



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



ESCUELA TÉCNICA
SUPERIOR INGENIEROS
INDUSTRIALES VALENCIA

Curso Académico:

Resumen

El proyecto ofertado consiste en el desarrollo de una plataforma de ensayo que implemente un algoritmo de control de la orientación de un sistema quadrotor. Dada la relativa complejidad del proyecto propuesto, se desglosa en dos TFG separados, pero que requieren de una gran coordinación entre los alumnos. En este TFG, se va a llevar a cabo todo el desarrollo hardware, el cual irá desde la construcción de la plataforma para instalar el prototipo de un helicóptero de cuatro motores, hasta el montaje de todos los componentes electrónicos que configuran el sistema (placa Raspberry Pi, sensores de orientación IMU, drivers para los motores, etc.). El TFG se desarrollará por tanto en varias fases:

- Inicialmente se procederá a diseñar y construir una plataforma que permita el giro con tres grados de libertad del sistema Quadrotor (helicóptero con 4 motores).
- Paralelamente, se estudiará la posibilidad de incluir un grado más de libertad, la altura. Para ello se evaluará la posibilidad de construir un sistema con dos tubos telescópicos que permitan un movimiento en el eje z.
- Una vez construida la parte mecánica del sistema, se abordará el montaje de todo el sistema Quadrotor, el cual consta de diferentes dispositivos a conectar eléctrica y electrónicamente.

Palabras clave: Control, Raspberry Pi, Hardware, Quadrotor, RPAS.

A mis padres, porque me han hecho la persona que soy.

A mi hermano, por ser un referente y el mejor amigo.

A mis compañeros y amigos, con quienes he disfrutado cada momento.

A mi tutor, por darme la oportunidad de realizar este trabajo.

Este escrito consta de los siguientes documentos:

MEMORIA
PRESUPUESTO
PLANOS

MEMORIA

Índice de la memoria

1. Introducción	7
1.1. Propuesta y objetivos	7
1.2. Estructura del documento	7
2. Estado del arte	9
2.1. Generalidades de quadrotores y UAVs	9
2.2. Breve reseña histórica	9
2.2.1. Evolución de los quadrotores	9
2.2.2. Evolución de los UAV	10
2.3. Actualidad	11
2.4. Futuro	11
2.5. Normativa y licencias	12
2.6. Justificación del trabajo	12
3. Modelo matemático	15
3.1. Configuración de los ejes en el quadrotor	15
3.2. Ángulos de posición	16
3.3. Matrices de cambio de base	16
3.3.1. Cambio de base de magnitudes lineales	17
3.3.2. Cambio de base de magnitudes angulares	17
3.4. Acciones aplicadas sobre el quadrotor	18
3.4.1. Empuje	18
3.4.2. Par	19
3.5. Fuerzas y pares	19
3.5.1. Fuerza de la gravedad	20
3.5.2. Fuerza de empuje	20
3.5.3. Pares generados por los rotores	20
4. El quadrotor	21
4.1. Elementos básicos del quadrotor	21
4.1.1. Sensores	21
4.1.2. Controlador	22

4.1.3. Actuadores.....	23
4.1.4. Chasis y protección	23
4.1.5. Otros componentes	24
4.2. Solución constructiva	24
4.2.1. Chasis.....	24
4.2.2. Motores	25
4.2.3. Hélices.....	27
4.2.4. Drivers.....	29
4.2.5. Raspberry Pi 2.....	30
4.2.6. Puente de conexiones Raspberry-Arduino	32
4.2.7. Conversor de tensiones	33
4.2.8. IMU	34
5. Conexiones y comunicación	37
5.1. Conexiones eléctricas y alimentación	37
5.2. Comunicación y transferencia de información	38
5.2.1. Comunicación I2C	38
5.2.2. Señales PWM.....	40
5.2.3. Protocolo UDP	42
5.2.4. Esquema general del flujo de información.....	43
6. Base de pruebas.....	45
6.1. Concepto	45
6.2. Materiales.....	46
6.2.1. Base y mástil	46
6.2.2. Tubo de carbono.....	46
6.2.3. La rótula.....	46
7. Diseño	47
7.1. Diseño del quadrotor	47
7.2. Diseño de la base de pruebas completa	48
7.3. Modelado de componentes	49
8. Software	51
8.1. Sistema Operativo en Tiempo Real	51

8.2. Comunicación con el quadrotor (HMI).....	52
8.3. Algoritmo de control	53
8.3.1. Control mediante PID	53
8.3.2. Esquema del PID implementado	54
8.3.3. Obtención de los parámetros del PID.....	55
8.3.4. Necesidad de un SOTR.....	56
9. Ensayos.....	57
9.1. Control del Roll.....	57
9.1.1. Respuesta a cambios de referencia en Roll.....	57
9.1.2. Respuesta ante perturbaciones en Roll.....	58
9.2. Control del Pitch.....	59
9.2.1. Respuesta a cambios de referencia en Pitch.....	59
9.2.2. Respuesta ante perturbaciones en Pitch.....	60
9.3. Control del Yaw	61
9.3.1. Respuesta a cambios de referencia en Yaw	61
9.3.2. Respuesta ante perturbaciones en Yaw	62
9.4. Observaciones	62
10.Conclusiones y trabajos futuros.....	63
10.1.Conclusiones.....	63
10.2.Trabajos futuros	64
Bibliografía	65
Anexos.....	67
Anexo I	67
Anexo II	69

Índice de ilustraciones

Ilustración 1. Modelo A, desarrollado por Convertawings.	9
Ilustración 2. Predator MQ-1.	10
Ilustración 3. Amazon PrimeAir.....	11
Ilustración 4. 3 DOF Hover, cortesía de Quanser.	13
Ilustración 5. Configuración en cruz.....	15
Ilustración 6. Configuración en equis.....	15
Ilustración 7. Ángulos de posición.....	16
Ilustración 8. Empujes y pares del quadrotor.	18
Ilustración 9. 5DOF Accelerometer Gyroscope, cortesía de SparkFun.	21
Ilustración 10. Sensor de ultrasonidos HC-SR04, cortesía de Cytron Technologies.....	22
Ilustración 11. Microcontrolador, cortesía de Microchip.	22
Ilustración 12. IGEPv2, Arduino UNO, Raspberry Pi 2.	23
Ilustración 13. Brushless Outrunner Motors, cortesía de Scorpion Power System.	23
Ilustración 14. QAV250 G10 Mini FPV Quadcopter, cortesía de Lumenier.....	23
Ilustración 15. Set de chasis mk40, cortesía de MikroKopter.	24
Ilustración 16. Montaje del chasis.....	25
Ilustración 17. Motores Brushless Yellow Trainer 1500 usados en el trabajo.	26
Ilustración 18. Definición de 'paso', cortesía de www.bolly.com.au	27
Ilustración 19. Diagrama vectorial de fuerzas y pares en la sección de una hélice.	27
Ilustración 20. Conjunto de hélices y porta hélices usados en el trabajo.....	28
Ilustración 21. Montaje de los motores y hélices sobre el chasis.....	28
Ilustración 22. Drivers Turnigy Basic 25A usados en el trabajo.	29
Ilustración 23. Montaje del chasis con motores y drivers instalados.	30
Ilustración 24. Raspberry Pi 2 usada en el trabajo.....	31
Ilustración 25. Montaje de la Raspberry Pi sobre el chasis.....	31
Ilustración 26. Raspberry Pi to Arduino Shields Connection Bridge usada en el trabajo.	32
Ilustración 27. Montaje del conjunto quadrotor con el puente de conexiones instalado.	32
Ilustración 28. Montaje con el convertidor de tensiones incluido.	33
Ilustración 29. Convertidor de tensiones LM2596 DC-DC usado en el trabajo.....	33

Ilustración 30. 10 DOF IMU Sensor usado en el trabajo.	34
Ilustración 31. Montaje de la IMU sobre el quadrotor.	35
Ilustración 32. Montaje completo del quadrotor (sin protección).	35
Ilustración 33. Montaje completo del quadrotor (con protección).	36
Ilustración 34. Esquema general de las conexiones de alimentación.....	38
Ilustración 35. Esquema de un sistema de comunicación I2C.	39
Ilustración 36. Ejemplo de transferencia de datos en I2C, extraído de www.luisllamas.es	40
Ilustración 37. Esquema de conexión I2C entre la Raspberry y la IMU.	40
Ilustración 38. Diagramas de PWM.....	41
Ilustración 39. Esquema de conexión PWM entre Raspberry y Drivers.....	41
Ilustración 40. Esquema de conexión UDP entre Raspberry y el PC Base.	42
Ilustración 41. Esquema general del flujo de información.	43
Ilustración 42. Base de pruebas con quadrotor acoplado.	45
Ilustración 43. Listón y tablero usados para construir la base.....	46
Ilustración 44. Igubal GFSM-08-IG, extraída de www.igus.eu	46
Ilustración 45. Imagen del diseño 3D de la torre de placas.	47
Ilustración 46. Diseño 3D del quadrotor sin carcasa.....	47
Ilustración 47. Diseño 3D del quadrotor con carcasa.	48
Ilustración 48. Diseño 3D de la plataforma de pruebas completa.....	48
Ilustración 49. Imagen del diseño 3D de la unión entre el quadrotor y la base.	49
Ilustración 50. Aspecto gráfico de la interfaz HMI usada en el trabajo.	52
Ilustración 51. Esquema general de un bucle de control mediante PID.....	53
Ilustración 52. Esquema implementado del bucle de control mediante PID.....	54
Ilustración 53. Respuesta ante cambios de referencia en el ángulo de Roll.....	57
Ilustración 54. Respuesta ante perturbaciones y cambios de referencia en el ángulo de Roll.	58
Ilustración 55. Respuesta ante cambios de referencia en el ángulo de Pitch.....	59
Ilustración 56. Respuesta ante perturbaciones y cambios de referencia en el ángulo de Pitch.	60
Ilustración 57. Respuesta ante cambios de referencia en el ángulo de Yaw.	61
Ilustración 58. Respuesta ante perturbaciones y cambios de referencia en el ángulo de Yaw.....	62
Ilustración 59. Modelo CAD de la Raspberry Pi 2 Model B.	67
Ilustración 60. Modelado CAD del convertor de tensiones LM2596.	68

1. Introducción

1.1. Propuesta y objetivos

La redacción de este documento surge de la propuesta de realizar un Trabajo Final de Grado que consistiese en el desarrollo de un sistema de vuelo auto-tripulado. Para ello, se decidió llevar a cabo esta labor empleando nuevas unidades de procesamiento, con el fin de mejorar la respuesta del dispositivo y aprovechar la sencillez constructiva y el reducido coste de estos nuevos componentes. En el presente trabajo se propuso optar por emplear una Raspberry Pi 2 dado el auge que está teniendo a nivel global, debido a su potencia, bajo precio y a una vasta comunidad de desarrolladores mundial que está exprimiendo todas las posibilidades que ofrece esta pequeña unidad.

Paralelamente al desarrollo del quadrotor, se pretendía llevar a cabo un sistema de simulación de vuelo que permitiese estudiar la respuesta de cualquier dispositivo que se instalase sobre él, con el fin de comprobar la fiabilidad del algoritmo y del sistema físico en su totalidad, desde el propio laboratorio y de forma que no se pusiese en riesgo la integridad física del dispositivo ni la de las personas que lo manipulasen.

Ambos objetivos, por tanto, convergieron en el desarrollo global de una plataforma de ensayo que permitiese el testeo de diferentes algoritmos, diseñados para el control de orientación y altura de cualquier sistema quadrotor.

Debido a la extensión del proyecto propuesto, este se dividió en dos Trabajos Final de Grado. El presente trabajo, centrado en el diseño y desarrollo de hardware y soporte físico se ha llevado a cabo conjuntamente con otro alumno, quien se ha encargado de la implementación y gestión del software¹.

1.2. Estructura del documento

La memoria está estructurada en nueve capítulos principales, divididos cada uno de ellos en apartados que abordan puntos más concretos. A continuación se esboza brevemente el orden y la amplitud de los contenidos que aquí se presentan.

Se comenzará haciendo una pequeña introducción al estado del arte y se hablará de la historia, actualidad y futuro del mundo de los RPAS (Remotely Piloted Aircraft System).

¹ Para complementar el presente trabajo, véase “Diseño de una plataforma software para el control de orientación 3D de un sistema quadrotor basado en Raspberry Pi 2”, escrito por Rui Pedro Machado da Silva.

Después se presentará el modelo matemático de un quadrotor, con la intención de ofrecer una explicación teórica del comportamiento y funcionamiento básico de estos sistemas.

Tras ello, se pasará a abordar la constitución real de un dron. Se expondrán los elementos básicos y generales que lo componen, para después pasar a hablar de los componentes que conformarán el modelo diseñado para este trabajo.

También se comentarán las conexiones eléctricas y electrónicas que permiten conjugar todos los elementos en un sistema dependiente y totalmente integrado.

Una vez planteada toda la morfología de la unidad del quadrotor, se pasará a hablar de la base de pruebas. Se presentará el concepto en el que se basa dicha estructura, así como los materiales escogidos para su materialización.

Ya conocidos los diferentes elementos y componentes que darán lugar a la plataforma de control, se explicará y mostrará la fase de diseño previa a la ejecución.

Para completar la constitución del sistema quadrotor íntegro, se comentarán brevemente algunos de los aspectos más relevantes del software; sin entrar en detalles, pues este aspecto se escapa del alcance del TFG.

Tras haber expuesto la constitución total de la plataforma, compuesta por el quadrotor y la base de pruebas, se mostrarán los resultados obtenidos en los ensayos y se comentará la validez del algoritmo de control implementado en el sistema.

Por último, se hará una conclusión global de lo extraído tras la realización del trabajo y se expondrán algunas ideas surgidas a lo largo de todo el desarrollo, que se consideran interesantes para ser abordadas en futuros trabajos.

2. Estado del arte

2.1. Generalidades de quadrotores y UAVs

El quadrotor (o quadcopter) es una tipología específica de helicóptero que consigue el empuje necesario para el vuelo gracias a sus cuatro hélices. Esta solución constructiva fue de las primeras que se emplearon para desarrollar vehículos aéreos, dada la facilidad para contrarrestar los pares de torsión que genera el giro de las hélices sobre la estructura; para ello tan solo es necesario disponer de una pareja de hélices girando en sentido horario a la vez que las otras dos lo hacen en sentido anti-horario.

Los UAV (Unmanned Aerial Vehicle), como su nombre indica, son sistemas de vuelo no tripulado que han obtenido la mayor parte de su proyección en el ámbito militar. La capacidad para llevar a cabo operaciones de reconocimiento y ataque sin necesidad de comprometer la vida de pilotos, ha fomentado un gran desarrollo para estos sistemas a lo largo de los últimos años. Actualmente, se implementan también con motivos de carácter lúdico y civil (como lucha contra incendios o rescate).

2.2. Breve reseña histórica

2.2.1. Evolución de los quadrotores

Teóricamente, la solución constructiva que se plantea en los quadrotores parece ser la mejor para realizar vuelos verticales y mantener una adecuada estabilidad. Sin embargo, el desarrollo de estos modelos ha sido lento y no ha estado exento de dificultades.

La primera máquina de esta tipología que consiguió elevar el vuelo data de 1907, aunque tan solo despegó del suelo unos pocos metros. No fue hasta principios de los años 20 cuando aparecieron algunos prototipos, como el estadounidense “Bothezat” y el francés “Oemichen No.2”, que consiguiesen cierto control del desplazamiento y una estabilidad aceptable, siendo el modelo galo claramente superior en cuanto a diseño y prestaciones. Sin embargo, estas máquinas seguían teniendo numerosos problemas y limitaciones que los relegaban a simples curiosidades tecnológicas. Ya en 1956, Convertawings desarrolló su “Modelo A” (Ilustración 1), que presumía de unas características suficientes como para poder ser comercializado; pero la falta de interés por parte de los posibles clientes forzó a la empresa a cancelar el proyecto.

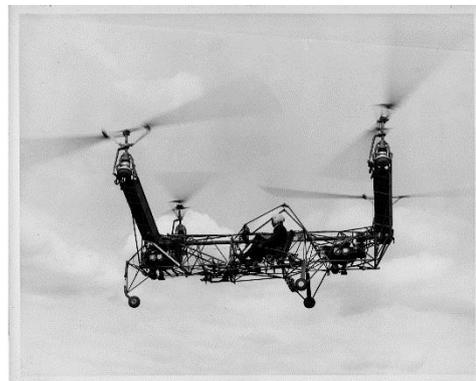


Ilustración 1. Modelo A, desarrollado por Convertawings.

2.2.2. Evolución de los UAV

Lejos de la idea actual de dron, el primer UAV concebido data de 1849; fecha en la cual los austríacos lanzaron una ofensiva contra la ciudad de Venecia, consistente en una flota de globos cargados con explosivos. Arcaico a la vez que improductivo (dado que este sistema dependía de la dirección del viento y muchos globos acabaron desviándose o regresando a territorio austríaco) pero con un gran potencial, este concepto comenzó a desarrollarse.

Durante la Primera Guerra Mundial se produjo una gran madurez de esta tecnología y entre los años 1916 y 1917 se diseñaron varios modelos funcionales. Uno de los más destacables fue el *"Hewitt-Sperry Automatic Airplane"*, que incorporaba giróscopos para llevar a cabo el control. Pero la guerra acabó antes de que estos nuevos sistemas pudieran producirse en masa.

Entre las dos grandes guerras se siguió avanzando sobre este campo. Se crearon algunos drones a partir de aeroplanos ya existentes, como el *"Standard E-1"* y se empezaron a usar los aviones radio-controlados como dianas aéreas para entrenamiento militar.

