandos necesarios. Debemos descargar el código fuente de PX4 directamente de GitHub y ejecutar el archivo Ubuntu.sh para que empiece el proceso. GitHub es un repositorio online creado para alojar el código de aplicaciones libres que utiliza el sistema de control de versiones Git. Los desarrolladores suben el código y herramientas de sus aplicaciones y además permite la colaboración con ellos.

```
Git clone https://github.com /PX4/PX4-Autopilot.git --recursive
```

I. Memoria

```
Bash ./PX4-Autopilot/Tools/setup/ubuntu.sh
```

El entorno de simulación Gazebo viene preconfigurado con la instalación del entorno de desarrollo y lo que necesitamos es realizar una simulación para que se termine de instalar por completo. Para ello se entra en el directorio donde se ha guardado lo descargado previamente y se manda la orden *make*, tal que así.

```
cd/path/PX4-Autopilot
```

```
make px4_sitl gazebo
```

Según el vehículo que vayamos a simular este comando cambiará, siendo la primera parte igual, pero modificando la última, a continuación, se muestra un ejemplo de los muchos vehículos que podemos simular.

Vehículos	Comandos
Quadrotor	make px4_sitl gazebo
Standard Plane	make px4_sitl gazebo_plane
Standard Plane (with catapult launch)	make px4_sitl gazebo_plane_catapult
Standard VTOL	make px4_sitl gazebo_standard_vtol
Tailsitter VTOL	make px4_sitl gazebo_tailsitter
Ackerman UGV (Rover)	make px4_sitl gazebo_rover
Boat (USV: Unmanned Surface Vehicle)	make px4_sitl gazebo_boat
Cloudship (Airship)	make px4_sitl gazebo_cloudship

Tabla 5.1. Tabla de comandos SITL según el vehículo a simular. [23]

Cada uno de estos comandos tiene asociada toda la información necesaria de los vehículos, desde su geometría, reparto de pesos, hasta sus coeficientes aerodinámicos. Para realizar simulaciones con vehículos VTOL similares al BabyShark, solo disponemos del Standard VTOL, que en realidad poco se parece éste, ya que se trata de un ala volante con 4 motores VTOL y uno de cola, sin estabilizador horizontal ni timón de cola, como se muestra en la figura [5.2]. Es por eso que deberemos modificar los archivos existentes

referidos a esta simulación para ajustarlos a nuestra aeronave, lo que veremos en los próximos apartados.

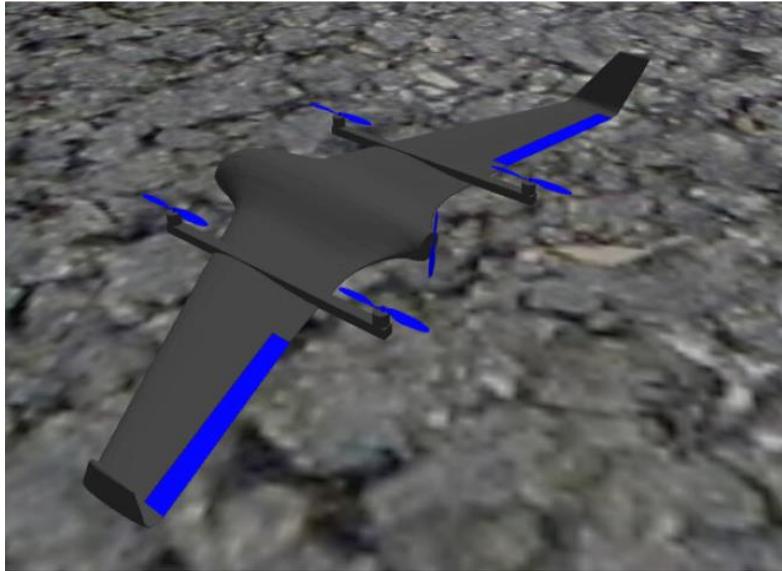


Figura 5.2. Modelo del vehículo original de gazebo como standard VTOL.

5.1.2. Configuración de la aeronave en PX4

Para llevar a cabo la simulación hemos de introducir y modificar todos los datos relacionados con las simulaciones Standard VTOL y sustituirlos por la información de nuestra aeronave. Dicha información se almacena en la carpeta `Tools/sitl_gazebo` del repositorio descargado de GitHub previamente. Principalmente existe el archivo de extensión `.sdf` (*Simulation Description File*) donde se definen las partes del vehículo, sus movimientos y su relación con el entorno. A continuación, se detalla la composición de estos archivos [8].

- **Links:** Contiene las propiedades físicas del cuerpo del modelo, siendo el modelo una serie de links que contienen cada parte del cuerpo. Se deben dividir en diferentes links las partes que tengan movimientos respecto a otras, como hélices o superficies de control. En este caso, para el dron, se crearán links para las hélices, para cada alerón, ambas superficies inclinadas de la cola y cuerpo del vehículo. Cada Link tiene diferentes propiedades modificables:
 - **Collision:** Encapsula el elemento utilizando una geometría, normalmente formas simples ya que agilizan las simulaciones gastando menos recursos, tipo cubos o cilindros, con la intención de comprobar si colisiona con cualquier otro elemento.
 - **Visual:** Se utiliza para visualizar el link, se pueden tener más de un elemento visual, o ninguno, por cada link.
 - **Inertial:** Este elemento contiene la información dinámica, tanto la masa como las inercias.
 - **Sensor:** Este elemento se encarga de recopilar los datos del entorno para emplearlos en los plugins.

I. Memoria

- Light: Describe una fuente de luz para iluminar y producir ciertas sombras creando mayor realismo.
- Joints: Se encargan de unir los links entre sí, creando relaciones padre – hijo junto con otros parámetros como ejes de rotación o límites.
- Plugins: Se trata de librerías externas que se adjuntan al modelo para controlarlo. Principalmente se usan en estas simulaciones los de `liftdrag_plugin` y `gazebo_motor_model`, actuando sobre los elementos y generando las fuerzas aerodinámicas, de empuje y momentos que simulan la dinámica en vuelo de la aeronave.

En primer lugar, para realizar la configuración del archivo `.sdf`, debemos elaborar un modelo 3D, que se realizó en Fusion 360 por [1], a partir de los datos la aeronave que proporciona el fabricante. Se ha dividido el modelo 3D en varias partes guardadas como archivos STL que corresponderán a los links, estos son: dos alerones, dos superficies de control de cola, las cuatro hélices de los motores verticales, la hélice del rotor de cola para vuelo en FW y el cuerpo de la aeronave. Se dividen de esta forma porque al simulador solo necesita diferenciar los que van a tener dinámicas o movimientos diferentes. Todos los archivos STL se han añadido a la carpeta correspondiente al `standard_vtol` para tenerlos localizados y referenciarlos en el archivo `.sdf`. Para las simulaciones no se ha modelado la cámara gimbal ya que añade cierta complejidad, pero si se ha tenido en cuenta para el centrado de la aeronave, que se midió con la bancada de equilibrado cumpliendo con el margen de la posición indicada por el fabricante.

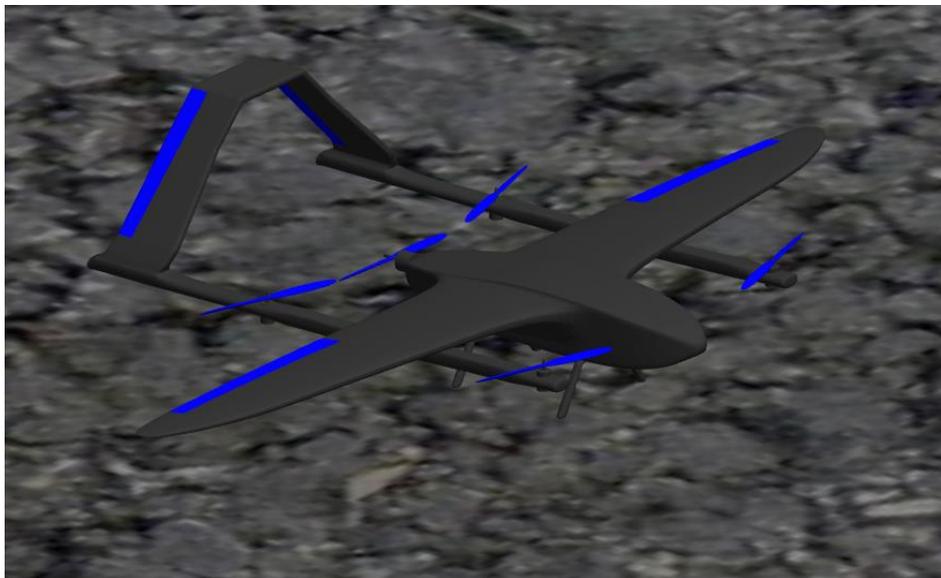


Figura 5.3. Modelo 3D de la aeronave BabyShark en Gazebo.