Con el inicio de la Segunda Guerra Mundial estalló la producción de estos dispositivos a manos de la empresa *Reginald Denny Industries*; pero su uso todavía se veía confinado a prácticas de tiro, a pesar de que el ejército estadounidense ya pensase en usarlos a modo de armas. Tan solo se realizaron algunos ataques menores, por miedo a que sus enemigos descubriesen el funcionamiento de esta tecnología y pudiesen desarrollar las defensas pertinentes.

Durante la Guerra Fría se continuó desarrollando nuevos modelos, que se irían perfeccionando con el tiempo. En este período, el principal uso que recibían era el de señuelo, pero también se utilizaron como herramientas de reconocimiento. Fue en la Guerra de Vietnam donde realmente se demostraron las bondades de este tipo de vehículos como sistemas de recolección de información. Sin embargo, a partir de los años 80 se empezaron a emplear como armas de guerra, alcanzando su máximo exponente con los *"Predator"* estadounidenses a mediados de los 90 (*Ilustración 2*).



Ilustración 2. Predator MQ-1.

2.3. Actualidad

Durante la primera década del presente siglo se empezaron a plantear utilidades que están lejos del ámbito que les dio vida. En los últimos años ha aumentado el uso de los UAV en el mundo de la fotografía, la investigación, el entretenimiento y la seguridad.

En la actualidad, se ha convertido en un recurso muy habitual el uso de drones para la filmación de imágenes y la toma de fotografías aéreas, por parte de las empresas audio-visuales, cinematográficas y fotográficas. Pero no solo las empresas con grandes presupuestos pueden disponer de ellos, ya que existen modelos más económicos que están al alcance de cualquiera. Por otra parte, su capacidad para volar en diversos ambientes y su reducido tamaño, han propiciado su uso en favor de la investigación científica en lugares de difícil acceso o para la obtención de información sobre acontecimientos atmosféricos adversos. Estas mismas capacidades los ha hecho una herramienta interesante para el rescate de personas en cualquier lugar del planeta.

Los nuevos intereses que rodean a este tipo de vehículos los ha llevado a requerir ciertas características. Se necesita que dispongan de una elevada maniobrabilidad y que su manejo y configuración resulten lo más sencillos posibles; con el fin de hacerlos accesibles a todo el mundo, reduciendo su complejidad de pilotaje y los costes asociados a su producción y mantenimiento. Es por ello que se les ha atribuido a estos dispositivos una configuración de quadrotor, que rara vez es alterada. Los quadrotores permiten un manejo, control y estabilidad que los hace ideales para estos desempeños.

2.4. Futuro

A la trayectoria histórica que hemos trazado, habría que añadirle cual es la proyección de futuro que se plantea para estos dispositivos.

Siguiendo la tendencia actual, se prevé una mejora en su explotación en el rescate. Ya no solo se espera que permitan localizar personas perdidas, sino que además, sean capaces de proporcionarles



Ilustración 3. Amazon PrimeAir.

medios suficientes para mantenerlos con vida hasta que la ayuda pueda materializarse de forma segura. En esta misma línea se han planteado otras ideas similares; como drones de primeros auxilios equipados con desfibriladores portátiles que puedan llegar en el mínimo tiempo posible a cualquier emergencia. Así como la entrega de víveres en poblados aislados durante las épocas invernales o el reparto de paquetes en mensajería, tal como tiene intención hacer la empresa Amazon con su proyecto PrimeAir (Ilustración 3).

2.5. Normativa y licencias

Una vez que los drones dejaron de tener interés únicamente en el ámbito militar e hicieron acto de presencia en actividades civiles, surgió la necesidad de redactar una serie de normativas que regulasen su uso; así como la aparición de licencias habilitantes para su pilotaje y vuelo en determinadas aplicaciones.

En España existe actualmente una ley (Real Decreto-ley 8/2014, de 4 de julio) que regula el uso de drones de menos de 150 kg (los únicos permitidos para uso no militar) con carácter comercial y civil. Para el caso de uso comercial, en el documento oficial se tratan: la tipología del dron, especificando en qué casos deben estar inscritos en el registro de aeronaves; las condiciones de uso del espacio aéreo, así como el proceso de la solicitud de vuelo a AESA (Agencia Estatal de Seguridad Aerea); los términos de seguridad que se deben cumplir y las sanciones económicas que puede acarrear la violación de estos términos; y por último, se establecen los estándares y requerimientos para la obtención del carnet oficial para el manejo de drones.

Como se ha especificado, este decreto ley tan solo afecta al uso comercial de drones; sin embargo, también existen ciertas recomendaciones para el uso personal y recreativo de estos dispositivos, que de no ser cumplidas, podrían acarrear multas de miles de euros. Entre estas normas se destacan las prohibiciones de vuelos a más de 120 metros de altura o sobre espacio aéreo perteneciente a aeropuertos o aeródromos, así como sobrevolar zonas urbanas públicas o aglomeraciones de personas, de modo que se pudiera poner en peligro la seguridad de terceros. A todo ello habría que añadir las restricciones que cada comunidad autónoma y municipio haya decidido establecer en su territorio.

La materia referente a normativa, por lo menos en cuanto a España se refiere, aún está en proceso de madurez y todavía existen puntos que no están completamente definidos. Por ello, se espera que el Ministerio de Fomento, en colaboración con AESA, redacte pronto una ley que defina totalmente el uso y restricciones de estos dispositivos y despeje las incertidumbres que existen respecto a la legalidad de su explotación.

2.6. Justificación del trabajo

La realización de este trabajo, como ya se indicó en el apartado de “Propuesta y objetivos”, queda justificada por diversos aspectos. El primero de ellos se focaliza en el desarrollo de un quadrotor controlado por un mini-ordenador Raspberry Pi 2, con el fin de continuar estudiando las aplicaciones y posibilidades que proporciona esta tecnología y poder probar la nueva versión comercial de la familia Raspberry Pi.

Por otra parte, se presentaba indispensable la utilización de una base de pruebas que permitiese testear la acción del algoritmo sin el requerimiento de tener que probar el quadrotor en vuelo real. La necesidad de llevar a cabo este módulo de pruebas surge de la presencia de diversos factores.

El primero de ellos, implica a las normativas y licencias que se han comentado en el apartado precedente. Al no estar permitido el vuelo en zonas públicas, resultaría imposible realizar ensayos en las proximidades del laboratorio, que se encuentra dentro del recinto de la UPV; y dadas la baja autonomía del dispositivo y la necesidad de utilizar cierto instrumental del laboratorio, también quedaría descartada la posibilidad de realizar estos vuelos en zonas alejadas. Por otra parte, al tratarse de vuelos de ensayo de un dispositivo en desarrollo, la probabilidad de la presencia de fallos sería elevada, lo que podría suponer un riesgo, tanto para el quadrotor, como para las personas presentes. Por último, se considerarían las comodidades que ofrece una base de pruebas que permitiese realizar los ensayos desde el propio laboratorio, con la consiguiente rapidez y sencillez de estos.

Por todo ello se tomó la decisión de utilizar este módulo de pruebas. La primera opción que se planteaba era la de adquirir un dispositivo comercial para tal efecto. Pero en el mercado no existe una gran oferta de estos productos, y los que existen, son excesivamente caros. Un ejemplo de ello es el 3 DOF Hover de Quanser (*Ilustración 4*) que permite testear los tres grados de libertad, referidos a la orientación, del dron. Sin embargo, su precio ronda los 21000€, por lo que se posiciona como una opción inviable. Fue por ello que se tomó la decisión de construir una base de pruebas propia, más sencilla, pero que igualmente permitiese probar el dispositivo en vuelo.



Ilustración 4. 3 DOF Hover, cortésia de Quanser.

3. Modelo matemático

En este apartado se van a presentar algunos aspectos teóricos sobre la física del movimiento en la que se basa el control y actuación de los quadrotores. No se pretende llegar a un modelo matemático completo que permita estudiar la dinámica ni elaborar un algoritmo de control analítico, debido a que la complejidad aerodinámica de este tipo de sistemas hace que abordar esta cuestión exceda las dimensiones del trabajo.

Antes de empezar a abordar los aspectos físicos, conviene hacer una presentación de algunos aspectos relevantes; como los ángulos de inclinación (que serán la piedra angular del trabajo, pues el control se realizará sobre ellos) y los ejes de referencia, así como su posición y configuración sobre nuestro quadrotor.

3.1. Configuración de los ejes en el quadrotor

Para todo elemento del que se vaya a realizar un control, seguimiento o estudio es necesario establecer una serie de sistemas de referencia que permitan conocer el estado de todos sus grados de libertad. En nuestro caso, se usarán dos sistemas de referencia. El primero permanecerá fijo al quadrotor, de forma que la posición de sus ejes proporcione información de la localización del dron, y será nuestro sistema de referencia móvil (x^m, y^m, z^m); el segundo se mantendrá inmóvil en un punto de la superficie terrestre, constituyendo el sistema de referencia global (x^g, y^g, z^g).

Los ejes de referencia globales, se considerarán dispuestos de la siguiente forma: el eje ' x^g ', tangente a la corteza terrestre, se encontrará señalando la posición del Norte; el eje ' y^g ', también tangente a la superficie, indicará la localización del Este; y por último, ' z^g ' apuntará al centro de la Tierra.

El sistema móvil resulta inherente al dispositivo; entonces, la cuestión que se plantea ahora es la manera en que este sistema estará unido al cuerpo. Normalmente se opta por uno de entre los dos controles más sencillos, que son los llamados controles en "cruz" y en "equis".

La configuración en "cruz" posiciona los ejes de tal modo que permanecen coincidentes a los brazos del quadrotor (*Ilustración 5*), mientras que la configuración en "equis" los posiciona girados 45° con respecto a los brazos (*Ilustración 6*). Para el desarrollo del modelo matemático se usará la segunda. Aunque el resultado final será igual que en el caso de haber usado la otra configuración, tan solo habría que aplicar un desfase de 45° sobre los ángulos de orientación (que se expresan sobre los ejes).

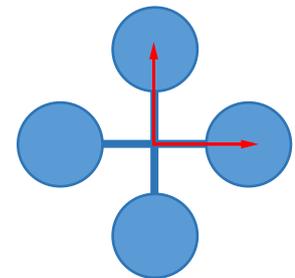


Ilustración 5. Configuración en cruz.

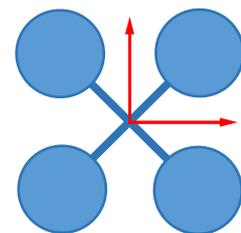


Ilustración 6. Configuración en equis.

3.2. Ángulos de posición

Para que el quadrotor pueda mantener el vuelo y no se precipite al vacío es necesario que este se mantenga íntegro y no experimente ningún tipo de deformación en su estructura. Partiendo del supuesto de que esto no sucederá, se puede considerar al quadrotor como un sólido rígido y por lo tanto, se afirmará que posee seis grados de libertad. Esto es, que para poder definir totalmente su posición en el espacio, será necesario conocer la localización de su centro de masas (mediante las coordenadas globales en ' x^g ', ' y^g ' y ' z^g ') y la orientación de su sistema de referencia local (mediante los ángulos de Euler).

Es necesario explicar detenidamente cómo se definirá la orientación, pues es una parte importante del control que se va a desarrollar como objeto de este trabajo.

La orientación quedará definida mediante los tres ángulos de Euler, que describirán la posición de los ejes que componen el sistema de referencia móvil (*Ilustración 7*). Estos ángulos son:

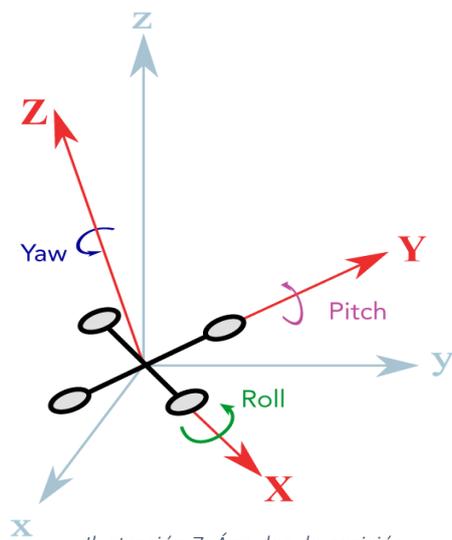


Ilustración 7. Ángulos de posición.

Roll. Es el ángulo que indica el giro respecto del eje ' x^g '. Se conoce como ángulo de alabeo y se representa con la letra ϕ .

Pitch. Ángulo que mide el giro sobre el eje ' y^g '. También se llama cabeceo y se expresa con la letra griega θ .

Yaw. Es el nombre del ángulo girado respecto el eje ' z^g ', también conocido como guiñada. Se denota por la letra ψ .

Consideraremos que, ante valores nulos de estos tres ángulos, los ejes del sistema de referencia móvil permanecerán paralelos a sus homólogos del sistema de referencia global.

3.3. Matrices de cambio de base

Ahora que ya se encuentran definidos los sistemas de referencia y se ha explicado la existencia y distribución de los ángulos de Euler, se hace necesario plantear unas herramientas (matrices) que nos permitan realizar un cambio de base; con el fin de trasladar las magnitudes medidas en uno de los sistemas al otro. La necesidad de esto radica en que algunas de las magnitudes con las que trabajaremos, como la fuerza de la gravedad, son conocidas sobre una referencia sin giros; mientras que otras magnitudes, como los ángulos de inclinación o el empuje de los rotores, se obtendrán sobre el sistema de referencia móvil generalmente girado.

3.3.1. Cambio de base de magnitudes lineales

El método que se ha seguido para llegar a esta matriz ha consistido en ir aplicando giros consecutivos respecto a los ángulos de Euler. Primero, se ha realizado un giro respecto del ángulo ϕ y se ha obtenido una matriz que correlaciona la nueva posición del sistema de referencia, con la posición inicial. Después se ha hecho lo mismo, pero girando esta vez sobre el ángulo θ . Finalmente, se ha girado el ángulo ψ . Una vez que se tienen las tres matrices, tan solo hay que combinarlas, para que den lugar a la matriz de cambio de base (L) que relacione la posición inicial sin ningún tipo de giro (x^0, y^0, z^0), con una posición genérica a la que se le han aplicado tres giros (x^1, y^1, z^1).

$$\begin{pmatrix} x^1 \\ y^1 \\ z^1 \end{pmatrix} = L \cdot \begin{pmatrix} x^0 \\ y^0 \\ z^0 \end{pmatrix} \rightarrow \begin{pmatrix} x^m \\ y^m \\ z^m \end{pmatrix} = L \cdot \begin{pmatrix} x^g \\ y^g \\ z^g \end{pmatrix} \quad (3.1)$$

Estando L definida como:

$$L = \begin{bmatrix} \cos\theta \cos\psi & \cos\theta \sin\psi & -\sin\theta \\ \sin\phi \sin\theta \cos\psi - \cos\phi \sin\psi & \sin\phi \sin\theta \sin\psi + \cos\phi \cos\psi & \sin\phi \cos\theta \\ \cos\phi \sin\theta \cos\psi + \sin\phi \sin\psi & \cos\phi \sin\theta \sin\psi - \sin\phi \cos\psi & \cos\phi \cos\theta \end{bmatrix} \quad (3.2)$$

Esta relación no solo es válida para traspasar coordenadas de un punto de un sistema de referencia a otro; también sirve para relacionar las velocidades lineales expresadas en un sistema y en otro.

$$\begin{pmatrix} v_x^1 \\ v_y^1 \\ v_z^1 \end{pmatrix} = L \cdot \begin{pmatrix} v_x^0 \\ v_y^0 \\ v_z^0 \end{pmatrix} \Rightarrow \begin{pmatrix} v_x^m \\ v_y^m \\ v_z^m \end{pmatrix} = L \cdot \begin{pmatrix} v_x^g \\ v_y^g \\ v_z^g \end{pmatrix} \quad (3.3)$$

3.3.2. Cambio de base de magnitudes angulares

Para establecer la relación entre las componentes ($\omega_x^m, \omega_y^m, \omega_z^m$) de una determinada velocidad angular expresada sobre el sistema móvil con las de esa misma velocidad referida al sistema global ($\dot{\phi}, \dot{\theta}, \dot{\psi}$) tan solo es necesario girar el sistema de referencia móvil respecto a los ángulos de Euler (ϕ, θ, ψ) de tal modo que la orientación de este sistema coincida con la del global. Después, solo falta proyectar la resultante de los componentes de ($\dot{\phi}, \dot{\theta}, \dot{\psi}$) sobre cada uno de los ejes de ($\omega_x^m, \omega_y^m, \omega_z^m$). De todo ello, tras despejar, resulta:

$$\begin{pmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix} = A \cdot \begin{pmatrix} \omega_x^m \\ \omega_y^m \\ \omega_z^m \end{pmatrix} \quad (3.3)$$

Siendo 'A' la matriz de cambio de las cantidades angulares:

$$A = \begin{bmatrix} 1 & \text{sen}\phi \tan\theta & \text{cos}\phi \tan\theta \\ 0 & \text{cos}\phi & -\text{sen}\phi \\ 0 & \text{sec}\phi \text{sec}\theta & \text{cos}\phi \text{sec}\theta \end{bmatrix} \quad (3.4)$$

Pero esta matriz se puede simplificar de forma aproximada. Dado que los ángulos de giro que se producirán en el quadrotor no deberán de ser muy grandes (serán menores de 20°), la relación entre los dos sistemas queda:

$$\begin{bmatrix} 1 & \text{sen}\phi \tan\theta & \text{cos}\phi \tan\theta \\ 0 & \text{cos}\phi & -\text{sen}\phi \\ 0 & \text{sec}\phi \text{sec}\theta & \text{cos}\phi \text{sec}\theta \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \rightarrow \begin{pmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix} = \begin{pmatrix} \omega_x^m \\ \omega_y^m \\ \omega_z^m \end{pmatrix} \quad (3.5)$$

3.4. Acciones aplicadas sobre el quadrotor

A partir de ahora, tendremos en cuenta la configuración de los ejes en equis, con el fin de que los resultados obtenidos en el estudio de fuerzas y pares se asemejen a las ecuaciones que finalmente se implementarán en el algoritmo de control. En la *Ilustración 8* se muestra la posición de los ejes, la distribución de los motores y el sentido de giro de cada uno de estos.