Modelo SDF

A continuación, se comentarán ciertos aspectos importantes q a la hora de configurar este archivo, que sigue la estructura antes descrita de links, joints y plugins.

Se definen los links fijos, correspondientes al cuerpo de la aeronave, que será el *parent* al cual los demás links de las otras partes de la aeronave harán referencia. Se introduce

5. Simulaciones y preparación para el vuelo

un comando llamado <pose> que hace referencia al centro de masas, puesto que el modelo se ha realizado alrededor del centro de masas esta línea corresponde con seis ceros, uno por cada grado de libertad, tres de posición y 3 de rotación. Si quisiéramos modificar el centro de gravedad solo deberíamos añadir aquí la variación respecto al inicial. Se introducen las inercias y la masa total del cuerpo, 8 kg. A la hora de introducir los demás elementos se hace de la misma forma que el cuerpo a excepción de los rotores, que si se han de situar con <pose> en la posición oportuna ya que se han creado con centro en el eje de rotación, por lo que es fácil situarlos de manera exacta.

Pasamos a definir los elementos de los links, también descritos antes. Las colisiones se han simplificado de forma que se considera un paralelepípedo que envuelve el dron. Esto hace que los impactos se vean afectados ya que la dinámica de colisión de la caja es muy diferente al dron, pero como en realidad solo lo necesitamos para que sepa cuando colisiona con el suelo al aterrizar, nos es suficiente con dicha simplificación. El posicionamiento se realiza indicando el centro con la línea <pose> y las dimensiones con el comando <geometry><box><size> siendo las dimensiones XYZ en metros.

Los elementos visuales se añaden mediante la introducción de la ubicación de los elementos STL provenientes del programa de modelado. Puesto que el modelo se ha centrado previamente en el programa en el centro de gravedad no es necesario aplicar moverlo con <pose>, ni el cuerpo ni ninguna otra parte del modelo, la posición respecto al origen de coordenadas ya la obtiene del archivo STL. Se cambiaron de color a azul los elementos que tienen movimientos como las superficies de control para poder ver mejor como se mueven durante las simulaciones y sus efectos.

Una vez creados todos los links, pasamos a los joints. Principalmente vamos a necesitar ejes de rotación a partir de los cuales se moverán tanto hélices como superficies de control. Se introducen un <child>, que será el elemento que se mueva respecto al <parent> que es el cuerpo de la aeronave. Para las hélices creamos el eje de rotación perpendicular al motor y en el centro de las hélices. Mientras que para las superficies de control hemos de crear el eje mediante una referencia de posición.

Para finalizar, se deben introducir los plugins, como ya se había comentado antes, el principal es el LiftDragPlugin. Este realiza el cálculo de las fuerzas y momentos aerodinámicos, aunque de manera simplificada ya que no tiene en cuenta la interacción entre superficies, como el efecto del ala en la cola, o el efecto de los rotores. Tampoco aplica los efectos de aeroelasticidad de los materiales que conforman la aeronave, limitando el cálculo aerodinámico, pero proporcionando un cálculo suficientemente aproximado como para poder estudiar el comportamiento dinámico de la aeronave.

Para realizar el cálculo, esta librería se basa en ciertos coeficientes aerodinámicos que debemos introducir. Podemos hacerlo mediante procedimientos experimentales o empleando algún programa que los calcule simulando la geometría. Se utilizó el programa XFLR5 en [1], que, aunque no nos va a proporcionar unos coeficientes de precisión excepcional ya que no se emplean las ecuaciones completas, nos proporciona unos resultados suficientemente aproximados para realizar las simulaciones. De este

I. Memoria

modo consideraremos los perfiles son iguales en todas las superficies de control y por ende sus coeficientes adimensionales también.

También debemos introducir el plugin encargado de calcular y simular las fuerzas propulsivas de los motores que se aplica a cada uno de los rotores independientemente de si son los verticales o el horizontal. Las direcciones de empuje las obtiene de los ejes de rotación en los links y joints.

Incluimos el plugin que nos permite realizar la comunicación entre PX4, el simulador y la estación de control a través de los puertos TCP y UDP especificados. Este plugin es el `mavlink_interface`. En él introducimos los canales que se relacionan con los mezcladores, que más adelante se especificarán, que permiten al autopiloto mover las superficies de control y controlar la velocidad de los motores. Cada canal se corresponde con el orden que tiene en el archivo de mezclador.

Mezcladores

Este tipo de archivos contiene la información acerca de los canales de salida que corresponden a cada servo o ESC, es por eso que son indispensables tanto en simulaciones como en vuelo, ya que limitan y escalan las salidas de los controladores para proporcionar a los actuadores los valores correctos. El orden en que se escriben es fundamental, ya que corresponden con las salidas de los actuadores y con los índices introducidos en el archivo SDF. Los archivos de los mezcladores se encuentran en la carpeta ROMFS del repositorio PX4, en ella están los mezcladores para el piloto automático para todas las aeronaves disponibles. Cuando realizamos la primera simulación, estos se copian a una carpeta llamada *build*, para tener un mejor acceso durante el vuelo.

Dado que nuestro avión no es el standard VTOL y no tiene dos elevadores y un rudder, sino que tiene dos superficies inclinadas, debemos mezclar la salida de las acciones de estabilizador horizontal y rudder para que concuerde con lo que realmente ocurre cuando deflectamos nuestras superficies inclinadas. En la figura [5.4] vemos el archivo de extensión `.main.mix` modificado que corresponde con el BabyShark.

```

Mixer for standard vtol plane (SITL) with motor x
configuration (Baby Shark)
=====

This file defines the main mixer for an A-tail standard vtol
plane (SITL Gazebo) with motors in X configuration. The plane
has two ailerons and two 45° tail surfaces.
Mixer for right A-tail right is defined in file
'standard_vtol_sitl.aux.mix'.
|
R: 4x

# mixer for the pusher/puller throttle
M: 1
O:      10000  10000      0 -10000  10000
S: 1 3      0  20000 -10000 -10000  10000

# mixer for the left aileron
M: 1
O:      10000  10000  0 -10000  10000
S: 1 0 -10000 -10000  0 -10000  10000

# mixer for the right aileron
M: 1
O:      10000  10000  0 -10000  10000
S: 1 0  10000  10000  0 -10000  10000

# A-tail mixer left
M: 2
S: 1 2  7000  7000      0 -10000  10000
S: 1 1  8000  8000      0 -10000  10000

```

Figura 5.4. Archivo mezclador principal empleado para SITL.

Debido a la forma en que se leen este tipo de archivos solo es posible introducir 4 mezcladores además de los 4 motores VTOL, por mucho que se escriban más, estos son ignorados. Por eso debemos crear otro archivo auxiliar, en este caso `.aux.mix` donde pondremos el mezclador que falta, que es el de la superficie inclinada derecha de la cola.

Pero no es suficiente con crearlo y situarlo en la misma carpeta que el principal, debemos decirle al autopiloto que existe el archivo auxiliar y que debe leerlo. Esto se hace desde el archivo de inicialización del *airframe* donde escribe los parámetros iniciales de ciertas variables dependiendo del *airframe*. El nombre del archivo es `1040_standard_vtol`, los parámetros que se cargan son los comunes a las aeronaves VTOL y posteriormente parámetros respecto a la ley de navegación, valores de controladores y ciertos parámetros específicos para el tipo concreto de VTOL. Al final del archivo contiene una línea donde se indica el tipo de VTOL, `set MAV_TYPE 22`, el número corresponde con el standard VTOL. Y por último la línea donde se indica la ubicación del mezclador, `set MIXER_FILE etc/mixers-sitl/standard_vtol_sitl.main.mix` y al final debemos introducir `set MIXER custom` para que sepa que hay un archivo auxiliar y que debe leerlo. Pero no acaba aquí, debemos darle la ubicación exacta del mezclador auxiliar, y debemos hacerlo en otro

I. Memoria

archivo, el 1040_standard_vtol.post. Consta de una sola línea donde mediante el comando `mixer append /dev/pwm_output0` introducimos la ubicación del archivo mezclador: `etc/mixers-sitl/standard_vtol_sitl.aux.mix`.