3.4.1. Empuje

Sobre el quadrotor actúa, principalmente, la fuerza de la gravedad, que intentará arrastrarlo hasta la superficie de la Tierra. Como respuesta a ello, el dron deberá generar una serie de fuerzas que le proporcionen el empuje necesario para contrarrestar la gravedad y así, poder levantar el vuelo y mantenerse en el aire. Este empuje será el resultado de la suma de los empujes individuales que desarrolle cada uno de los conjuntos motor-hélices:

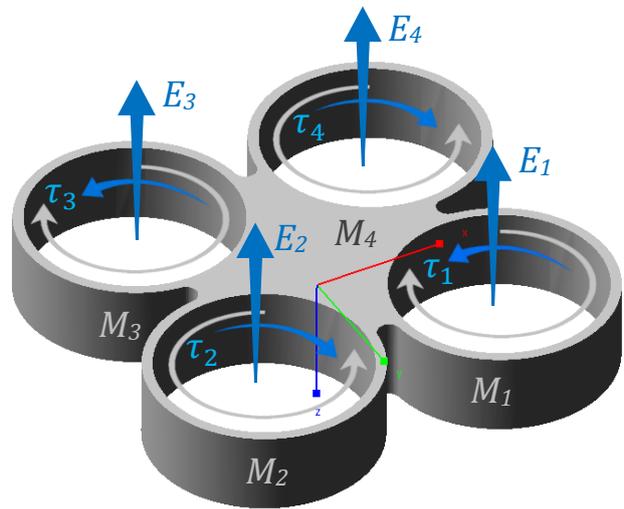


Ilustración 8. Empujes y pares del quadrotor.

$$E_T = E_1 + E_2 + E_3 + E_4 \quad (3.6)$$

Por su parte, el empuje es la reacción que provoca el aire sobre el quadrotor cuando, al girar, las hélices generan un flujo continuo del fluido. Es por ello que el empuje que aporta cada conjunto rotor es función de la velocidad de giro y del paso de la hélice. Siendo esta última constante, se puede establecer una relación directa entre la velocidad y el empuje, tal que:

$$E = k_E \cdot \omega^2 \quad (3.7)$$

Siendo 'E' el empuje de uno de los motores, 'k_E' una constante y ω la velocidad angular de dicho motor.

3.4.2. Par

El empuje queda definido sobre el sistema de referencia inercial, por lo que siempre tendrá el mismo sentido, independientemente de la combinación de las diferentes velocidades de giro de cada motor. Sin embargo, las fuerzas que provocan los motores, dado que están alejados del centro de masas, generarán unos pares sobre los tres ejes; dependiendo, cada uno de ellos, de la velocidad de giro de cada rotor en ese instante. Siendo 'l' la distancia entre dos motores consecutivos, los pares resultan:

$$\tau_x = (-E_1 - E_2 + E_3 + E_4) \cdot \frac{l}{2} \quad (3.9)$$

$$\tau_y = (+E_1 - E_2 - E_3 + E_4) \cdot \frac{l}{2} \quad (3.10)$$

$$\tau_z = (-\tau_1 + \tau_2 - \tau_3 + \tau_4) \quad (3.11)$$

3.5. Fuerzas y pares

Dado que el empuje se desarrollará gracias a la combinación de los empujes individuales de los cuatro motores del quadrotor, que siempre mantendrán la orientación de sus componentes vectoriales (en dirección y sentido) con respecto al sistema de referencia móvil, será mucho más sencillo expresar el sistema de fuerzas aplicadas desde el punto de vista del quadrotor. Esto también resulta lógico si se tiene en cuenta que la forma en que se controla la totalidad del movimiento del dispositivo es actuando directamente sobre la potencia que desarrolla cada uno de los motores; con lo cual, de esta forma se obtendrá una interpretación más rápida y sencilla de la acción que es necesario aplicar en cada momento.

3.5.1. Fuerza de la gravedad

Por las razones expuestas, se va a expresar la fuerza de la gravedad referida al sistema móvil. Para ello, se utilizará la matriz de cambio de base de las magnitudes lineales (L) que se ha obtenido en el apartado 3.3.1. Siendo ' F^g ' la fuerza de la gravedad en el sistema global y ' F^m ' la misma fuerza referida a los ejes solidarios al quadrotor, se tiene:

$$F^g = \begin{pmatrix} 0 \\ 0 \\ mg \end{pmatrix} \rightarrow F^m = L \cdot F^g \Rightarrow F^m = mg \cdot \begin{pmatrix} -\text{sen}\theta \\ \text{sen}\theta \cos\phi \\ \cos\theta \cos\phi \end{pmatrix} \quad (3.12)$$

3.5.2. Fuerza de empuje

Tal como se ha indicado en el primer párrafo del apartado 3.4.1, el empuje total resulta de la suma de los empujes parciales de cada rotor. Expresando esta fuerza de nuevo sobre los ejes del sistema móvil se tiene:

$$E^m = \begin{pmatrix} 0 \\ 0 \\ -E_T \end{pmatrix} \quad (3.13)$$

Como se puede apreciar, al referir las fuerzas respecto de este sistema, la expresión del empuje es mucho más sencilla; ya que sus componentes en los ejes ' x ' e ' y ' siempre tendrá un valor nulo.

3.5.3. Pares generados por los rotores

Como se explicó en el apartado 3.4.2, las combinaciones de los diferentes empujes de cada motor generan unos pares que tienden a hacer girar al quadrotor. Dada la configuración de nuestro sistema móvil (en equis), se pueden expresar estos pares resultantes como una combinación vectorial de las expresiones 3.9, 3.10 y 3.11:

$$\tau^m = \begin{pmatrix} (-E_1 - E_2 + E_3 + E_4) \cdot \frac{l}{2} \\ (+E_1 - E_2 - E_3 + E_4) \cdot \frac{l}{2} \\ (-\tau_1 + \tau_2 - \tau_3 + \tau_4) \end{pmatrix} \quad (3.14)$$

4. El quadrotor

4.1. Elementos básicos del quadrotor

En cualquier dispositivo del que se precise un funcionamiento autónomo, o incluso para aquellos de los cuales se pretenda controlar algunos parámetros, será necesario que el sistema sea capaz de controlar todas o algunas de las variables que influyen en su situación con respecto al entorno en el que se sitúa.

Para que un sistema cualquiera tenga “conciencia” de su estado con respecto al mundo real y pueda adaptarse a él de modo automático y suficiente, es necesario que presente tres elementos clave: sensores, controladores y actuadores; además de una estructura que permita recogerlos a todos ellos. También, podrá disponer de otros componentes que complementen sus funcionalidades y mejoren su funcionamiento.

A continuación se va a hablar de esos elementos, particularizados para el caso de un sistema quadrotor.

4.1.1. Sensores

Al igual que los humanos y el resto de animales disponemos de sentidos como la vista y el oído (para situarnos en el espacio) o el sentido del equilibrio (para mantenernos en una posición vertical); un sistema quadrotor debe poseer componentes que le permitan captar información análoga a esta. Principalmente, debe recibir información de su orientación y posición.

Por ejemplo, para que un dron sea capaz de mantenerse a una altura determinada por sí mismo, es necesario que en todo momento conozca a que altura se encuentra; de modo que, si se produce una variación de esa altura, pueda percibirla y corregirla adecuadamente.

Los sensores más básicos que se necesitan para este tipo de sistemas son los acelerómetros y los giróscopos (*Ilustración 9*), de los cuales se puede obtener información de la posición y la orientación, respectivamente; además, también permiten conocer las velocidades y aceleraciones lineales y angulares. Sin embargo, la información referente a la altura que proporcionan los acelerómetros de gama media-baja no suele ser muy precisa. Es por ello que se necesita disponer de otros sensores. Se suele utilizar sensores de ultrasonidos para vuelo en interiores, ya que proporcionan medidas muy precisas en un rango de 0 a 3 metros y necesitan volar sobre superficies planas y de poca inclinación (*Ilustración 10*). Para vuelos en exterior

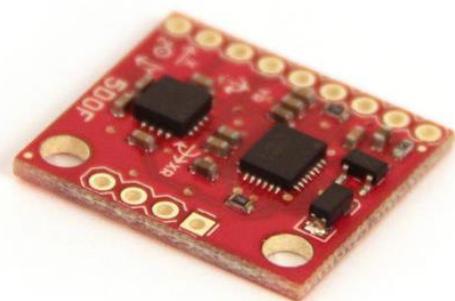


Ilustración 9. 5DOF Accelerometer Gyroscope, cortesía de SparkFun.

se emplean barómetros, que a pesar de tener errores en la lectura de algunos centímetros (de poca relevancia en alturas de más de 3 metros), su rango de actuación no tiene límites y la superficie que sobrevuelan no influye en la medida.

Para exteriores, el acelerómetro es insuficiente y con el tiempo produce derivas en las medidas². Es por ello que se emplean sensores GPS que proporcionan una elevada precisión, con errores no superiores a un metro. Cabe destacar que el uso de estos sensores en interiores resulta imposible, dado que la comunicación por satélite dentro de edificios es muy deficiente.



Ilustración 10. Sensor de ultrasonidos HC-SR04, cortesía de Cytron Technologies.

También pueden implementarse sensores láser para el control de posición, con elevadísima precisión; magnetómetros, para la posición en exteriores (en interiores las medidas se distorsionan debido a los campos electromagnéticos provocados por los elementos eléctricos); cámaras de vídeo, para el control manual, etc.

4.1.2. Controlador

Una vez se tiene información del estado del sistema, es necesario procesar esos datos para que el quadrotor adquiera “conciencia” de su estado y pueda actuar de una forma u otra, según se haya implementado en el algoritmo que gobierna el control.

En procesos sencillos, en los que no existen demasiadas variables de entrada y sus variaciones en el tiempo no son muy bruscas y/o no requieren de una respuesta rápida, suelen emplearse microcontroladores (Ilustración 11). Sin embargo, en el control de un quadrotor, el controlador va a recibir un gran flujo de datos procedente de diversos sensores y debe ser capaz de procesar toda esa información en un tiempo críticamente rápido para lograr una estabilidad adecuada del sistema. Es por esto que se precisa de controladores mucho más potentes, es decir, se necesitan microordenadores. Algunos ejemplos de este tipo de controladores son las placas IGEP, Arduino o la Raspberry Pi (Ilustración 12).



Ilustración 11. Microcontrolador, cortesía de Microchip.

² En metrología se denomina deriva a una desviación lenta y continuada de las medidas, ya sea por una tendencia viciosa del instrumento de medición o por la acumulación de errores.



Ilustración 12. IGEPv2, Arduino UNO, Raspberry Pi 2.

4.1.3. Actuadores

Tras el procesado de la información, se obtiene la respuesta que el sistema debe proporcionar para preservar el estado que se quiere mantener. Para ello, es necesario que se disponga de unos actuadores, capaces de interactuar físicamente con el medio. En el caso de los quadrotores, los actuadores más básicos e imprescindibles son los motores (*Ilustración 13*). A los drivers les llega la información, en forma de datos, de la acción que deben realizar, y estos transforman esa información en niveles de tensión que aplican directamente a los motores.



Ilustración 13. Brushless Outrunner Motors, cortesía de Scorpion Power System.

4.1.4. Chasis y protección

Los elementos que se acaban de comentar son esenciales para el funcionamiento de un quadrotor, pero para que el dispositivo forme un sistema íntegro se debe disponer de una estructura que los recoja y permita su interconexión. Por ello es necesario el uso de un chasis (*Ilustración 14*). En el mercado existen multitud de modelos y formas, pero todos ellos coinciden en su robustez y ligereza.



Ilustración 14. QAV250 G10 Mini FPV Quadcopter, cortesía de Lumenier.

También resulta habitual el uso de protecciones, cuya función principal es la de proteger al dron en general y a las hélices en especial.

4.1.5. Otros componentes

Según el uso que se le vaya a dar al dispositivo o el desempeño que tenga que realizar, podrá albergar una serie de elementos adicionales que no resultan imprescindibles para la gestión del vuelo, aunque sí podrían ayudar a mejorar su respuesta. Estos elementos pueden variar desde videocámaras de alta resolución para la grabación profesional de imágenes o la toma de fotografías, hasta la inclusión de desfibriladores para su uso como unidades de rápida intervención en emergencias.

Otros ejemplos, podrían ser: sensores infrarrojos para el estudio geotécnico y la vigilancia; sensores de proximidad para evitar posibles contactos físicos; brazos mecánicos que permitan sujetar y transportar cargas, etc.

4.2. Solución constructiva

A continuación se presentan los distintos componentes escogidos para la construcción de nuestro sistema quadrotor. Junto a sus características, se explicará el criterio de selección y se comentarán las razones de su elección.

El orden con el que se exponen ha sido escogido con el fin de mostrar la estructuración física del dispositivo y para poder visualizar mejor el proceso de montaje.

4.2.1. Chasis

Para la estructura se ha optado por utilizar un chasis comercial; en concreto, se ha montado el set de chasis mk40 del fabricante MikroKopter (*Ilustración 15*), sobre el cual se disponen, en forma de torre, las diferentes placas que componen el dispositivo. El conjunto está formado por cuatro tubos de aluminio de sección cuadrada perforados (tres negros y uno rojo), dos placas centrales y tornillería variada para el montaje. La distancia entre motores es de 40 cm y su peso máximo (dependiendo de la cantidad de tornillos y tuercas empleados) es de 110g, lo que lo hace bastante ligero.



Ilustración 15. Set de chasis mk40, cortesía de MikroKopter.

En la *Ilustración 16* se puede ver el chasis montado, incluyendo algunos tornillos y separadores que serán empleados para montar los distintos elementos sobre él.



Ilustración 16. Montaje del chasis.

4.2.2. Motores

A la hora de seleccionar un tipo de motor, los principales parámetros de selección que se deben tener en cuenta son el valor de su K_v y el empuje que es capaz de desarrollar en conjunto con las hélices. A continuación se explica brevemente cada uno de ellos.

El coeficiente K_v indica la capacidad del motor para transformar la potencia eléctrica que consume en velocidad. Esto quiere decir que dados dos motores (A y B) con la misma potencia pero de diferente K_v ($K_{vA} > K_{vB}$), se tendrá que aquel con un mayor K_v puede desarrollar una mayor velocidad ($v_A > v_B$) pero, por el contrario, proporcionará un menor par ($\omega_A < \omega_B$)³. Hay que tener presente que cada quadrotor requerirá más par o más velocidad según su peso y forma. Así, un dron pesado requerirá de mayor empuje para poder volar, aunque su velocidad de reacción será menor; por el contrario, un dron de bajo peso, no precisará tanto empuje y sí una mayor velocidad de reacción de sus motores. Una regla general para acotar la elección de un adecuado K_v en función del peso es:

Menos de 500 g → Entre 1300 y 2200 K_v

Entre 500 g y 1 Kg → Entre 900 y 1300 K_v

Más de 1 Kg → Entre 700 y 900 K_v

Por otro lado, tenemos un parámetro mucho más intuitivo; se trata del empuje. La función principal de los motores en un sistema de este tipo es elevar y mantener en el aire a todo el conjunto (aparte

³ Esto es así porque la potencia de giro en un eje es igual al producto del par y la velocidad angular $P = \omega \cdot \tau$.

de permitir su control); es por eso que estos deben ser capaces de vencer el peso del quadrotor. Pero no solo eso, también deben disponer de capacidad suficiente como para poder maniobrar sin tener que llegar a su potencia máxima⁴. Por eso, la regla general en el proceso de diseño y selección de componentes es que los motores sean capaces de desarrollar un empuje igual al doble del peso total del conjunto:

$$\text{Empuje de los motores} \geq 2 \cdot \text{Peso}$$

Sin embargo, el empuje no depende únicamente del motor, sino que depende, en gran medida, de las hélices que se coloquen en ellos. Así es, los motores proporcionan la potencia, pero son las hélices las encargadas de transformar esa potencia en un empuje útil; y esa transformación la realizan en función de su constitución aerodinámica⁵. Generalmente, los fabricantes proporcionan información sobre la fuerza que pueden proporcionar sus motores según los diferentes tipos de hélices que se les acople en distintos niveles de carga.

A parte de los aspectos citados, también hay que tener presente la corriente máxima que pueden llegar a demandar, ya que tanto el sistema de alimentación como los drivers deben ser capaces de proporcionar dicha intensidad. Es habitual que entre las especificaciones se aconseje el uso de unas baterías determinadas, con el fin de asegurar que a los motores llegará la corriente necesaria, y de unos drivers recomendados.

Los motores que se han escogido para este proyecto han sido los Brushless Yellow Trainer 1500 (Ilustración 17) que, con un peso de 45 g por motor, poseen un Kv de 1400 y son capaces de levantar un peso máximo de 1500 g; con un consumo de corriente de hasta 16 A. En este caso, el fabricante recomienda el uso de unas hélices de 8x4 para las especificaciones dadas. También se recomienda el uso de una batería de 2-3 celdas en caso de ser LiPo o una de 6-12 celdas en caso de ser NiMh.



Ilustración 17. Motores Brushless Yellow Trainer 1500 usados en el trabajo.