Como última modificación para incluir el mezclador auxiliar en el autopiloto, debemos incluirlo también en el archivo `CMakeLists` que está en la carpeta de los mezcladores. A continuación, se muestran dos figuras de los archivos `CMakeLists` y el mezclador auxiliar introducido.

```
px4_add_romfs_files(  
    autogyro_sitl.main.mix  
    boat_sitl.main.mix  
    delta_wing_sitl.main.mix  
    package_drop.aux.mix  
    plane_sitl.main.mix  
    quad_x_vtol.main.mix  
    rover_ackermann_sitl.main.mix  
    rover_diff_sitl.main.mix  
    standard_vtol_sitl.main.mix  
    standard_vtol_sitl.aux.mix  
    tiltrotor_sitl.main.mix  
    uuv_x_sitl.main.mix  
    vectored6dof_sitl.main.mix  
    tiltrotor_sitl_direct.main.mix  
)
```

Figura 5.5. Archivo `CMakeList` de la carpeta de mezcladores

```
Mixer for standard vtol plane (SITL) with motor x  
configuration (Baby Shark)  
=====
```

This file defines the auxiliary mixer for an A-tail standard vtol plane (SITL Gazebo) with motors in X configuration. The plane has two ailerons and two 45° tail surfaces. The other mixers are defined in file `'standard_vtol_sitl.main.mix'`.

```
# A-tail mixer right  
M: 2  
S: 1 2 -7000 -7000 0 -10000 10000  
S: 1 1 8000 8000 0 -10000 10000
```

Figura 5.6. Archivo mezclador auxiliar empleado para SITL.

En los archivos mezcladores solamente se leen las líneas que empiezan con R, H, M o S, por lo que todas las demás son comentarios que podemos dejar a modo de información o instrucciones. Estas letras corresponden a multirrotor (R), helicóptero (H), suma (M) y nulo (S).

Los mezcladores R, se utilizan para los motores de tipo multicóptero, ya sean 4, 6 u 8, se controla la velocidad de los motores combinando los controles de guiñada, alabeo, cabeceo y empuje. La letra indica la disposición de los motores respecto al cuerpo de la aeronave, en nuestro caso X. Las entradas están escaladas y los actuadores reciben señales entre -1 y 1, a excepción de los motores que van de 0 a 1, mientras que los valores se almacenan multiplicados por 10000.

Los mezcladores tipo suma se emplean para controlar el vuelo en ala fija. Actúan sobre un único control y lo pasan al actuador, es decir, el control de alabeo pasa a los actuadores de los alerones, por ejemplo. Después de la M se indica cuantos canales se van a sumar, que son en uno excepto para las superficies inclinadas, que combinan cabeceo y guiñada, cosa que se detallará más adelante. La fila O muestra la escala para valores positivos, negativos, un desfase opcional y los límites inferior y superior, en ese orden. Los factores se multiplican por 10000, como y se había dicho, el desfase se lleva a 0 y los límites son entre -1 y 1, que corresponden con -10000 y 10000.

En la tercera fila, S:, se indica el tipo de actuador, el control específico que se llevará a cabo. El primer número corresponde al grupo de control, los actuadores VTOL son el grupo 1, mientras que el segundo número indica el control exactamente el movimiento a controlar, 0 para el alabeo, 1 para cabeceo, 2 para guiñada y 3 para empuje. Los siguientes números corresponden con las mismas escalas que las filas O, escala valores positivos, negativos, desfase, límite inferior y superior.

Control Group #1 (Flight Control VTOL/Alternate)

- 0: roll ALT (-1..1)
- 1: pitch ALT (-1..1)
- 2: yaw ALT (-1..1)
- 3: throttle ALT (0..1 normal range, -1..1 for variable pitch / thrust reversers)
- 4: reserved / aux0
- 5: reserved / aux1
- 6: reserved / aux2
- 7: reserved / aux3

Figura 5.7. Grupo de control de mezcladores para VTOL.[24]

En cuanto al alabeo, los factores de escala introducidos son los mismos, siempre teniendo en cuenta que los signos de los mezcladores son cruciales, ya que podrían invertir el sentido de giro en los actuadores. Sin embargo, para el motor de cola los valores de las entradas se escalan a 0 y a 20000 y se añade un desfase de -10000, con esto los valores quedan entre -10000 y 10000.

La superficie inclinada de cola tiene un tratamiento especial, ya que cuando se actúa sobre estas superficies se tiene un efecto combinado de cabeceo y guiñada, lo que corresponde a dos controles diferentes del grupo de control. Para realizar correctamente la mezcla, en este caso sí que introducimos el número 2 en la línea M:,

I. Memoria

sumaremos el control de cabeceo (S: 1 1) y guiñada (S: 1 2). En este caso se escalan las entradas a los valores de 7000 y 8000, lo que corresponde en un 70% y 80% a la hora de la suma. Los signos del escalado de guiñada han de ser diferentes para cada superficie, ya que, dada la disposición de las superficies, en cada lado, la misma actuación corresponde a guiñadas contrarias.

5.1.3. Ejecución de la simulación y resultados

Una vez tenemos todos los archivos necesarios modificados e introducido los nuevos para realizar las simulaciones de nuestra aeronave, pasamos a realizarla.

En primer lugar, debemos abrir el programa QGroundControl ya que la conexión entre este y gazebo se realiza cuando compilamos el código, por lo que, si la estación de control no está funcionando, la conexión no se realizará. A continuación, se abre la terminal de Linux y se escribe el código mostrado en la instrucción (1). De este modo accederemos al directorio de PX4 y realizaremos la simulación del vehículo que hemos estado configurando.

```
cd /path/to/PX4-Autopilot (1)
```

```
make px4_itl gazebo_standard_vtol (2)
```

Además de la configuración de los puertos en el plugin de mavlink dentro del archivo SDF, también debemos activar la autoconexión mediante los puertos UDP dentro del programa QGC para que se conecte en cuanto detecte que los datos se están enviando. Esto se hace en *Application Settings > General* simplemente marcando la casilla.

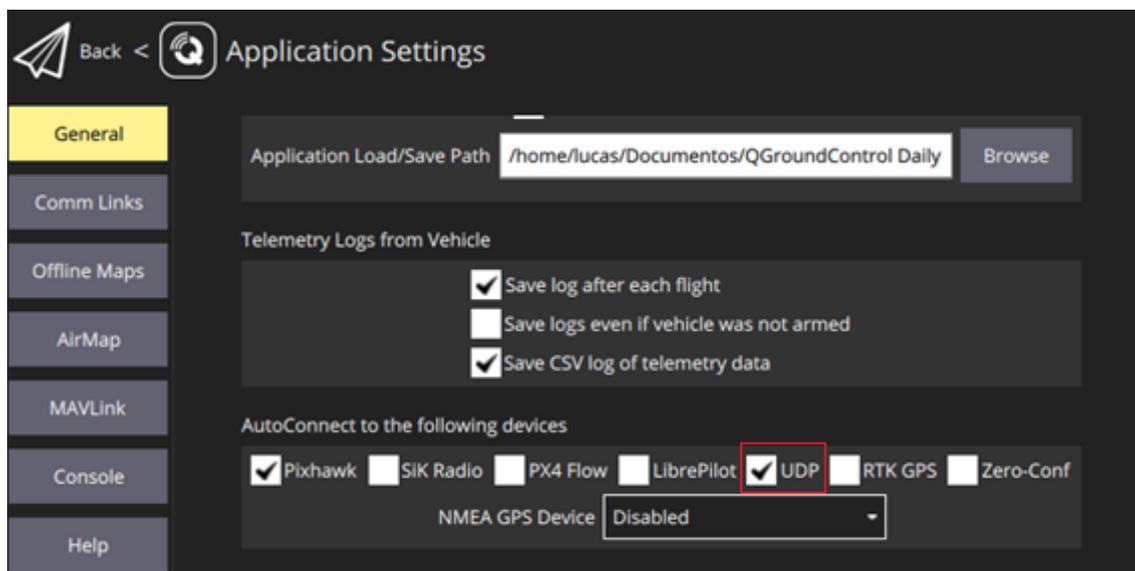


Figura 5.8. Ventana de configuración de QGroundControl.

Durante las primeras simulaciones se creó una misión sencilla y autónoma que consistía en un despegue VTOL, transición hacia vuelo en ala fija, un recorrido de unos pocos *waypoints* donde se realizarán virajes a izquierdas y derechas y posteriormente un

5. Simulaciones y preparación para el vuelo

aterrizaje VTOL, con la previa transición hacia modo multirotor. También contaba con un tramo de descenso, de 30m a 15m entre los *waypoints* 6 y 7, previo al último tramo de aterrizaje. Los planes de vuelo se pueden guardar en el ordenador para luego cargar el plan que vayamos a realizar, facilitando la labor.

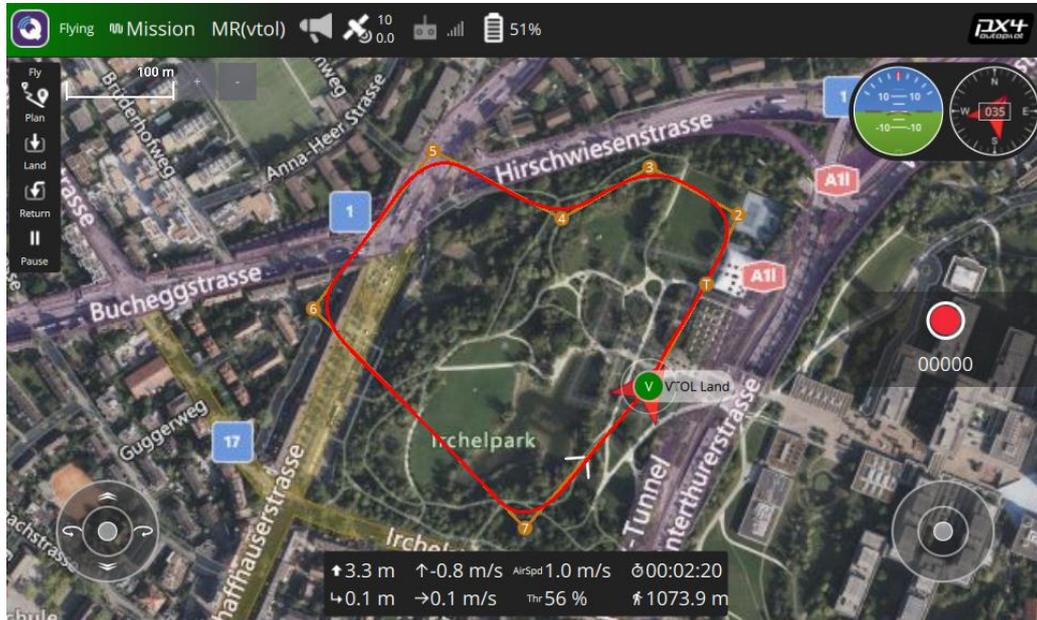


Figura 5.9. QGroundControl durante la simulación de un vuelo.

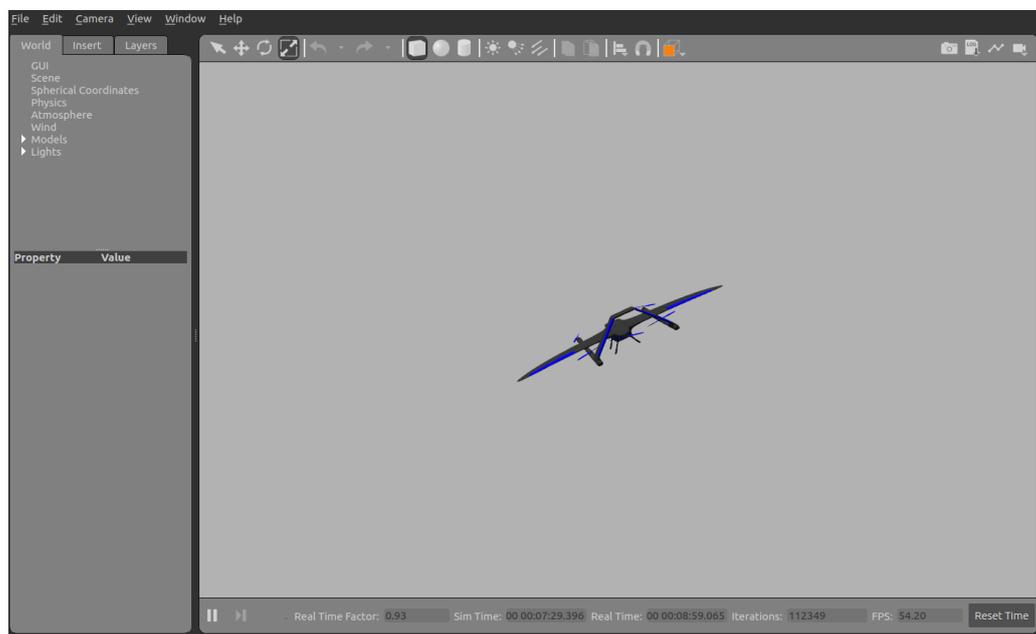


Figura 5.10. Vuelo del BabyShark en Gazebo durante una simulación.

El comienzo de la misión se realiza desde QGroundControl pulsando el botón de *start mission* y deslizando el *slider* de confirmación que aparece en la parte inferior. Aunque la misión se realice de forma autónoma, podemos pararla cambiando del modo *mission* al modo *hold*, *position* o cualquier otro modo que necesitemos y luego volver a reanudar la misión por donde la habíamos dejado.

I. Memoria

También podemos realizar el vuelo paso a paso de forma manual. En este caso con el botón de *takeoff* indicando la altura que deseamos en un *slider* que aparece en vertical a la derecha. En la barra de herramientas de arriba se indica el modo de vuelo que está en curso y si estamos volando en ala fija o multirrotor. Cuando pulsamos en el modo de vuelo podemos cambiarlo al que deseamos, mientras que si pulsamos en el tipo de vuelo podemos transicionar hacia el otro cuando queramos. Por último, podemos aterrizar con *Land* o mediante *Return* lo que hará que la aeronave vuelva al lugar de despegue o a un sitio concreto que podemos establecer como *home*, por supuesto, antes de realizar el aterrizaje, hará la transición a MC si estábamos volando en FW.

Durante la transición hacia FW la aeronave empieza a acelerar con el motor de cola, cuando llega a cierta velocidad se empiezan a mezclar los controles de MC y FW, hasta que llega a la velocidad de transición, a partir de la cual se apagan los motores MC y se considera que la aeronave ha transicionado, quedando únicamente los controles de actitud del modo FW, luego ya continúa acelerando hasta la velocidad de crucero. Dichas velocidades y más son configurables desde el panel de *setup* de QGroundControl. De forma inversa ocurre cuando pasamos de FW a MC, el motor de cola reduce se apaga, la aeronave aumenta su cabeceo para mantener altitud hasta que se encienden los motores MC, que actúan tanto como freno y ayudan a mantener la altitud.

Para los movimientos en modo MC, el autopiloto permite configurar si queremos activar el motor de cola en los modos *altitude*, *position* y *mission* para movernos hacia delante principalmente, y si deseamos hacerlo en ciertas condiciones como en el aterrizaje. Esto puede suponer una ventaja en ciertos momentos, ya que la aeronave tiene una superficie alar elevada, debemos tener cuidado con la interacción de esta durante el vuelo en MC, por ejemplo, si cabecea ligeramente para avanzar, induciendo un ángulo de ataque negativo en el ala, podría hacer que la aeronave perdiera altitud y complicar el control de actitud y altitud del modo MC. Es por esto que se ha configurado la aeronave para que, si pueda utilizar el motor de cola cuando el autopiloto considere oportuno, principalmente para moverse hacia delante, durante los modos *altitude*, *position* y *misión*, excepto durante el aterrizaje.

La estación de control en tierra permite activar el joystick virtual que se sitúa dos de ellos, uno en cada esquina inferior de la pantalla principal de la estación de control, como se ve en la figura [5.9] mostrada previamente. Sin embargo, actuar sobre ellos con el ratón es difícil, por lo que se puede conectar unos joysticks externos como los de un mando de videoconsola. En este caso, como simple prueba del modo manual se conectó un controlador de Xbox One, con lo que se verificó el buen funcionamiento de dicho modo de vuelo, tanto en MC como FW.

5. Simulaciones y preparación para el vuelo

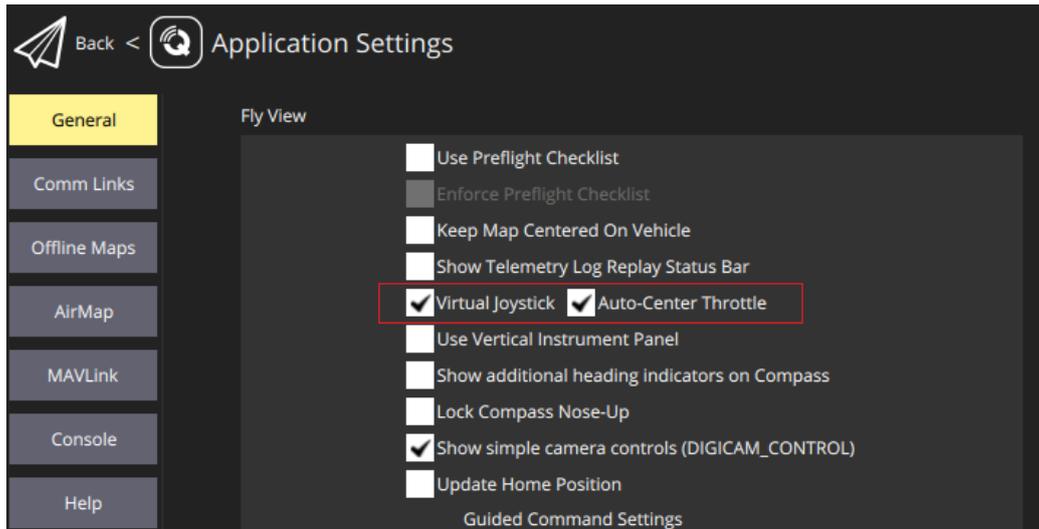


Figura 5.11. Ventana de configuración de QGC para activar el joystick virtual.