⁴ Trabajar durante un tiempo prolongado al máximo nivel de potencia puede sobrecalentar los bobinados de los motores y dañarlos. Además, si el control depende de un excesivo empuje, los drivers pueden llegar a saturar, dando lugar a serios problemas de estabilidad.

⁵ En el apartado siguiente 4.2.3. Hélices se hablará más detenidamente sobre este tema.

4.2.3. Hélices

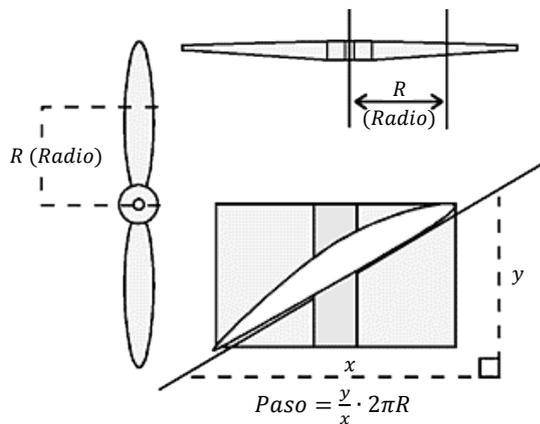


Ilustración 18. Definición de 'paso', cortesía de www.bolly.com.au.

Como se ha señalado en el apartado anterior, las hélices se encargan de materializar el empuje necesario para la sustentación del quadrotor. Sin embargo, y dada la estrecha relación que mantienen con los motores, las dimensiones y características aerodinámicas requeridas vienen propuestas por los fabricantes de estos, según el empuje a desarrollar. Es por eso que las hélices también juegan un papel importante en la selección de los motores, dado que la combinación escogida proporcionará unas cualidades determinadas.

El efecto aerodinámico de las hélices depende de la superficie barrida y del paso. Dada una inclinación de la sección, cuanto mayor sea la anchura, y sobre todo, la longitud de la pala, mayor será el volumen de aire desplazado, generando un mayor empuje. El paso se define como la distancia longitudinal que recorrería la hélice al girar una revolución (Ilustración 18). Para valores altos de paso se tendrá un mayor par, aunque la velocidad de giro se reducirá; y al contrario sucederá con valores de paso pequeños.

Se debe buscar una combinación adecuada entre longitud de pala, paso y potencia del motor. Una potencia grande de motor supondrá un mayor consumo de corriente y precisará de baterías de mayor calibre, con el consiguiente aumento del coste, espacio necesario y peso. Palas muy largas requerirán mucho más espacio y harán que aumente considerablemente el tamaño del quadrotor. Los pasos de valores altos darán lugar a mayores pares torsores y provocarán más vibraciones e inestabilidades (Ilustración 19).

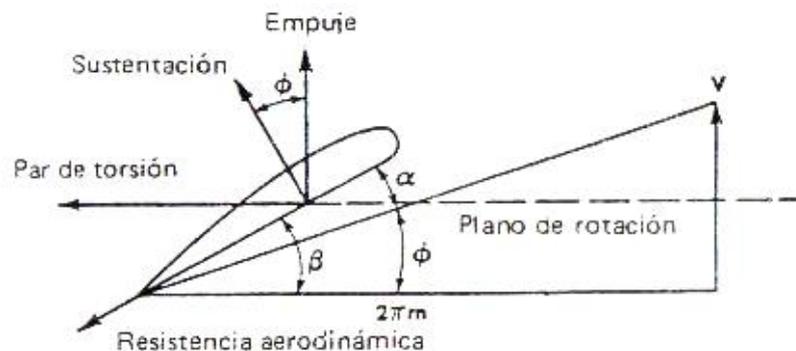


Ilustración 19. Diagrama vectorial de fuerzas y pares en la sección de una hélice, cortesía de www.alasewm.com.ar.

La nomenclatura que se suele emplear para caracterizar las dimensiones de las hélices es la siguiente:

Longitud total (pulgadas) × Paso de la hélice (pulgadas)

En nuestro caso, como el fabricante de los motores recomendaba montar unas hélices de 8x4 (8 pulgadas de envergadura y 4 pulgadas de paso) se ha decidido optar por unas de 8x4.5, para proporcionar un pequeño empuje extra (*Ilustración 20*).

Como se puede apreciar en la imagen referida, el juego de hélices está formado por un par de sentido de giro horario (CW) y por otro par de giro anti-horario (CCW). Esto es necesario para poder compensar los pares torsores que se provocarán al producir el empuje.



Ilustración 20. Conjunto de hélices y porta hélices usados en el trabajo.

La *Ilustración 21* muestra el conjunto de motores y hélices montados sobre el chasis.



Ilustración 21. Montaje de los motores y hélices sobre el chasis.

4.2.4. Drivers

Los motores giran con mayor o menor potencia dependiendo de la tensión que se les aplica, siendo esta, generalmente, de un orden de magnitud de decenas de voltios. La encargada de ordenar a los motores aumentar o reducir la velocidad es la CPU, pero esta envía esa información en forma de datos, de modo que su mayor tensión en la salida del conector no podría superar los 5 V (que se corresponderían con el estado a nivel alto). Tanto el nivel de tensión como la naturaleza de las señales⁶ imposibilitan la realización de una conexión directa. Es por esto que son necesarios unos intermediarios, que interpreten la información procedente del procesador y la conviertan en señales eléctricas admitidas por los motores. Estos elementos son los drivers, también llamados controladores, variadores o ESC (Electronic Speed Controller).

La CPU manda una señal PWM⁷ a los drivers con la información de potencia con la que deben actuar los motores, y estos transforman esa información en niveles de tensión aplicables directamente a los motores. Para ello, los drivers se alimentan directamente desde la batería o desde la fuente de alimentación, aunque tengan una línea de comunicación conectada al procesador.

El principal aspecto a considerar en la selección de los ESC es que la corriente máxima que puedan trasegar sea suficiente para abastecer la demanda de los motores a máxima potencia. Por lo demás, y a pesar de que algunos poseen mejoras respecto a otros (como filtros de tensión), la elección de un tipo u otro no es demasiado relevante. Aun así, es habitual que los fabricantes de motores recomienden el uso de determinados modelos.

Para el presente trabajo se han empleado unos drivers Turnigy Basic 25A (*Ilustración 22*) de 25 A de corriente nominal y 28 A de corriente máxima, con un peso de 22 gramos cada uno.



Ilustración 22. Drivers Turnigy Basic 25A usados en el trabajo.

⁶ El motor demanda una señal analógica en un rango de tensiones mayor mientras que la CPU proporciona una señal digital en un rango de tensiones mucho menor, por no hablar de la imposibilidad de proporcionar la corriente necesaria para alimentar los motores.

⁷ Este tipo de comunicación se explica con detenimiento en el apartado 5.2.2. *Señales PWM*.

En la *Ilustración 23* se puede ver el montaje con los drivers instalados en el chasis y conectados a los motores.



Ilustración 23. Montaje del chasis con motores y drivers instalados.

4.2.5. Raspberry Pi 2

Esta placa es la encargada de recibir y gestionar la información de los sensores, procesar los datos y controlar a los actuadores. Por otra parte, también permite la comunicación con un PC a través de una conexión wifi.

La decisión de usar una Raspberry Pi proviene, como se explicó en el apartado de “Propuestas y objetivos”, de su gran relación potencia/precio (su valor ronda los 40€) y de la amplia comunidad de usuarios que hay tras ella; la cual aporta un flujo constante de ayuda y nuevas ideas.

La Raspberry fue concebida con aspiraciones educativas, por lo que se pretendía ofrecer un ordenador lo suficientemente potente a un precio muy reducido de forma que fuese accesible por todo el mundo; y es precisamente esto, junto con la inclusión de puertos GPIO⁸, lo que ha propiciado su gran éxito.

⁸ Los puertos GPIO (General Purpose Input/Output) son puertos de comunicación digitales programables por el usuario. Pueden activarse y desactivarse, funcionar como entrada o salida e incluso se pueden activar como pines de interrupciones.

Existen varias versiones y modelos de esta placa; tres versiones principales y una versión reducida. Cualquiera de ellas sería válida para el desarrollo de un dispositivo como el que nos ocupa. La elección de la versión Raspberry Pi 2 Modelo B (*Ilustración 24*) proviene de la intención de probar las mejoras implementadas, ya que, en trabajos anteriores, se había probado la Raspberry Pi. Se descartó el uso de la Raspberry Pi 3 porque las únicas mejoras sensiblemente apreciables consistían en que esta placa incluye un receptor wifi incorporado y dispone de comunicación Bluetooth; de los cuales, tan solo el wifi presentaría interés para nuestro propósito, sin embargo, con un dispositivo USB-Wifi se puede realizar la misma función.



Ilustración 24. Raspberry Pi 2 usada en el trabajo.

La versión utilizada monta un procesador ARM Cortex-A7 de cuatro núcleos que funciona a 900MHz y dispone de 1 GB de RAM. En cuanto a la conectividad de entrada y salida de datos, posee cuarenta pines GPIO, cuatro puertos USB, un puerto Ethernet, un puerto HDMI, conexión para jack de audio de 3.5mm y video compuesto y una ranura para tarjetas SD.

Gracias a su procesador y a su núcleo gráfico VideoCore IV 3D, es posible instalar en ella toda la gama de distribuciones ARM GNU/Linux e incluso existe una versión del sistema operativo Windows 10 pensada expresamente para esta placa. En nuestro caso, se ha decidido instalar Raspbian (un SO en base Linux) junto con el parche Preempt_RT, debido al requerimiento de disponer de un sistema operativo en tiempo real⁹.

Se puede ver el montaje de la Raspberry Pi sobre la estructura en la *Ilustración 25*.



Ilustración 25. Montaje de la Raspberry Pi sobre el chasis.

⁹ En el apartado *8.1. Sistema Operativo en Tiempo Real* se explicará más detenidamente la necesidad de utilizar un SOTR.

4.2.6. Puente de conexiones Raspberry-Arduino

La placa utilizada para redistribuir los pines de la Raspberry y convertirlos en una distribución típica de Arduino fue concebida con la intención de poder disponer sobre ella módulos diseñados e implementados para Arduino. En nuestro caso, al no tener intención de acoplar ninguno de esos módulos, esta placa no posee mayor funcionalidad que la de hacer la conexión mucho más sencilla. Sin embargo, se ha optado por incluirla con el fin de facilitar el conexionado de las líneas de información y disponer de convertidores analógico-digitales en caso de ser necesarios en trabajos futuros; aunque en el presente proyecto es totalmente prescindible.

En el quadrotor se ha montado el Raspberry Pi to Arduino Shields Connection Bridge de Cooking Hacks (*Ilustración 26 Izda.*). Se trata de la segunda versión de este modelo y dispone de 8 pines digitales, pines RX/TX, pines I2C (SDA, SCL), pines SPI que pueden ser utilizados como GPIO y 8 canales de convertidores analógico-digitales.

Se puede ver el resultado de su acoplamiento en la Raspberry Pi a la derecha de la *Ilustración 26* y su instalación sobre el conjunto del quadrotor en la *Ilustración 28*.

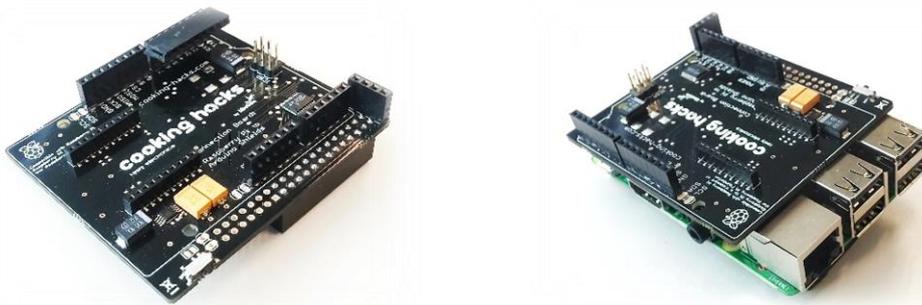


Ilustración 26. Raspberry Pi to Arduino Shields Connection Bridge usada en el trabajo (Izda.) y acoplamiento sobre la Raspberry (Dcha.).



Ilustración 27. Montaje del conjunto quadrotor con el puente de conexiones instalado.

4.2.7. Conversor de tensiones

Con el fin de poder suministrar a todos los elementos eléctricos y electrónicos su tensión necesaria desde una única fuente de alimentación se ha decidido instalar un convertidor de tensiones. El propósito de este componente no es otro que el de transformar la tensión de 11.1 V, suministrados por la fuente y necesarios para mover los motores, en una tensión de 5 V, requeridos por la Raspberry.

Para este montaje se ha utilizado el circuito convertidor LM2596 DC-DC Buck Step Down (*Ilustración 28*) que admite una tensión de entrada de entre 4.5 y 40 V de corriente continua y proporciona una tensión de salida de entre 1.5 y 35 V de continua ajustables. Pudiendo suplir una intensidad máxima de 3 A.

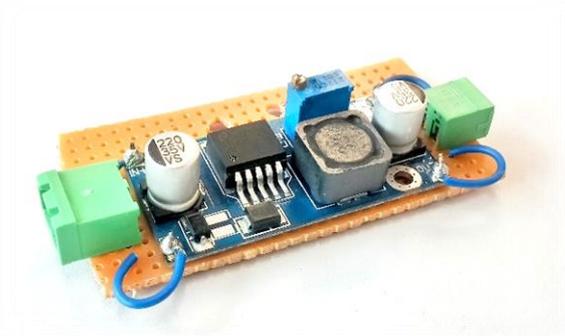


Ilustración 29. Convertidor de tensiones LM2596 DC-DC usado en el trabajo.

En la *Ilustración 29* se muestra el sistema quadrotor con dicho convertidor instalado.



Ilustración 28. Montaje con el convertidor de tensiones incluido.

4.2.8. IMU

Dados el alcance y el propósito del presente trabajo, en el quadrotor tan solo se va a instalar un dispositivo de medición que nos aporte información sobre la posición y movimiento del sistema. Para ello se va a emplear una IMU (Unidad de Medición Inercial). Este aparato combina el uso de acelerómetros y giróscopos para proporcionar información de la aceleración lineal y la inclinación. Dependiendo de la IMU (de su calidad y precio), se tendrá información de más o menos grados de libertad; por ejemplo, hay sensores con 5 DOF¹⁰ que proporcionan los valores de los giros respecto a los tres ejes de coordenadas y tan solo los valores de la posición sobre los ejes 'x' e 'y', mientras que otros proporcionan hasta 9 DOF al incluir magnetómetros con información sobre los tres ejes del sistema cartesiano.

Constructivamente, resulta interesante indicar que la IMU debe encontrarse sobre el plano de las hélices o muy cercano a él, con el propósito de obtener mediciones sobre los giros lo más puras posibles (es decir, sin componentes lineales que hubiera que corregir).



Ilustración 30. 10 DOF IMU Sensor usado en el trabajo.

En el desarrollo de nuestro trabajo se ha utilizado una IMU de diez grados de libertad; en concreto, se ha empleado un 10 DOF IMU Sensor (*Ilustración 30*). En realidad se trata de un total de tres sensores integrados, que proporcionan conjuntamente información de los giros, aceleraciones y campos magnéticos sobre los 3 ejes coordenados, además de aportar datos sobre la presión mediante un barómetro. Estos sensores son: un acelerómetro y un giróscopo, ambos integrados en el MPU6050, y que proporcionan datos de 6 DOF; un magnetómetro (HMC5883L) que aporta otros 3 DOF y un barómetro (MS5611), que adiciona el décimo DOF.

En la *Ilustración 31* se puede ver la IMU acoplada al montaje del quadrotor.

De todos los sensores, tan solo se van a utilizar el MPU6050 (para obtener información sobre los giros) y el MS5611 (para conocer la altura). Los giros se leerán directamente del sensor, aunque será necesario implementar un filtro de Kalman para poder interpretarlos correctamente; mientras que el barómetro, junto con la lectura de un termómetro que incluye el sensor, proporciona datos sobre la altura.

Experimentalmente se puede comprobar que las lecturas de los ángulos son bastante precisas. En cuanto a la altura, se aprecia que responde bien a los cambios de posición, pero presenta un error de

¹⁰ DOF son las siglas de Degrees Of Freedom.

lectura de ± 10 cm aproximadamente. Esto no supone un error importante para vuelos en exterior de más de 5 metros (donde el error sería inferior al 2%), pero en interiores resulta un error importante. Por lo que resultaría interesante incluir un sensor de ultrasonidos, que se encargase de realizar medidas mucho más precisas en alturas pequeñas



Ilustración 31. Montaje de la IMU sobre el quadrotor.

A continuación se muestra una imagen del quadrotor completo con una placa extra de protección para la IMU.



Ilustración 32. Montaje completo del quadrotor (sin protección).

Para acabar, se presenta una imagen del quadrotor con la carcasa de protección (*Ilustración 33*). Esta protección ha sido reutilizada de un proyecto anterior. Para adaptarla a nuestras necesidades se le han realizado una serie de modificaciones; como la eliminación de material en ciertas zonas, la inclusión de un sistema de contacto magnético (para poder fijarla y extraerla cómodamente) y el aporte de una capa de pintura.



Ilustración 33. Montaje completo del quadrotor (con protección).

5. Conexiones y comunicación

Ahora que ya se han presentado todos los elementos físicos que formarán parte del sistema, se va a esbozar el modo en que estos se comunican entre sí: tipos de comunicación, flujos de datos, conexiones eléctricas, etc. Con esto quedará definida la parte del hardware que compone el quadrotor.

5.1. Conexiones eléctricas y alimentación

En el esquema general del montaje se tienen diferentes componentes de naturaleza eléctrica o electrónica, y por lo tanto, todos ellos necesitarán estar conectados a una fuente de alimentación que les proporcione la tensión necesaria para su funcionamiento. Sin embargo, no todos precisan del mismo potencial. Estas son las tensiones de trabajo de cada uno de los distintos elementos:

Motores: los cuatro motores estarán alimentados a 11.1 V.