El programa QGroundControl dispone de una herramienta muy potente, el registro de los datos de vuelo, lo que se conoce como *logs* de vuelo. Estos se almacenan en la carpeta `build/px4_sitl_default`, donde se clasifican según la fecha de creación. Podemos también descargarlos directamente del programa QGC mediante el menú *Analyze tools > Log Download* donde se ordenan por fecha de simulación. Se guarda un log de cada vuelo que realizamos.

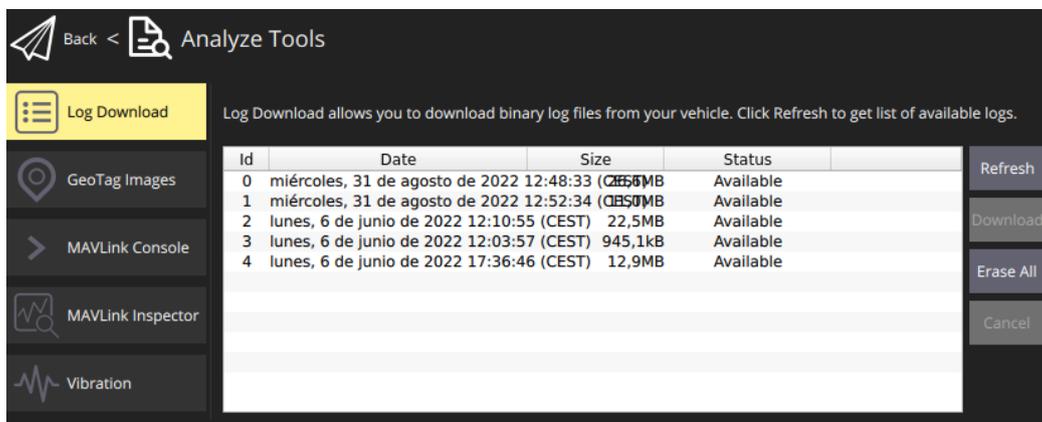


Figura 5.12. Herramienta de QGC para gestión y descarga de logs de vuelo.

Posteriormente podemos ver estos datos mediante la herramienta online *Flight Review*. Aquí podemos cargar el log y visualizar el recorrido realizado, velocidades, datos de empuje, los ángulos de cabeceo, alabeo y guiñada a lo largo de todo el vuelo mediante un gráfico que nos muestra los valores estimados y los calculados. Se trata de una herramienta muy completa donde los gráficos nos permiten conocer y mejorar el ajuste de los controladores PID y mejorar así el control de la aeronave.

Sin embargo, en la última versión del autopiloto este proceso de ajuste de los controladores PID, se puede realizar de dos formas, una manual y otra automática, ambas desde QGC, con una interfaz gráfica disponible en `vehicle setup > PID Tuning Setup`. En la manual los gráficos nos muestran la variable objetivo y la respuesta de la

I. Memoria

aeronave a medida que vamos modificando los valores de las constantes del controlador y lo vamos ajustando. Mientras que, para el ajuste automático, debemos estar en el modo de vuelo *hold* para vuelo en MC y en el modo *position* para FW, y a partir de ahí dar comienzo al proceso.

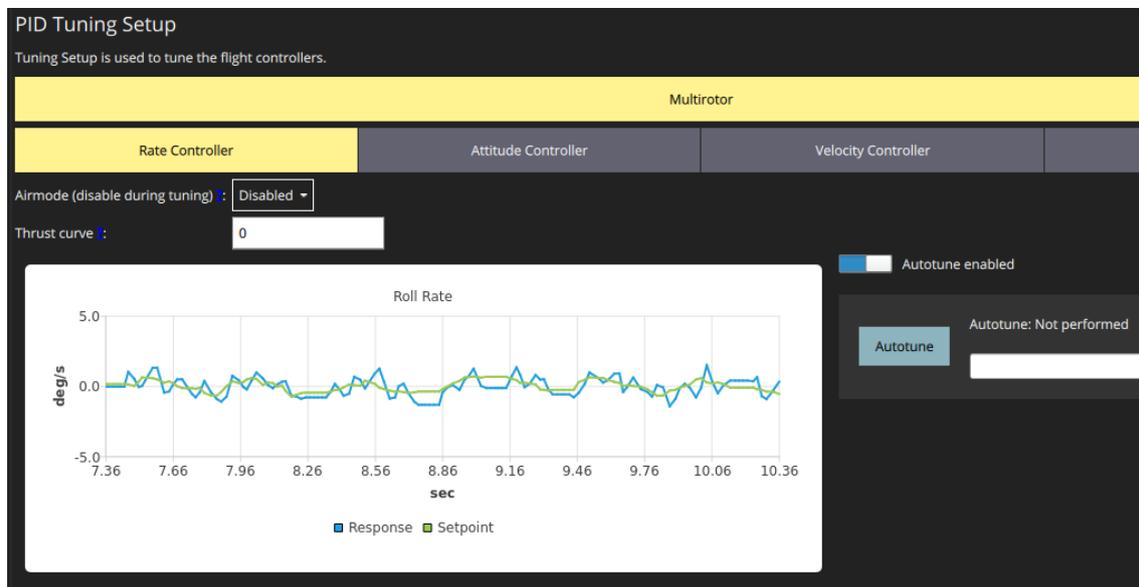


Figura 5.13. Herramienta de configuración de los controladores PID.

Lo que debemos tener en cuenta es que se ha de partir de una configuración de PIDs que nos permita volar, ya que si nuestra configuración inicial no es nada buena el ajuste automático de los PIDs no se hará correctamente. Dado que la configuración inicial del *airframe* ya contaba con unos controles PID aceptables, se ha podido realizar el ajuste de los PIDs para nuestra aeronave mediante el modo automático. Podemos ver como el hecho de realizar un *autotune* mejora bastante el control de la aeronave. El efecto que ha tenido sobre el control de cabeceo se muestra en las siguientes figuras, obtenidas con la herramienta *Flight review*.



Figura 5.14. Angulo de cabeceo durante vuelo en ala fija previo al autotune.



Figura 5.15. Angulo de cabeceo durante vuelo en ala fija posterior al autotune.

Puesto que ya partíamos de unos controles relativamente buenos, el alabeo y la guiñada ya tenían un buen ajuste y ha sido el cabeceo donde mayores ajustes se han realizado. Se observa claramente como durante uno de los giros ha de aumentar el cabeceo para no perder altitud, y como luego viene el tramo de cambio de altitud y lo disminuye para bajar.

Ajustados los controladores para FW y MC, queda optimizar la transición en ambos sentidos. La transición de MC a FW es ciertamente la más suave y fácil de ejecutar. Lo que debemos ajustar es el porcentaje de *throttle* durante ella, el 100%, la velocidad a partir de la cual empieza a mezclar controladores de ambos tipos de vuelo, ajustada en 8 m/s, la velocidad a la cual se considera que la transición ha concluido, 12 m/s, y la velocidad de crucero, hasta la cual acelerará al final del proceso de transición propiamente dicho, 21 m/s, que es el valor dado por el fabricante. Estos serían los parámetros más significativos para la transición.

En cuanto a la transición de FW a MC, debemos indicar el valor de *throttle* para el motor de cola cuando iniciamos la transición, en este caso 0. Se podría llegar hasta -1, en caso de que nuestro motor pudiera girar al revés, realizando una inversión de empuje y frenando. La aeronave reduce la velocidad hasta llegar a la de mezcla de controles, se encienden los rotores verticales y la aeronave cabecea ligeramente para frenar, tanto por aumento de resistencia como por el efecto de los motores. Se establece una duración de 12 segundos para la transición.

5.2. Hardware in the loop

Para esta simulación será directamente la placa controladora Cube Orange la que realice todo el trabajo del autopiloto, no el ordenador, como en las simulaciones SITL. Se simulan las lecturas de los sensores, mientras que las salidas de todos los actuadores de la placa y motores permanecen apagados por motivos de seguridad.

En este caso el simulador se comunica con la controladora PX4 mediante USB, mientras que las comunicaciones entre el simulador y la estación de control se realiza con

I. Memoria

MAVLink internamente, también podemos conectar un joystick externo como en las SITL.