Drivers: son los encargados de gestionar la tensión que recibirán los motores, por ello necesitarán estar alimentados a la tensión de estos (11.1 V). De esa alimentación tomarán también la energía necesaria para su propio funcionamiento.

Raspberry Pi: la placa de control estará alimentada a 5 V. Es importante que la tensión no supere este valor, porque con el más mínimo aumento, la CPU sufre una sobretensión e incurre en un funcionamiento errático. Por ello es preferible trabajar a niveles de tensión ligeramente inferiores (entre 4.8 y 4.9 V).

IMU: necesitará estar conectada a 3.3 V, pero será la propia Raspberry la que suministre dicha tensión.

En resumen, a los drivers llegará una tensión de 11.1 V, y estos alimentarán a los motores, mientras que a la Raspberry entrará una tensión de 5 V y de ella saldrán los 3.3 V que demanda la IMU. Por lo tanto, se alimentará al sistema desde una fuente de alimentación que aporte 11.1 V para los motores y se realizará una derivación que pasará por el convertidor de tensiones para reducirlos en los 5 V necesarios para la CPU.

En el siguiente esquema se puede ver el flujo de la alimentación descrito anteriormente, con su paso por los distintos elementos y las diferentes transformaciones.

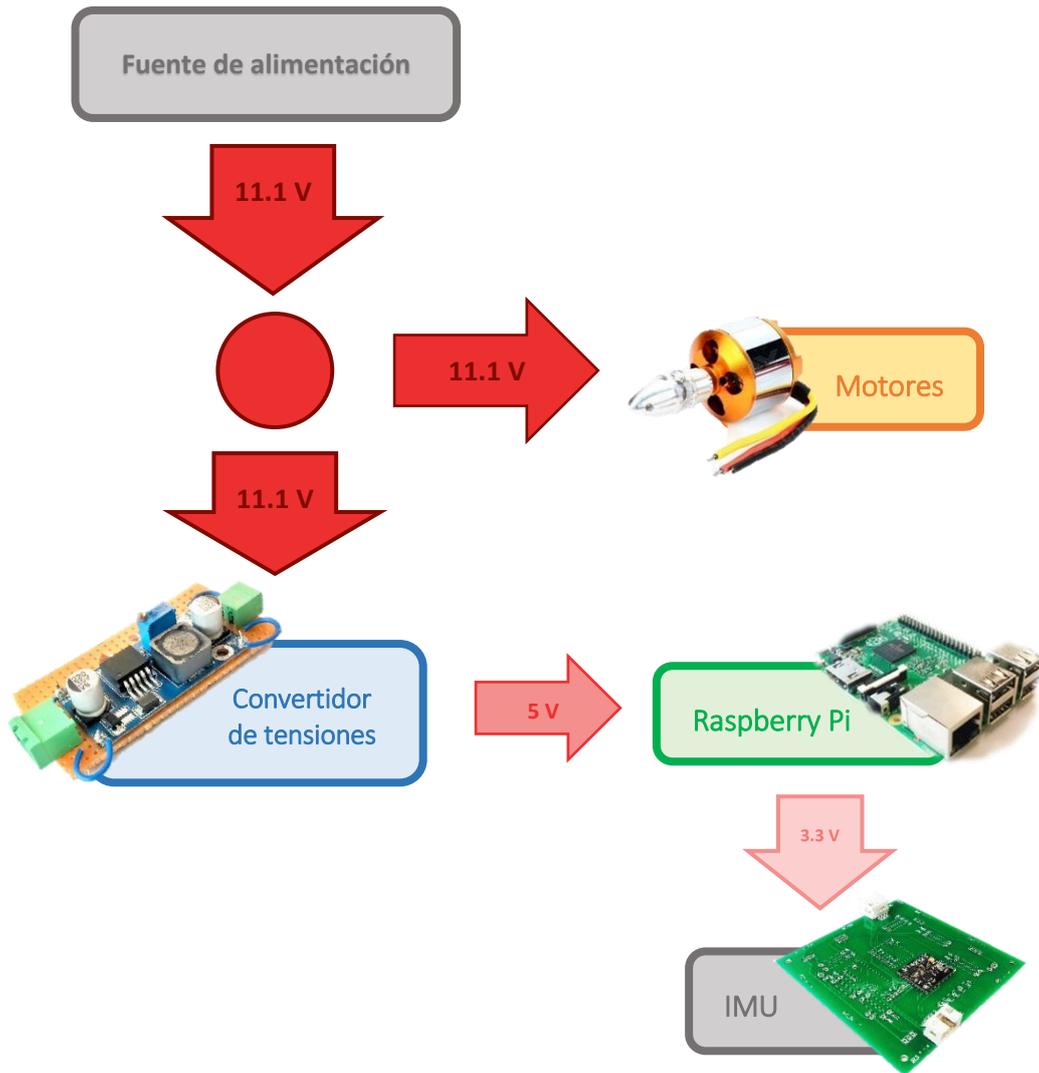


Ilustración 34. Esquema general de las conexiones de alimentación.

5.2. Comunicación y transferencia de información

Al igual que cada componente requiere de una tensión de alimentación específica, también se comunican según protocolos de comunicación diferentes para cada caso. Así, la IMU envía datos en protocolo I2C, mientras que los drivers solo pueden leer señales PWM y la Raspberry transfiere y recibe datos de la interfaz hombre-máquina mediante comunicación UDM.

Seguidamente se describen brevemente las bases y funcionamiento de cada una de ellas.

5.2.1. Comunicación I2C

La comunicación I2C consiste en un protocolo de transferencia síncrona de datos en serie, formada por dos líneas principales, una destinada a los datos (SDA) y otra reservada a la señal de reloj (SCL), y una tercera línea de masa (GND). Todos los dispositivos se encuentran conectados a estas mismas

líneas y aquellos con características de Maestro (con habilidad de iniciar la comunicación) pueden enviar datos a los Esclavos (habilitados solamente para responder a una solicitud de comunicación) que ellos seleccionen, cuando la línea esté desocupada. (Ilustración 35).

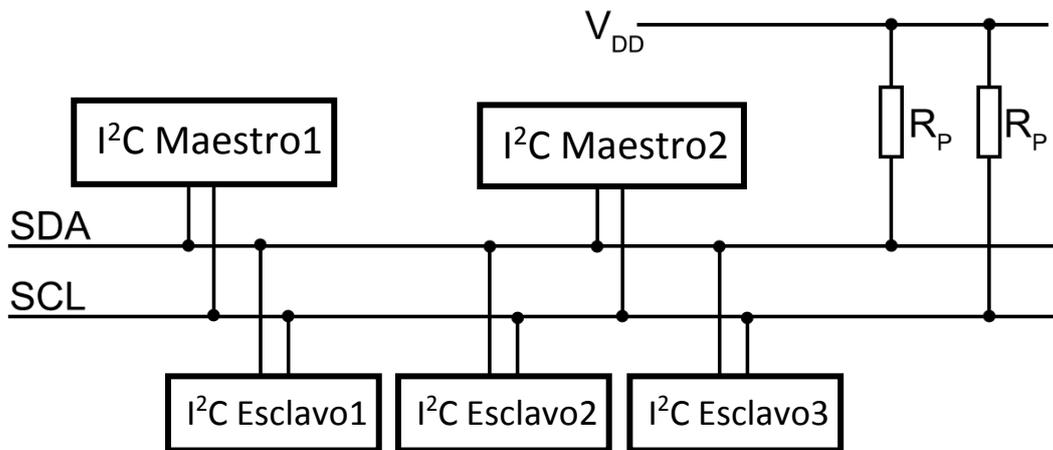


Ilustración 35. Esquema de un sistema de comunicación I2C.

Cuando las líneas están libres, ambas se encuentran a nivel alto¹¹ y cualquier Maestro puede ocuparlas para enviar datos. Para ello, el Maestro pone a nivel bajo la línea SDA, inicia una secuencia de reloj, y seguidamente envía la dirección del esclavo con el que quiere comunicarse (reservando el último bit para indicar si se va a leer o escribir). Tras esto, deja la línea SDA a nivel alto. Si en dicha dirección hay un esclavo disponible, este pondrá la línea de datos a cero (acción de reconocimiento, ACK) y se procederá a la transmisión de datos. Si la acción era de escritura, el Maestro enviará datos al Esclavo mientras siga recibiendo acciones de reconocimiento o hasta que acabe la transmisión; si era de lectura, recibirá datos del Esclavo hasta que deje de recibir acciones de reconocimiento o el Maestro decida parar. La secuencia de parada la lleva a cabo este último y consiste en poner a nivel alto la línea SDA durante dos ciclos de reloj. Por otra parte, si lo que quiere es reiniciar la secuencia de transmisión, ejecuta la secuencia de Inicio Reiterado poniendo la línea SDA a nivel alto durante un único ciclo de reloj, reanudado el proceso desde el principio. En la Ilustración 36 se muestra un ejemplo.

¹¹ Las líneas se encuentran a nivel alto mediante la inclusión de la resistencia Pull-Up.

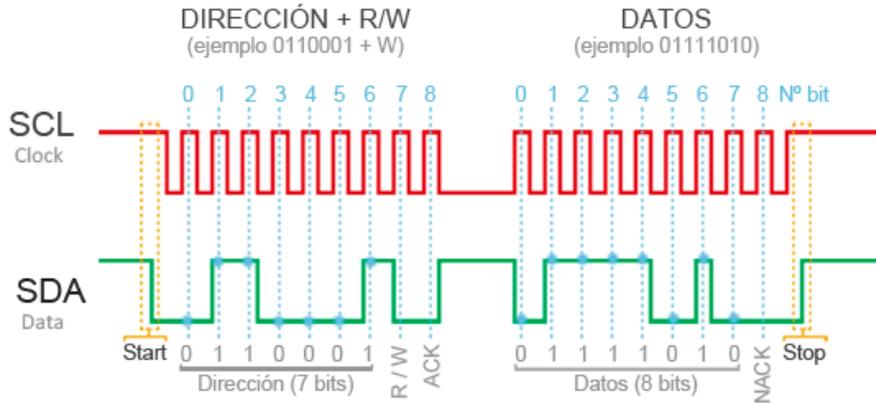


Ilustración 36. Ejemplo de transferencia de datos en I2C, extraído de www.luisllamas.es.

La Raspberry posee unas direcciones específicas para la comunicación I2C, por lo que basta con habilitarla y enlazar las entradas provenientes de la IMU (que se comunica por este protocolo) para obtener sus lecturas. La conexión física se materializa según el siguiente esquema:

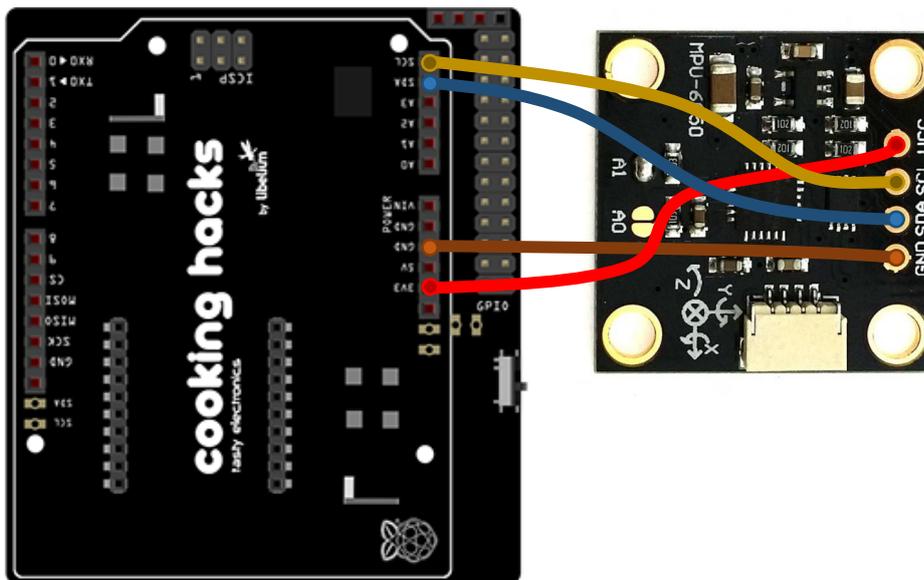


Ilustración 37. Esquema de conexión I2C entre la Raspberry y la IMU.

5.2.2. Señales PWM

Las señales PWM (Pulse-Width Modulation) consisten en señales de onda cuadrada periódicas con ancho de pulso variable, utilizadas para enviar información o variar la alimentación de una carga. Al modificar el ancho del pulso (tiempo que la señal se mantiene a nivel alto), se varía el valor eficaz de la tensión que se envía al receptor (Ilustración 38).

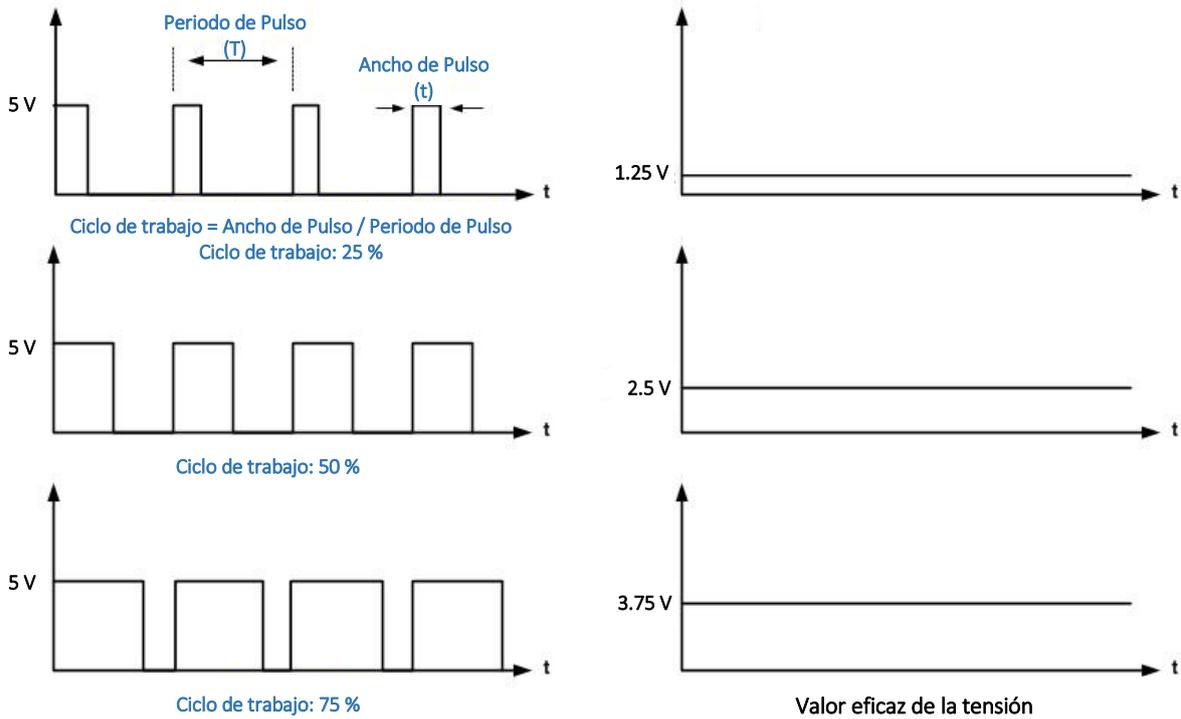


Ilustración 38. Diagramas de PWM.

En nuestro caso, la conexión PWM se realiza entre la Raspberry y los drivers. A estos les llega un nivel de tensión eficaz, según el cual, regulan la potencia que mandan a los motores.

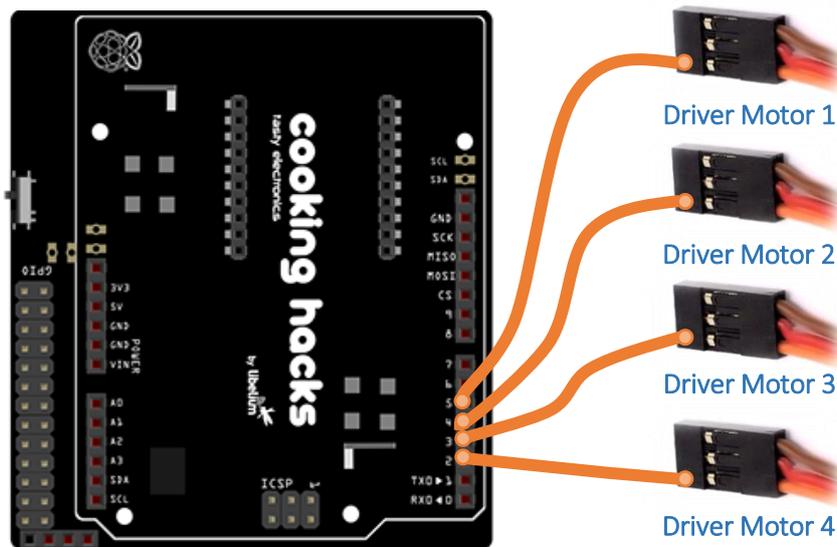


Ilustración 39. Esquema de conexión PWM entre Raspberry y Drivers.

5.2.3. Protocolo UDP

La comunicación UDP (User Datagram Protocol) consiste en un intercambio de información sin un establecimiento de conexión previa. Empleada en aquellos casos en los que la velocidad de transferencia es crucial, ya que no necesita emplear tiempo en conectar los dispositivos emisores y receptores.