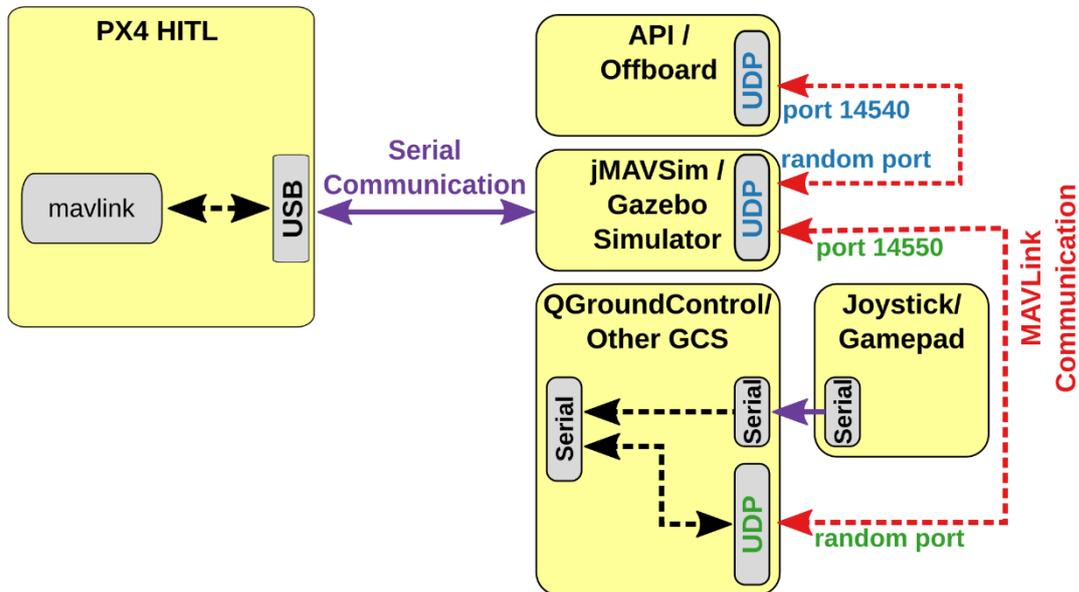


Figura 5.16. Comunicaciones HITL entre simulador, estación de control y PX4. [22]

Para realizar las simulaciones HITL debemos realizar ciertos cambios, tanto desde la estación de control como en los archivos SDF. En primer lugar, debemos seleccionar el *airframe* adecuado, ya que no todos soportan las simulaciones HITL, el que debemos utilizar es el HIL Standard VTOL QuadPlane, que es el mismo que utilizábamos desde el código de PX4 para las simulaciones SITL. También debemos activar el modo HITL del autopiloto, en el apartado de seguridad, como se muestra en la siguiente figura.

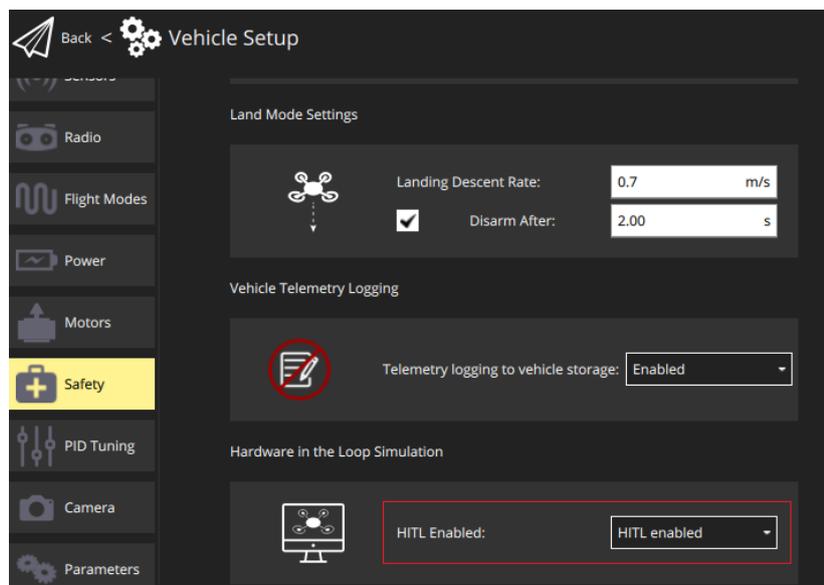


Figura 5.17. Ventana de configuración en QGC para activar la simulación HITL.

Al igual que en las SITL, debemos modificar ciertos parámetros de los archivos SDF. En este caso el archivo SDF es el mismo, a diferencia que se encuentra en otra carpeta que se utiliza exclusivamente para simulaciones HITL. Dentro del archivo debemos modificar

5. Simulaciones y preparación para el vuelo

los parámetros del plugin de MAVLink. Concretamente debemos habilitar la conexión *serial* introduciendo un 1, que es la conexión USB entre el simulador y la controladora, comprobar que el dispositivo es el adecuado, en este caso es el puerto *ttyACM0* por defecto en Linux, y por último habilitar la simulación HIL, escribiendo 1. Estas líneas a cambiar son las marcadas en rojo en la figura [5.18].

```
<plugin name='mavlink_interface' filename='libgazebo_mavlink_interface.so'>
  <robotNamespace></robotNamespace>
  <imuSubTopic>/imu</imuSubTopic>
  <magSubTopic>/mag</magSubTopic>
  <baroSubTopic>/baro</baroSubTopic>
  <mavlink_addr>INADDR_ANY</mavlink_addr>
  <mavlink_tcp_port>4560</mavlink_tcp_port>
  <mavlink_udp_port>14560</mavlink_udp_port>
  <serialEnabled>0</serialEnabled>
  <serialDevice>/dev/ttyACM0</serialDevice>
  <baudRate>921600</baudRate>
  <qgc_addr>INADDR_ANY</qgc_addr>
  <qgc_udp_port>14550</qgc_udp_port>
  <sdk_addr>INADDR_ANY</sdk_addr>
  <sdk_udp_port>14540</sdk_udp_port>
  <hil_mode>0</hil_mode>
  <hil_state_level>false</hil_state_level>
  <enable_lockstep>true</enable_lockstep>
  <use_tcp>true</use_tcp>
```

Figura 5.18. Archivo SDF para simulaciones HITL.

En cuanto al procedimiento que debemos seguir para realizar las simulaciones, es ligeramente diferente al realizado en las SITL. Debemos tener conectada la controladora al ordenador mediante USB, ya que cuando iniciemos la simulación, el simulador va a intentar conectarse al puerto *serial* establecido en el plugin y si no está disponible la simulación no funcionará. En cuanto a QGC, no importa si esta abierto o no, si no lo está, cuando se abra se conectará automáticamente con el autopiloto. Las líneas de código que debemos introducir en la terminal, y siempre desde la carpeta donde se ubican todos los archivos del autopiloto, son las mostradas a continuación.

```
DONT_RUN=1 make px4_sitl_default gazebo (3)
```

```
source Tools/setup_gazebo.bash $(pwd) $(pwd)/build/px4_sitl
_default (4)
```

```
gazebo Tools/sitl_gazebo/worlds/hitl_standard_vtol.world (5)
```

La primera línea inicializará gazebo y sus plugins pero sin abrirlo, la segunda configurará las variables del entorno y la última inicia Gazebo en el modo HITL desde el archivo *hitl_standard_vtol.world*. Desde este archivo se establecen las características del entorno de vuelo, lo que nos permite añadir un plugin de viento para simular viento y rachas en diferentes direcciones.

Una vez tenemos el entorno de simulación listo ya podemos pasar a realizar varias simulaciones. Volvemos a ajustar los PIDs de la misma forma que en SITL, con el

autotune y simulamos los planes de vuelo que ya habíamos probado con la simulación SITL, comprobando que la aeronave funciona correctamente.

5.3. Preparación para el vuelo

Hechas ya las simulaciones de vuelo de la aeronave y comprobado que el autopiloto funciona correctamente, pasamos a realizar la puesta a punto de todos los sistemas embarcados y en tierra que formarán parte de los vuelos de la aeronave.

5.3.1. Puesta a punto de todos los componentes

En primer lugar, el sistema de telemetría, una vez tenemos todos los receptores y transmisores correctamente conectados como se indica en el diagrama de conexiones del apartado 4, debemos asegurar que se realiza correctamente la conexión entre la aeronave y la estación de control en tierra.

Para ello debemos comprobar que recibimos la señal en la antena redundante SMRTS de los dos receptores, el de 868 MHz, el principal, y el de 433MHz, el secundario o redundante. Con el programa DMD Studio, instalado en el ordenador con Windows 10 que incorpora la estación de control GCSD4, podemos ver y modificar los parámetros de todos los componentes, conectándolos ya sea mediante el cable Ethernet como la antena redundante o mediante USB como los receptores de abordaje. Dentro del programa podemos realizar la configuración de la radio, configurar los modos de vuelo, realizar cambios en los botones y clavijas de la estación de control para posteriormente asignarles las funciones que queramos y más modificaciones, que podemos consultar en [11]. A partir del manual, que sirvió como guía, se llevó a cabo la configuración de los equipos. En él se detallan a que corresponden los parámetros y valores adecuados según el uso. A continuación, se muestra una captura del programa configurando el receptor RXLRS.