Para el envío de información, el emisor manda un paquete de datos que van precedidos por un encabezado. En este encabezado se presenta: la dirección del puerto de origen, la dirección del puerto de destino, longitud total del paquete (incluyendo el encabezado y los datos) y una suma de comprobación del encabezado (permite comprobar la integridad del segmento). En total, el encabezado utiliza 64 bits (16 para cada una de las partes). Si un receptor quiere responder, tan solo tiene que enviar un paquete de datos al remitente.

En nuestro caso, este sistema de comunicación se realizará mediante una red wifi generada desde la Raspberry y se empleará para permitir la transferencia de información entre esta y un PC estático controlado por el usuario, mediante una interfaz hombre-máquina (HMI)¹². Para conseguir la generación de esa red, será necesario el uso de un dispositivo USB Wifi.



Ilustración 40. Esquema de conexión UDP entre Raspberry y el PC Base.

¹² En el apartado 8.2. *Comunicación con el quadrotor (HMI)*, se hablará sobre esta interfaz.

5.2.4. Esquema general del flujo de información

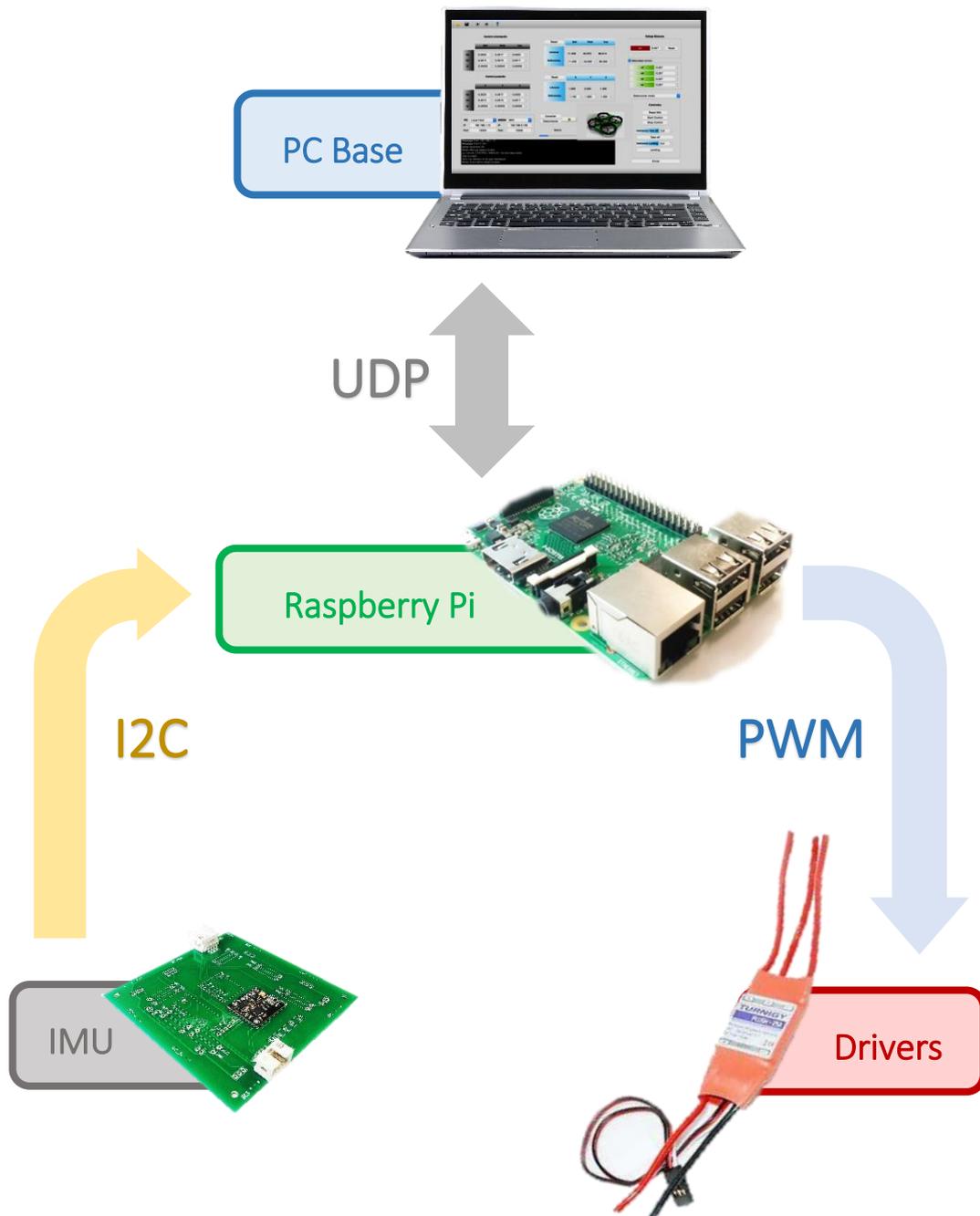


Ilustración 41. Esquema general del flujo de información.

6. Base de pruebas

6.1. Concepto

El propósito principal del trabajo consiste en el desarrollo de un sistema que implemente el control de orientación basado en Raspberry Pi (para lo cual se ha diseñado el quadrotor), junto con una plataforma capaz de testear la respuesta del algoritmo de control sin necesidad de realizar vuelos reales.

Debido al requerimiento de que pudiese ser estudiado el comportamiento de un quadrotor respecto a los tres ejes de giro, era necesario llevar a cabo una estructura que permitiese el movimiento del dron con, al menos, tres grados de libertad correspondientes a los giros. Para ello, se coincidió en que la mejor idea sería la de basar toda la base de pruebas entorno al uso de una rótula esférica tridimensional; partiendo de ideas similares presentes en modelos comerciales (*Ilustración 4*), pero con una ejecución diferente (*Ilustración 42*).



Ilustración 42. Base de pruebas con quadrotor acoplado.

Por otra parte, pareció interesante incluir un sistema de altura telescópica con el que poder ensayar la respuesta y control del quadrotor en referencias verticales. De esta manera, no solo se estaría llevando a cabo una base de pruebas equiparable a las comerciales con un coste mucho más reducido, sino que se estaría mejorando el concepto que estas implementan.

6.2. Materiales

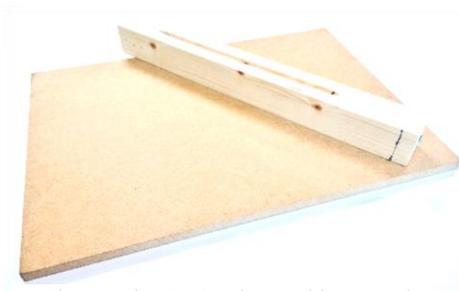


Ilustración 43. Listón y tablero usados para construir la base.

6.2.1. Base y mástil

Para sujetar la plataforma sin necesidad de fijar toda la estructura a una superficie, se ha optado por disponer de un pie (consistente en una tabla de aglomerado) de dimensiones lo suficientemente grandes como para compensar los momentos que se produzcan durante los ensayos y que tenderán a volcar la plataforma.

Para el mástil se pensó en utilizar un listón de madera lo suficientemente alto como para albergar en su interior una varilla telescópica, de al menos, unos 60 cm. Esta altura además, permitiría al quadrotor moverse libremente sin limitaciones físicas (excepto la de la propia rótula) y contribuiría a la comodidad a la hora de manipular el sistema.

En el mástil, se practicaría una perforación vertical profunda, acompañada de una ranura pasante para materializar el alojamiento del tubo que se deslizará permitiendo las subidas y bajadas del quadrotor.

6.2.2. Tubo de carbono

Para el elemento móvil que dará lugar a la elevación del dron, se ha optado por emplear un tubo de carbono. Este tubo, aparte de ser ligero y muy resistente, también permite una pequeña flexión, con lo que se le permitiría al sistema esbozar pequeños desplazamientos laterales, dando lugar a una simulación de vuelo más real.

6.2.3. La rótula

Con el fin de lograr una unión entre el quadrotor y la base de pruebas que no impida ninguno de los tres giros posibles se buscó, entre distintos catálogos y fabricantes, una rótula esférica que permitiese cualquier tipo de inclinación con márgenes amplios. Finalmente, se encontró la Igubal GFSM-08-IG (Ilustración 44).



Ilustración 44. Igubal GFSM-08-IG, extraída de www.igus.eu.

Esta rótula presenta unas características idóneas para el desarrollo de este proyecto. Soporta esfuerzos axiales de tracción de 150N (hasta 250N en situaciones puntuales de corta duración) y de compresión de 375N (hasta 750N en instantes puntuales de corta duración); siendo estos valores sobradamente superiores a los que se tendrán en los ensayos. Por otra parte, su libertad de giro en cualquier plano perpendicular al normal va desde los -34° hasta los 34° ¹³; margen más que suficiente ya que en un funcionamiento normal, el quadrotor no debería exceder los 20° de inclinación.

¹³ Experimentalmente se ha comprobado que el giro puede llegar hasta los 45° en ambos sentidos, aunque es posible que giros tan grandes, con el tiempo, sean perjudiciales para la rótula.

7. Diseño

Para modelizar, tanto el quadrotor como la base de pruebas, y estudiar la configuración óptima, así como sus dimensiones y materiales necesarios, se ha empleado el software de CAD 3D y diseño mecánico Autodesk® Inventor® 2014 en su versión para estudiantes.

7.1. Diseño del quadrotor

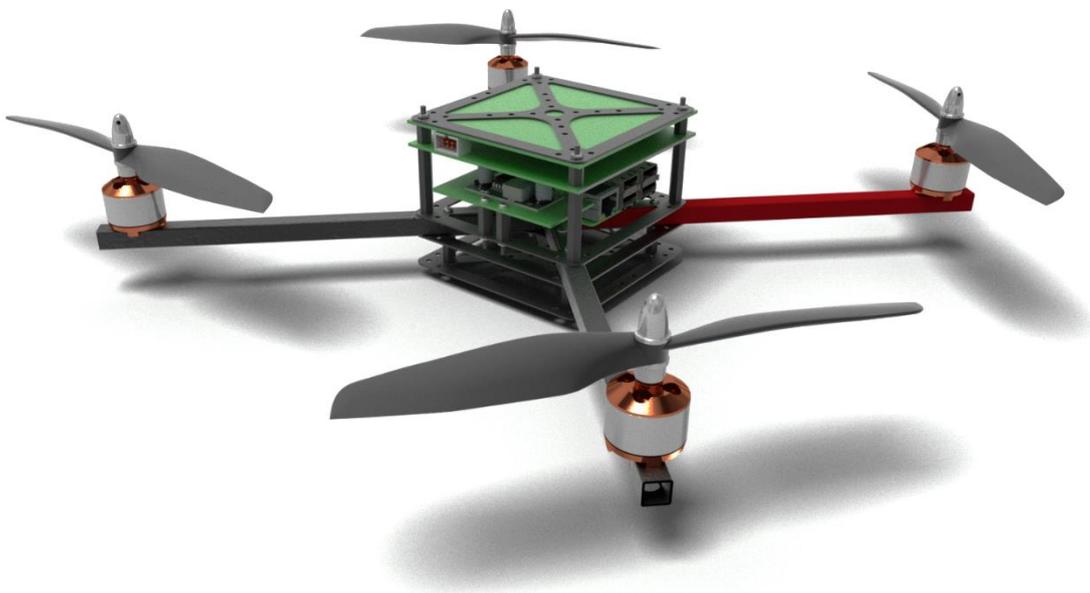


Ilustración 46. Diseño 3D del quadrotor sin carcasa.



Ilustración 45. Imagen del diseño 3D de la torre de placas.



Ilustración 47. Diseño 3D del quadrotor con carcasa.

7.2. Diseño de la base de pruebas completa



Ilustración 48. Diseño 3D de la plataforma de pruebas completa.



Ilustración 49. Imagen del diseño 3D de la unión entre el quadrotor y la base.

7.3. Modelado de componentes

Algunos componentes utilizados para el diseño 3D han sido extraídos de la comunidad web GRABCAD COMMUNITY. Para más información, acudir al *Anexo I*.

8. Software

Toda la parte de hardware descrita hasta ahora necesita de un software que gestione y controle la información, con el fin de transformar todo el conjunto de elementos en un sistema integrado y autónomo. Todo el desarrollo del software, desde la estructuración de los hilos de control hasta la generación, lectura y tratamiento de los datos, ha sido llevado a cabo por Rui Pedro Machado da Silva¹⁴, con quien se ha realizado conjuntamente el presente proyecto.

A continuación se esbozan brevemente algunos de los puntos más importantes del software para el correcto funcionamiento del sistema que se está llevando a cabo en este trabajo.

8.1. Sistema Operativo en Tiempo Real

El sistema operativo que trabaje en la CPU se encargará de llevar a cabo todas las operaciones programadas en el algoritmo de control, gestionando la ejecución de las distintas tareas en cada momento. Es por esto por lo que se hace necesario un sistema operativo en tiempo real. En el algoritmo se implementan diferentes funciones, y cada una de ellas debe ser ejecutada con unas frecuencias determinadas. El SOTR nos permite ajustar estas frecuencias de forma independiente para cada función, cumpliendo así con los períodos de ejecución requeridos en cada situación. Pero además, al ser los tiempos de ejecución distintos, es posible que dos tareas requieran ser atendidas en el mismo instante. Este tipo de SO incluye la posibilidad de dotar de diferentes niveles de prioridad a cada tarea, de modo que, en caso de presentarse la situación comentada, se llevaría a cabo la tarea de mayor prioridad. Pero además, dispone de una funcionalidad de semáforo; esto es que, si el procesador está ejecutando una operación de prioridad alta en el momento en el que solicita atención una tarea de menor prioridad, el sistema dejará en espera a esta última hasta que termine con la que está en curso. Por el contrario, si fuese una tarea de mayor prioridad la que requiriese atención mientras se procesa otra de menor interés, se bloquearía la ejecución de esta y se atendería a la función entrante de mayor prioridad; al finalizar, el sistema retomará la tarea que ha congelado.

En nuestro caso, se ha instalado el sistema operativo Raspbian junto con el parche Preempt_RT, que habilita las características de un SOTR.

¹⁴ Se dispone de información mucho más detallada sobre el software en el TFG realizado por Rui Pedro Machado da Silva, "Diseño de una plataforma software para el control de orientación 3D de un sistema quadrotor basado en Raspberry Pi 2".

8.2. Comunicación con el quadrotor (HMI)

Para realizar vuelos y pruebas es necesario tener acceso a ciertos parámetros del algoritmo que alberga la Raspberry. De este modo, se podría iniciar y parar el control del sistema, así como variar las constantes del PID o manipular las referencias que le llegan al quadrotor. Para realizar esta comunicación con el programa, se ha utilizado la interfaz hombre-máquina (HMI)¹⁵ que permite el envío de datos a través de una red wifi.

Esta interfaz permite iniciar y parar el control del vuelo, cambiar las constantes de los PID que controlan los seis grados de libertad espaciales, resetear la IMU (poniendo sus lecturas a cero en la posición actual del quadrotor), aumentar y disminuir la potencia de los motores (tanto individual como conjuntamente) y cambiar las referencias de los ángulos y la posición. También está preparada para mostrar las lecturas de la IMU en tiempo real.

En la siguiente imagen se puede ver el aspecto del HMI.

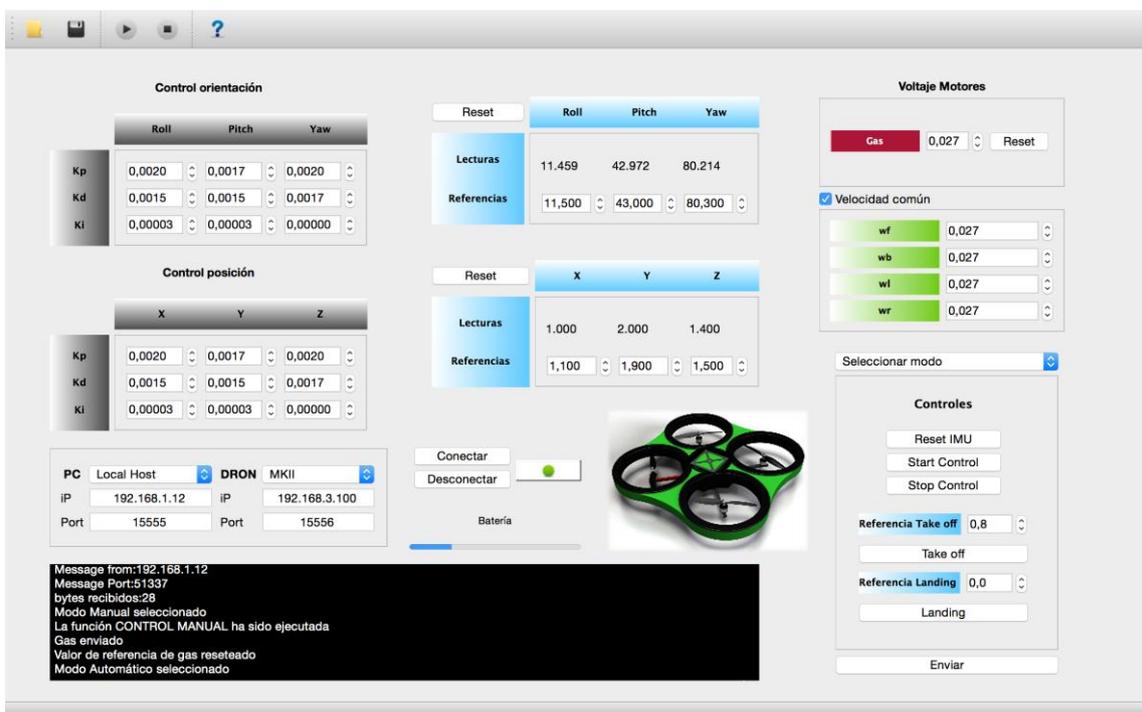


Ilustración 50. Aspecto gráfico de la interfaz HMI usada en el trabajo.

¹⁵ La interfaz HMI fue desarrollada por Natalia Díaz-Otero Zafrilla para su TFG “Desarrollo de un sistema de telemetría e interfaz gráfica para el control de un vehículo aéreo pilotado remotamente (RPAS)”.

8.3. Algoritmo de control

Para realizar el control se va a utilizar un controlador PID en paralelo, debido a su extremada sencillez de implementación y a su buena respuesta alrededor del punto de funcionamiento habitual; respecto del cual, no se van a producir variaciones muy grandes¹⁶.

8.3.1. Control mediante PID

El PID es un tipo de controlador basado en la realimentación negativa de la salida del sistema o proceso en bucle cerrado. La lectura de la variable a la salida del sistema es comparada con la referencia que se aplica, obteniendo un error. El error es procesado con tres operaciones que se superpondrán para dar lugar a una acción correctora que tenderá a hacer que la salida se aproxime al valor de referencia. Las tres operaciones son:

P. Una acción proporcional que solo interpreta la situación actual (presente). Permite aumentar la intensidad de la respuesta, pero puede llegar a causar inestabilidades en el sistema si el valor de la constante es demasiado elevado.

I. Una acción integral que tiene en cuenta la historia de las situaciones previas (pasado). Permite anular el error, aunque en sistemas con puntos de equilibrio inestables provoca un rápido descontrol del proceso.

D. Una acción derivativa que prevé la tendencia en situaciones venideras (futuro). Permite aumentar la rapidez con la que el sistema se estabiliza, pero si el valor de la constante es demasiado grande, puede desestabilizar debido a las vibraciones o al ruido de las lecturas.

El esquema general del bucle cerrado de un proceso controlado mediante PID resulta como el que se muestra a continuación:

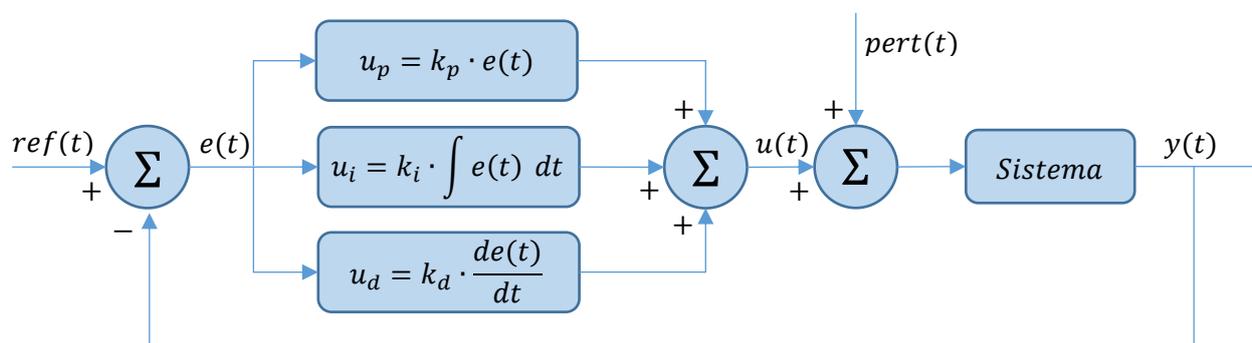


Ilustración 51. Esquema general de un bucle de control mediante PID.

¹⁶ El punto normal de funcionamiento se encontrará con los ángulos de Euler a 0° y, tal como se señaló en el apartado 3.3.2, no se espera que estos varíen más allá de $\pm 20^\circ$.

8.3.2. Esquema del PID implementado

El bucle que finalmente se ha implementado presenta una pequeña variación con respecto al esquema anterior. La acción derivativa se calculará con la velocidad del giro, en lugar de con la derivada del error. Esta modificación facilita el cálculo y no altera los resultados, ya que la variable de referencia va a permanecer constante durante la mayor parte del tiempo (excepto en los momentos en los que se varíe su valor mediante escalones). El esquema quedaría:

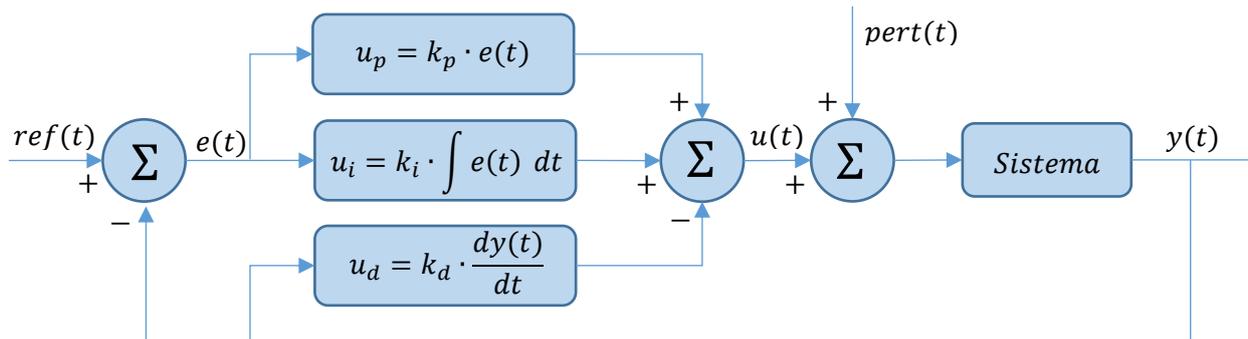


Ilustración 52. Esquema implementado del bucle de control mediante PID.

Las variables del bucle se identifican como:

$y(t)$: Orientación real del quadrotor medida por la IMU.

$ref(t)$: Referencia de los ángulos introducidos por el usuario a través del HMI.

$e(t)$: Diferencia entre la referencia y la lectura de la IMU.

$u_p(t)$, $u_i(t)$ y $u_d(t)$: Acción proporcional, integral y derivativa como respuesta al error.

$u(t)$: Acción enviada a los motores, resultante de la superposición de las acciones anteriores.

$pert(t)$: Perturbación física aplicada sobre el quadrotor.

Debido a la imposibilidad de trabajar con variables continuas, las variables tomarán valores discretos y se resolverán las ecuaciones de forma numérica. Teniendo todo esto en cuenta, el algoritmo quedaría así:

$$\begin{aligned}
 e_k &= ref_k - y_k ; \\
 u_{p_k} &= k_p \cdot e_k ; \\
 u_{i_k} &= u_{i_{k-1}} + k_i \cdot e_k ; \\
 u_{d_k} &= k_d \cdot y_k ; \\
 u_k &= u_{p_k} + u_{i_k} - u_{d_k} ;
 \end{aligned}$$

8.3.3. Obtención de los parámetros del PID

Una vez se tiene planteado el algoritmo, solo falta caracterizarlo para que genere el efecto deseado en nuestro sistema. Para ello, es necesario identificar cuáles van a ser los valores de las constantes k_p , k_i y k_d de nuestro PID. Es importante señalar ahora que, para simplificar este proceso, se van a diseñar cuatro bucles independientes para cada variable a controlar. Es decir, se va a ejecutar un PID para cada uno de los tres ángulos y un cuarto para la altura¹⁷. El algoritmo será idéntico para cada variable, tan solo variarán las constantes del control. Si no se hiciese esto, se tendría un sistema mucho más complejo y difícil de controlar, pero cuyo planteamiento no presentaría ninguna ventaja apreciable respecto al que aquí se propone.

Para identificar un PID que cumpla las especificaciones requeridas existen, al menos, cuatro formas posibles de abordar el problema. La primera de ellas es la ideal para el control de cualquier proceso y consiste en disponer de un modelo analítico del sistema que nos permita realizar un estudio matemático del bucle cerrado; pudiendo ajustar los parámetros del PID para que el lugar de las raíces (en el dominio de Laplace) se encuentre dentro de la zona de especificaciones. Sin embargo, debido a la complejidad aerodinámica del quadrotor, realizar un modelo del sistema a partir de ecuaciones matemáticas supondría una inversión de tiempo y esfuerzo que sobrepasa el alcance de este trabajo; además, la existencia de otros métodos más rápidos y sencillos propician el descarte de éste.

Por otra parte, y para atajar el problema del método anterior, podría identificarse un modelo empírico a partir de ensayos realizados con la plataforma desarrollada; estudiando el comportamiento del quadrotor al introducir cambios en las referencias. Sin embargo, el sistema con el que se está trabajando presenta un punto de equilibrio inestable alrededor del cual se pretende realizar el modelo, y por lo tanto, resulta imposible realizar ensayos que pudieran aportar datos útiles.

La imposibilidad de establecer una representación numérica del sistema, fuerza a obtener directamente los valores del PID por otros medios no tan exactos, aunque suficientemente satisfactorios. La primera opción que se presenta ahora sería la de ensayar el sistema para caracterizar empíricamente el control a partir de métodos como los de Ziegler-Nichols, CHR o Cohen-Coon. Pero vuelve a surgir el problema mencionado anteriormente, pues se requiere de un estudio sobre el comportamiento del quadrotor funcionando en modos imposibles de alcanzar con un sistema inestable como el nuestro. Por lo tanto, solo queda la opción de un ajuste empírico directo del PID.

Con el uso de la plataforma de ensayos y siguiendo unas pautas y precauciones, se procede a estudiar el comportamiento del sistema para distintas combinaciones de las constantes del controlador. De esta manera, observando las carencias y excesos, se puede ajustar el PID para que cumpla satisfactoriamente los requerimientos impuestos.

¹⁷ En el *Anexo II* se muestra el código donde queda implementado el control, pudiéndose apreciar los distintos PID para cada variable.

8.3.4. Necesidad de un SOTR

El control en un sistema se plantea para mantener unas determinadas condiciones. En nuestro caso se pretenden mantener la orientación y la altura en sus respectivos valores de referencia. Para ello se implementa el algoritmo de control. Ahora bien, la CPU se encarga de llevar a cabo esta tarea; pero junto a ella, debe ejecutar otra serie de funciones, como la lectura de los sensores, la adecuación de los datos, el envío de las acciones a los drivers, el intercambio de información con el HMI, etc. Debido a que el control es crítico, es necesario establecer un periodo de ejecución adecuado, capaz de cumplirse siempre. Por eso resulta indispensable el uso de un SOTR que sea capaz de priorizar el hilo de control por encima de otras tareas, con el fin de mantener una frecuencia adecuada e inalterable que proporcione una respuesta rápida y sin retraso.

Para el hilo de control se establece la mayor frecuencia de ejecución, con la tercera mayor prioridad. La primera y segunda posición en cuanto a prioridades las ocupan la lectura de la IMU y el tratamiento de los datos leídos, respectivamente. Esto es así porque para realizar el control se necesita conocer los valores de la salida del sistema en ese instante. De nada sirve realizar una acción de control de un estado que no sea el actual.

9. Ensayos

Durante los ensayos llevados a cabo en la base de pruebas, diseñada y desarrollada a lo largo del proyecto, se han podido ajustar los parámetros de todos los PID destinados al control de orientación; además de comprobar si la respuesta del quadrotor ante cambios en las referencias y ante perturbaciones era la adecuada.

En este punto conviene señalar que, a pesar de haber implementado en la base de pruebas un sistema que permite testear el control de altura de un quadrotor y de haber habilitado en el algoritmo la lectura y procesado de los datos de esta variable, no se ha llegado a ajustar un control de altura, debido a los errores en las lecturas¹⁸.

9.1. Control del Roll

9.1.1. Respuesta a cambios de referencia en Roll

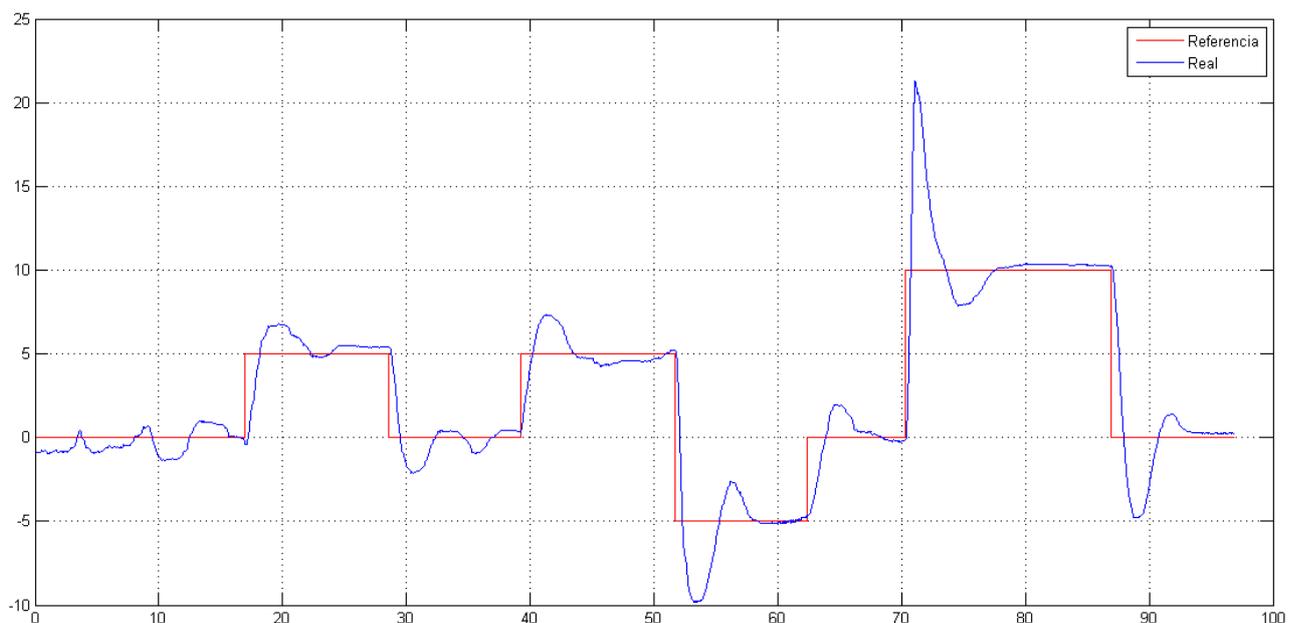


Ilustración 53. Respuesta ante cambios de referencia en el ángulo de Roll.

En la *Ilustración 53* se puede ver el seguimiento del quadrotor al cambio de referencia. Si bien la respuesta es lenta y subamortiguada el error final es prácticamente nulo. También se aprecia que el comportamiento es similar al producirse cambios de inclinación en ambos sentidos.

¹⁸ Se habla de estos errores en el apartado 4.2.8.

9.1.2. Respuesta ante perturbaciones en Roll

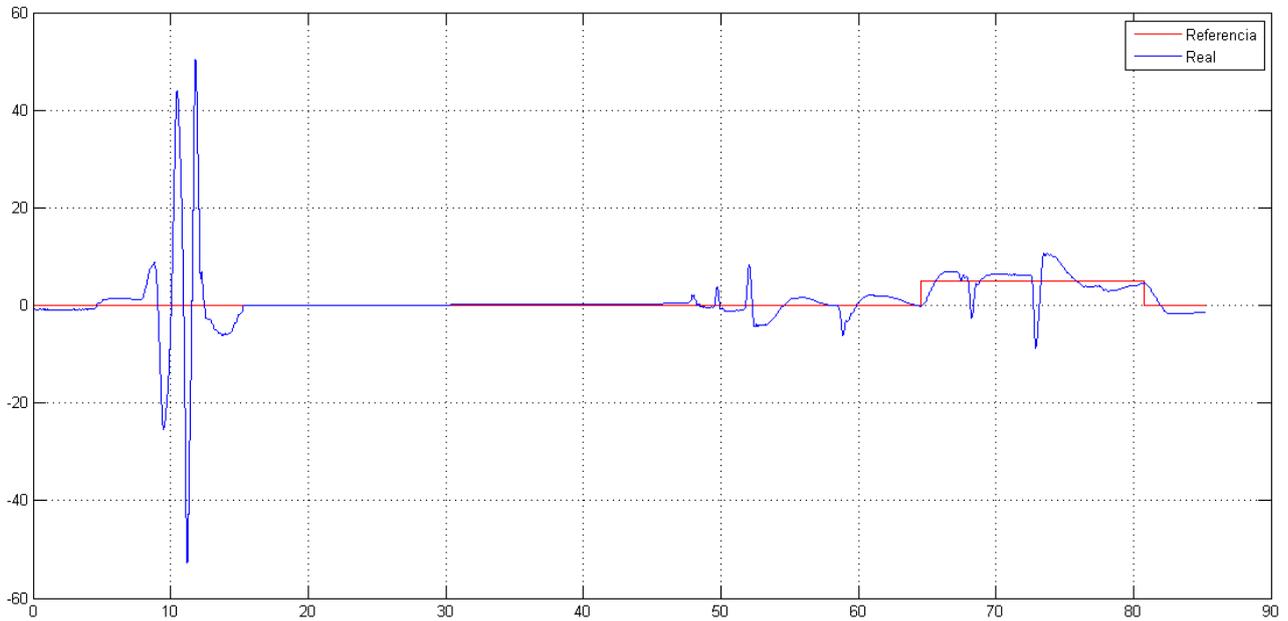


Ilustración 54. Respuesta ante perturbaciones y cambios de referencia en el ángulo de Roll.

En la primera parte del ensayo se puede apreciar la aparición de una fuerte inestabilidad al introducirse una perturbación de unos 10° .

A partir del segundo 45 se puede ver como la respuesta es buena para valores de las perturbaciones inferiores a 10° . Se puede apreciar como las sobreoscilaciones aumentan según aumenta la intensidad de las perturbaciones.

A pesar de todo, se puede aceptar que el comportamiento es adecuado, siempre y cuando las desestabilizaciones no superen un valor de 10° .