Figura 5.19. Configuración de radio principal en DMD Studio.[11]

5. Simulaciones y preparación para el vuelo

En cuanto a los receptores de video, la configuración se realiza sobre los mismos dispositivos. Disponen de un pequeño *display* donde se puede modificar los canales de recepción y ajustar así la imagen. Cuando el video se conecta y vemos la cámara FPV, también debemos ver todos los elementos disponibles correctamente, estos son los ángulos de actitud, horizonte artificial, velocidad, altitud, detalles de la batería, indicadores de recepción de señal de radio y demás elementos que se detallan en la imagen siguiente.



Figura 5.20. Objetos e instrumental de vuelo superpuestos al video de la cámara FPV.[11]

Para poder recibir toda esta información por medio del transmisor de video, debemos modificar los datos que queremos que salgan por el puerto de telemetría 2 de la controladora de vuelo, al que se conectan los transmisores. Esto se hace en los parámetros de MAVLink, desde QGC, cambiando el modo de MAVLink del puerto TELEM2, que modifica los mensajes que se transmiten por ese puerto, el modo que usaremos es el OSD (*On-Screen Display*).

QGC dispone de 12 canales de radio para realizar la configuración de los modos de vuelo, controles de la aeronave y demás funciones de acceso rápido, los 4 primeros corresponden a los joysticks. Para poder utilizar cada canal con los botones que deseamos de la estación de control, debemos acceder al receptor RXLRS mediante el programa DMD Studio y asignar los botones e interruptores al canal que queramos y darle un valor a cada uno. Puesto que por defecto ciertos interruptores no estaban asignados a ningún canal, se ha personalizado los interruptores y botones para facilitar el acceso a ciertas funciones durante el vuelo, aunque se podría realizar mediante la pantalla de QGC, es mucho más práctico e intuitivo realizarlo desde los botones.

En cuanto a los modos de vuelo, los asignamos a los botones de la parte inferior de la estación de control, correspondientes al canal 5. Mientras que a los interruptores de

I. Memoria

palanca le asignamos las funciones de armado, transición entre MC y FW, RTL (*Return to Launch*) y el apagado de emergencia (*Emergency Kill*).

Por último, debemos configurar el puerto de GCSD4 al que conectamos la antena inteligente para que QGC se conecte a ella y empiece a recibir los datos de la aeronave, cuando abrimos el programa. Lo que se realiza desde Application Settings > Comm Links. Seleccionamos el tipo *Serial*, el Baud rate a 115200 y activamos la auto conexión al iniciar la aplicación. Esto se muestra en la figura [5.21].

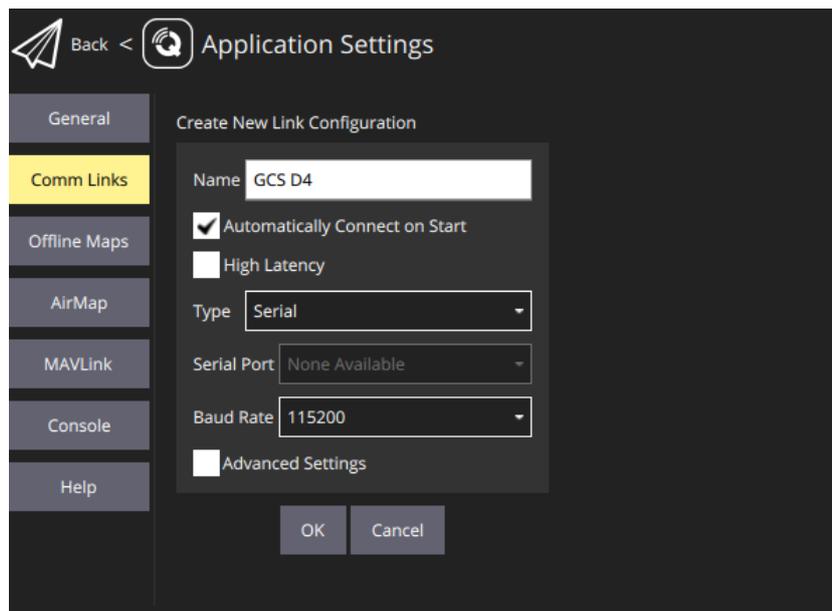


Figura 5.21. Configuración del enlace entre la estación de control y el sistema de comunicaciones.

Hechas todas estas modificaciones y comprobado que todo se conecta con normalidad, y que todos los motores y superficies de control responden adecuadamente, la aeronave esta lista para realizar el primer vuelo.

6. Conclusiones y posibles futuros trabajos

En este trabajo se ha introducido el concepto de UAS y se ha hablado de las normativas vigentes. Es un sector que ha crecido mucho durante los últimos años y muchas empresas están apostando por él, fabricando nuevas aeronaves innovadoras y sistemas para mejorar la gestión del tráfico aéreo, para poder al fin integrarlo junto con el tráfico tripulado, de la mano del U-Space en Europa.

Por otro lado, se ha trabajado con autopilotos de código libre, que nos ha demostrado el gran potencial que tienen, haciendo posible un buen control de la aeronave. También es cierto que aún carecen de ciertas funcionalidades para con el tráfico aéreo, como alerta de colisiones e información sobre otras aeronaves que estén volando cerca, cosa que seguro se irán añadiendo tanto a PX4 como a otros autopilotos de código libre, ya que son proyectos que están en continuo desarrollo. Aún así es una gran herramienta que nos ha permitido implementar el sistema de telemetría de largo alcance.

En cuanto al sistema de telemetría, dispone de unas funcionalidades muy amplias y una gran posibilidad de personalización que no se ha llegado a implementar, permitiendo también el control de varias cámaras simultáneamente. Por lo que quedaría como una próxima tarea a realizar, adaptar todos los controles y comandos dependiendo de las necesidades de la misión y de lo que el piloto prefiera.

No se ha podido llegar a realizar ningún vuelo experimental de prueba, aunque la aeronave y todos los sistemas necesarios se han dejado listos para ello y dicha primera prueba se realizará en un futuro próximo. Sin embargo, no estará en el ámbito de este proyecto, aunque si queda la puerta abierta a un próximo proyecto en el cual se realicen esos primeros vuelos y se adapten las simulaciones al vuelo real, consiguiendo un buen el equilibrado del vuelo de la aeronave. Además se podría incluir mejoras internas, en lo que a cableado, conexiones y mejor gestión del espacio interior se refiere, cosa que permitiría introducir carga de pago en su interior.

Bibliografía

- [1]. Smith Ballester, LC. (2022). Integración del modelo digital de un VTOL con el autopiloto. Universitat Politècnica de València. <http://hdl.handle.net/10251/180813>.
- [2]. Daniel Gillins y col. «EYES IN THE SKY: BRIDGE INSPECTIONS WITH UNMANNED AERIAL VEHICLES Final Report SPR 787».
- [3]. International Trade Administration. Página web sobre Unmanned Aircraft Systems. 2021. [Unmanned Aircraft Systems \(trade.gov\)](https://www.trade.gov/unmanned-aircraft-systems)
- [4]. SITEP. Página web sobre tipos de drones. [¿Qué tipo de dron escojo? Ala fija, multicóptero o helicóptero – Sitep Ingeniería GIS](https://www.sitep.es/que-tipo-de-dron-escojo/).
- [5]. SITEP. Página web sobre drones. url: <https://www.droneuropa.com/U-Space>.
- [6]. PX4 Autopilot. Controller Diagrams. 2021. url: [Controller Diagrams | PX4 User Guide](https://docs.px4.io/en/flight_controller/controller_diagrams.html)
- [7]. Dario Brescianini, Markus Hehn y Raffaello D'Andrea. «Nonlinear Quadcopter Attitude Control. Technical Report». <https://doi.org/10.3929/ethz-a-009970340>.
- [8]. Gazebo. Make a model. 2021. [Gazebo : Tutorial : Make a model \(gazebosim.org\)](https://wiki.gazebo.org/en/Tutorial_Make_a_Model.html)
- [9]. T-Motor. V505 KV260. Specifications. 2021. [V505 KV260 V Type Motors Multicopter T-MOTOR Store-Official Store for T-motor drone motor,ESC,Propeller \(tmotor.com\)](https://www.tmotor.com/v505-kv260-v-type-motors-multicopter-t-motor-store-official-store-for-t-motor-drone-motor-ESC-Propeller)
- [10]. Dualsky. ECO4120C-V2 Series Brushless Outrunners. 2021. [ECO4120C-V2 series brushless outrunners \(dualsky.com\)](https://www.dualsky.com/eco4120c-v2-series-brushless-outrunners)
- [11]. Manual de los sistemas XLRs. [Manuales XLRs indice \(dmd2.es\)](https://www.dmd2.es/manuales-xlrs/)
- [12]. Elementos de un UAS. [Basic Components of an Unmanned Aircraft System | Download Scientific Diagram \(researchgate.net\)](https://www.researchgate.net/publication/312111111_Basic_Components_of_an_Unmanned_Aircraft_System)
- [13]. PX4 VTOL frames. [Tailsitter VTOL | PX4 User Guide](https://docs.px4.io/en/flight_controller/tiltrotor_vtol.html)
- [14]. PX4 VTOL frames. [Tiltrotor VTOL | PX4 User Guide](https://docs.px4.io/en/flight_controller/tiltrotor_vtol.html)
- [15]. Tiltwing Frame. [BNA Tiltrotor UAV is Flight Ready - Defence Turkey Magazine](https://www.defence-turkey.com/news/bna-tiltrotor-uav-is-flight-ready)
- [16]. QuadPlane Frame. [Affordable And Reliable VTOL Drone For Mapping And Surveillance! \(yangdaonline.com\)](http://www.yangdaonline.com/quadplane-frame-affordable-and-reliable-vtol-drone-for-mapping-and-surveillance/)
- [17]. Aerotaxi Lilium. [Lilium Jet - The First Electric VTOL \(eVTOL\) Jet - Lilium](https://www.lilium.com/en/press-releases/lilium-jet-the-first-electric-vtol-evtol-jet)
- [18]. Aeronave Babyshark VTOL 260. [Foxtech Baby Shark VTOL 260 Drone | The Australian Drone Specialists \(integraldrones.com.au\)](https://www.integraldrones.com.au/foxtech-baby-shark-vtol-260-drone/)
- [19]. Placa Controladora Cube Orange. [Cube Orange Flight Controller | PX4 User Guide](https://docs.px4.io/en/flight_controller/cube_orange.html)
- [20]. Diagrama de puertos PX4. [The Cube Orange Standard Set \(ADS-B Carrier Board\) — Flying Robot](https://docs.px4.io/en/flight_controller/cube_orange.html)

- [21]. PX4 SITL. [Simulation | PX4 User Guide](#)
- [22]. PX4 HITL. [Hardware in the Loop Simulation \\(\HITL\\) | PX4 User Guide](#)
- [23]. PX4 SITL Simulations. [Gazebo Simulation | PX4 User Guide](#)
- [24]. Grupos de Control en mezcladores. [Mixing and Actuators | PX4 User Guide](#)

Parte II

Presupuesto

II. Presupuesto

7. Presupuesto del proyecto

En este apartado se tratará de cuantificar los recursos que se han empleado para la realización del proyecto, así como el coste que ha supuesto cada recurso y el coste total del proyecto.

7.1 Recursos

Personal

Código	Factor	Descripción	Cantidad
RP01	h	Alumno	1

Recursos informáticos

Código	Factor	Descripción	Cantidad
RI01	Ud	PC	1
RI02	Ud	QGroundControl	1
RI03	Ud	PX4	1
RI04	Ud	Gazebo	1
RI05	Ud	Windows 10	1

Materiales

Código	Factor	Descripción	Cantidad
MA01	Ud	Pack BabyShark	1
MA02	Ud	Fh230 TR 30x Gimbal camara	1
MA03	Ud	Batería Motor de cola	2
MA04	Ud	Batería para motores VTOL	2
MA05	Ud	Sistema de telemetría de largo alcance	1
MA06	Ud	Cámara FPV Caddx	2

El pack BabyShark consta tanto del fuselaje como de los componentes que venían con él, motores, hélices, ESCs, etc. Los costes de todos los componentes del pack BabyShark se contabilizan juntos ya que se presupuestaron en conjunto en el momento de la compra. Los detalles de los componentes se muestran a continuación.

Componentes	Marca	Modelo	Unidades
Fuselaje	Foxtech	BabyShark	1
Motor de cola	Dualsky	ECO 4120 V2	1
Hélice del motor de cola	n/a	15"x8"	1
ESC motor de cola	Hobbywing	FlyFun 120A V5 SBEC	1
4 x Motor VTOL	T-Motor	MN505-S KV260	4
4 x Hélice VTOL	MARKII	16" x 5.5"	4

II. Presupuesto

4 x ESC VTOL	T-Motor	FLAME 60A HV	4
4 x Servo superficies control	n/a	n/a	1
Autopiloto	Pixhawk	Cube Orange	1
GNSS	n/a	Here 3 GNSS	1
Power Monitor	n/a	n/a	2
Regulador Voltaje	Assan	UBEC-8A	1
Tubo de Pitot	n/a	Px4 Air Speedometer	1

En cuanto al sistema de telemetría, todos los componentes adquiridos se resumen en la siguiente tabla, todos del fabricante español DMD (Digital Micro Devices).

Componentes	Modelo	Unidades
Estación de control en tierra	GCSD4RR V2	1
Antena inteligente redundante	SMRBTS	1
Receptores radio control (868 MHz y 433MHz)	RXLRS	2
Transmisor y receptor de video de (1.2, 2.4 y 5.8 GHz)	XVID3B	3
Antena Biquad 868MHz	BQ89	1
Antena Biquad 1.2 GHz	BQ12	1
Antena moxon 433MHz	MX 433	1

Los materiales incluidos en la tabla son los que se han comprado para la realización del proyecto y los que se compraron antes, pero aún no se han amortizado. Aparte de los materiales incluidos en la tabla anterior, existen otros equipos necesarios para el funcionamiento de la aeronave, como son elementos de cableado, antenas embarcadas elementos de sujeción como tornillos arandelas y demás, que se compraron antes del inicio de este proyecto y están totalmente amortizados.

7.2 Precios unitarios

Para calcular el coste de personal se considerará el salario bruto por hora de 14€/h, para un ingeniero aeronáutico junior. Teniendo en cuenta todas las horas tanto de revisión de conceptos previos, realización de simulaciones, puesta a punto de todos los equipos y posterior redacción del proyecto se estiman unas 300 horas de trabajo.

Costes de Personal			
Código	Precio unitario	Cantidad	Importe [€]
RP01	14 €/h	300 h	4200.00
Total			4200.00

7. Presupuesto del proyecto

Para el cálculo de los equipos informáticos utilizados se tendrá en cuenta el gasto que suponen las licencias de los programas. Puesto que todos los programas utilizados y herramientas informáticas empleadas como apoyo a los principales programas son de software libre no tienen ningún coste de adquisición.

Considerando un precio de adquisición de 700€ para el PC y un periodo de amortización de 5 años, teniendo en cuenta que la duración del proyecto han sido 3 meses de los 60 totales de amortización. De las 300 horas de realización de proyecto aproximadamente durante unas 50 no se ha utilizado el ordenador, concretamente durante la puesta a punto de los sistemas, quedando un total de 250 horas.

Costes Recursos Informáticos			
Código	Precio unitario	Cantidad	Importe [€]
RI01	0.14 €/h	250 h	35.00
Total			35.00

Todos los materiales empleados en el proyecto forman parte de la aeronave y sus sistemas en tierra por lo que de ahora en adelante se van a utilizar en conjunto, acumulando así las mismas horas de uso

Los componentes del sistema de telemetría tienen los siguientes precios.

Componentes	Unidades	Importe [€]
Estación de control en tierra con antena inteligente redundante y receptores	1	5872.00
XVID3B 1.2 GHz	1	990.00
XVID3B 2.4 GHz	1	930.00
XVID3B 5.8 GHz	1	1010.00
Antena BQ89	1	63.00
Antena BQ12	1	63.00
Antena MX433	1	39.00
Total		8967.00

Costes de materiales				
Código		Precio unitario	Cantidad	Importe [€]
MA01		8898.21	1Ud	8898.21
MA02		4437.65	1Ud	4437.65
MA03		402.43	2Ud	804.86
MA04		230.00	2Ud	460.00

II. Presupuesto

MA05		8904.00	1Ud	8967.00
MA06		30.00	2Ud	60.00
			Total	23627.72

7.3 Presupuesto total

PRESUPUESTO TOTAL	
Tipo de coste	Importe [€]
Coste de Personal	4200.00
Coste de Recursos Informáticos	35.00
Coste de materiales	23627.72
TOTAL	27862.72

El coste total del proyecto llevado a cabo son **VEINTISIETE MIL OCHOCIENTOS SESEINTA Y DOS EUROS CON SETENTA Y DOS CÉNTIMOS.**