9.2. Control del Pitch

9.2.1. Respuesta a cambios de referencia en Pitch

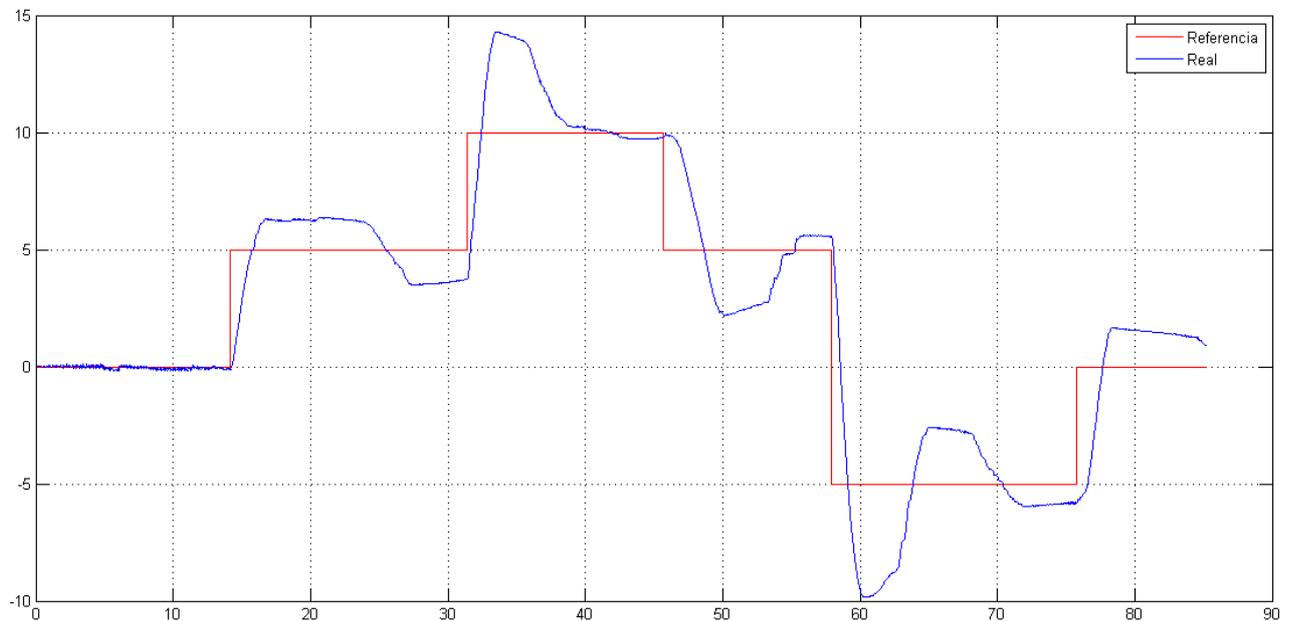


Ilustración 55. Respuesta ante cambios de referencia en el ángulo de Pitch.

A pesar de que el seguimiento de la referencia en el Pitch es adecuada y el sistema alcanza a estabilizar sin problemas, con un error aceptable, se aprecian ciertas dificultades de movimiento cuando el quadrotor debe cambiar el sentido de la velocidad de inclinación. Este comportamiento puede ser debido al desgaste de la rótula; ya que cuando se realizó este ensayo, se empezó a notar que el rozamiento entre las partes de la rótula comenzaba a ser excesivamente elevado. En una situación de vuelo real o con la unión en buenas condiciones, se espera que la respuesta sea igual a la que se observa en el Roll.

9.2.2. Respuesta ante perturbaciones en Pitch

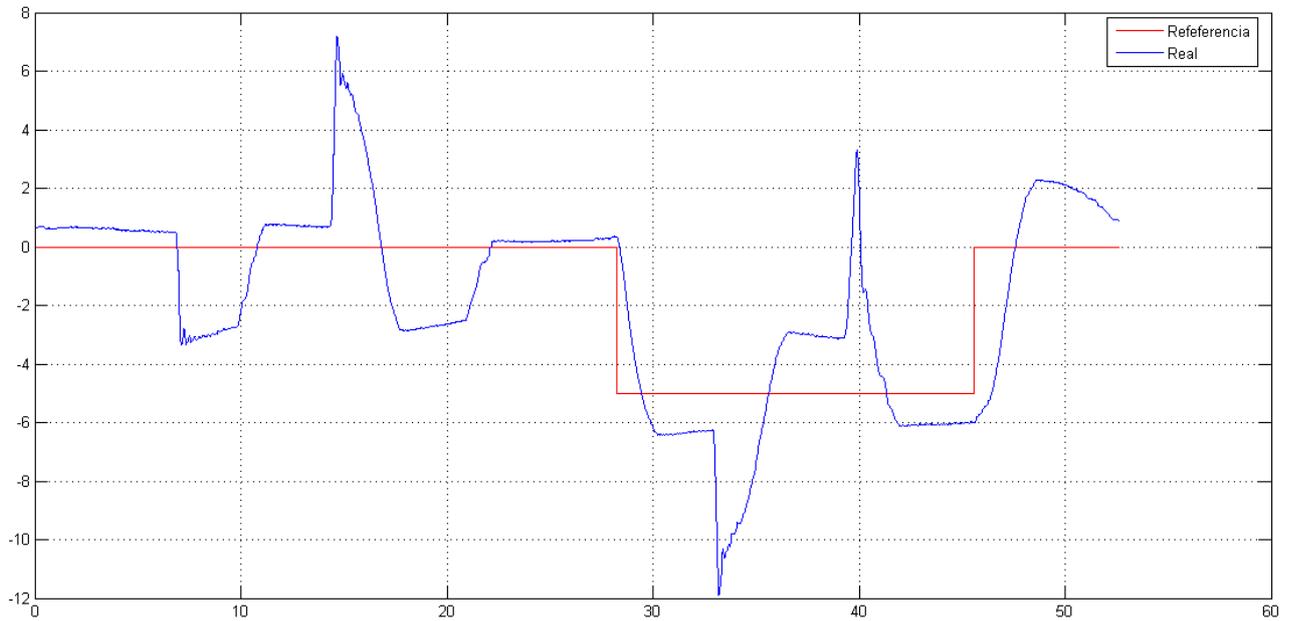


Ilustración 56. Respuesta ante perturbaciones y cambios de referencia en el ángulo de Pitch.

De nuevo aquí se vuelve a apreciar una dificultad en el movimiento al producirse cambios de sentido en la velocidad de inclinación. Sin embargo, y a pesar de ello, la respuesta es buena ante perturbaciones.

Los errores de posición que se aprecian a partir del segundo 30 pueden ser debidos a que el sistema se sometió a variaciones antes de que llegase a estabilizar del todo tras las perturbaciones previas.

9.3. Control del Yaw

9.3.1. Respuesta a cambios de referencia en Yaw

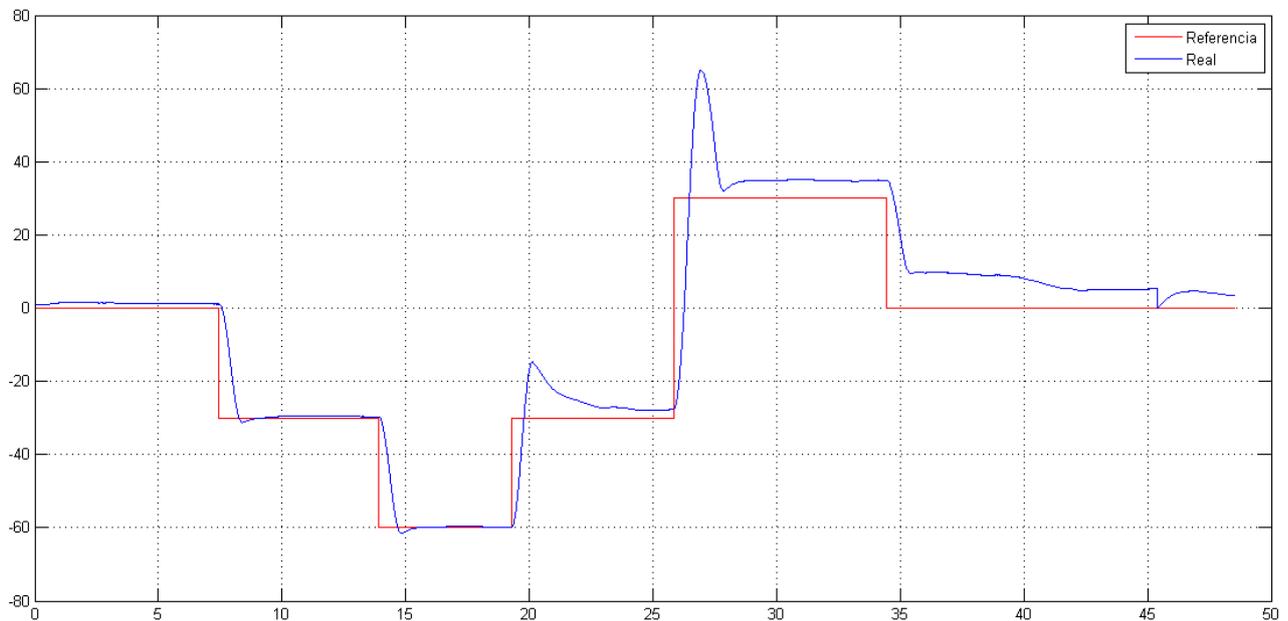


Ilustración 57. Respuesta ante cambios de referencia en el ángulo de Yaw.

En el control del seguimiento de una referencia para el Yaw se tiene una respuesta más que satisfactoria. Hasta el segundo 25 las respuestas son rápidas, con una sobreoscilación casi inapreciable y un error prácticamente nulo.

Alrededor del segundo 26 se produce un salto desde el ángulo -30° hasta el 30° , apareciendo una sobreoscilación considerable y un error que se mantiene en instantes posteriores.

Si se comparan los saltos sucedidos en los segundos 7 y 19, se puede notar cómo el control responde mejor al realizar giros negativos. Esto puede tener su explicación en las parejas de drivers/motores y habría que estudiar más detenidamente las prestaciones de cada motor por separado.

9.3.2. Respuesta ante perturbaciones en Yaw

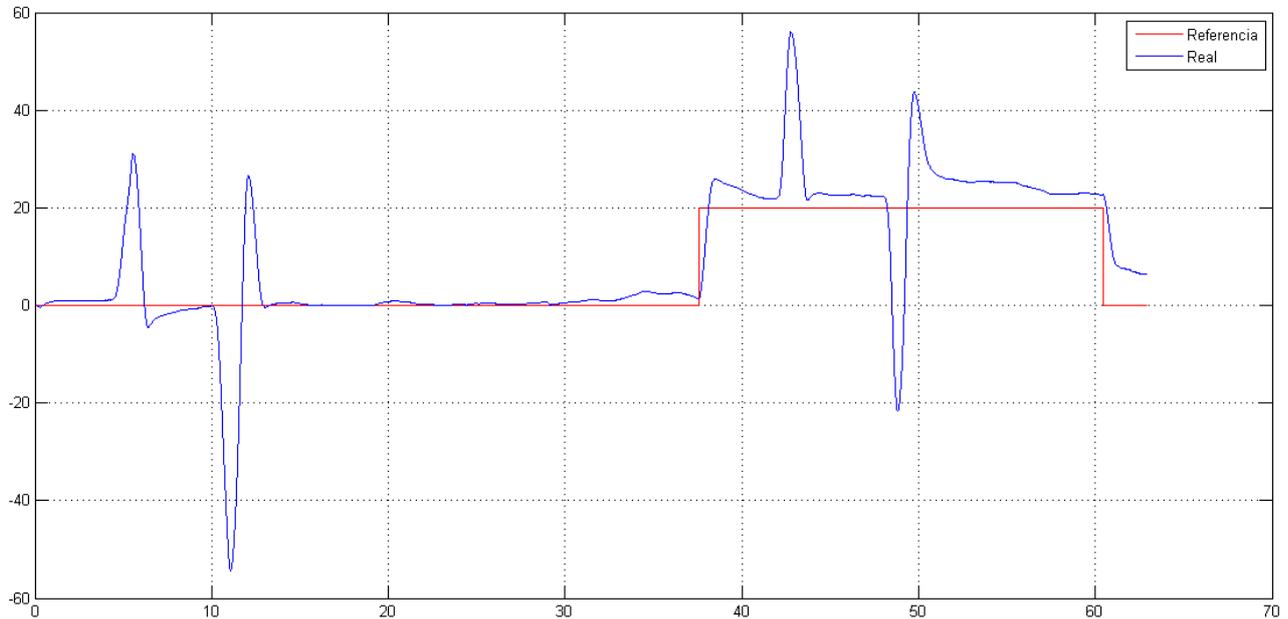


Ilustración 58. Respuesta ante perturbaciones y cambios de referencia en el ángulo de Yaw.

La respuesta ante perturbaciones no superiores a 30° a referencia 0° es bastante buena con sobreoscilación pequeña y error casi nulo. Sin embargo, al cambiar la referencia se acumula algo de error, pero la respuesta a las perturbaciones sigue siendo buena.

Con perturbaciones superiores a 30° , como las que se observan en los segundos 11 y 49, comienzan a aparecer respuestas subamortiguadas apreciables.

9.4. Observaciones

Si en los ensayos del Pitch se tiene en cuenta la influencia del mal estado de la rótula, y se supone que los resultados que se obtendrían en un ensayo adecuado se asemejarían a los obtenidos en los ensayos del Roll; se puede concluir que el control presenta un buen comportamiento para cambios en las referencias de Pitch y Roll no superiores a $\pm 5^\circ$ y en Yaw no superiores a $\pm 40^\circ$.

Respecto a las perturbaciones, la respuesta es buena mientras la referencia se mantenga a 0° (cumpliéndose para todos los ángulos) y el valor de las desviaciones no sea superior a unos 7° en Roll y Pitch y a unos 30° en Yaw.

Se espera que en vuelo real, libre de las vibraciones y limitaciones que impone la unión a una plataforma fija, la estabilidad y el control del quadrotor sean significativamente mayores.

10. Conclusiones y trabajos futuros

10.1. Conclusiones

Como objetivo de este trabajo se planteaba el desarrollo completo de una plataforma de implementación, testeo y ensayo de algoritmos destinados al control de orientación y altura de sistemas quadrotores gobernados por una CPU Raspberry Pi. Tras varios meses de trabajo, repletos de aciertos y contratiempos, se han conseguido completar todas las fases del proceso de desarrollo recogidos por este trabajo y que van desde el planteamiento del concepto de la base de pruebas, hasta el ajuste del último PID; pasando por el diseño de toda la plataforma, el estudio y selección de los materiales y componentes que lo forman, la adopción de la solución constructiva junto con la disposición, estructuración e interconexión física y digital de todos los elementos. Pero todo ello es únicamente la parte del hardware del proyecto; a esto hay que añadirle la componente del software desarrollada de forma paralela y conjunta por Rui Pedro Machado da Silva, y que contempla toda la estructuración del código; la gestión de los recursos del sistema operativo; la obtención, tratamiento, procesado y distribución de la información; la selección de puertos y registros de datos, etc.

Por todo lo citado en el párrafo anterior, se puede apreciar que se trata de un trabajo multidisciplinar, en el que han entrado en juego diversas ramas de la ingeniería. Electrónica, eléctrica, programación, control y mecánica de fluidos son algunos ejemplos de los campos ingenieriles que han formado parte del proyecto. Por otro lado, al haberse tratado de un trabajo realizado en conjunto ha sido posible explorar y mejorar habilidades imprescindibles para la vida profesional y el trabajo en equipo, como la coordinación y el reparto de las tareas, la organización, el respeto mutuo, la comunicación activa y precisa, la asunción de responsabilidades y la capacidad de autocrítica, entre otras. A parte de haber permitido potenciar competencias más individuales, como la habilidad de afrontar problemas nuevos y distintos proponiendo soluciones efectivas, diferentes e innovadoras.

Con respecto al trabajo en sí, éste venía motivado por el interés de probar la validez en sistemas quadrotores de nuevas CPU más pequeñas, potentes y baratas que las que se habían estado usando hasta el momento. Además, la incomodidad e imposibilidad de realizar vuelos en exteriores, debido a las normativas que regulan el uso de este tipo de dispositivos, propició la propuesta de una base de pruebas. Ambas ideas se conjugaron para dar lugar a la plataforma que se ha desarrollado. Pero resulta interesante recordar la verdadera motivación que llevó al planteamiento del diseño de una base de pruebas. Existen en el mercado (aunque no se producen en masa) algunas bases semejantes a la llevada a cabo; pero la opción de adquirir una de ellas quedó descartada debido a que su precio se elevaba por encima de los veinte mil euros. Tras la finalización del trabajo, se ha podido comprobar que el precio de estos aparatos queda justificada. Como se puede ver en el presupuesto base de licitación recogido al final del Presupuesto¹⁹, el precio total de nuestro proyecto asciende a 22.737,57€.

¹⁹ En este escrito, el Presupuesto es el documento que sigue a la Memoria.

que es incluso superior al del 3DOF Hover de Quanser; si bien es cierto que en nuestro caso se cargan gastos fijos como los de diseño e implementación del software en una única unidad.

En lo referente a los resultados, se puede decir que han sido, por lo general, bastante satisfactorios; salvo por pequeños matices, como el rápido deterioro de la rótula, que ha influenciado significativamente en los ensayos, o la ausencia de un sensor de altura preciso que permitiese el control de este parámetro. Aun así, se ha logrado mantener una orientación estable y con buena respuesta a cambios de referencia y perturbaciones, que era el objetivo primordial que se perseguía.

Por último, se podría hablar de la trascendencia de este proyecto y de cómo afectará a futuros trabajos. La disposición de una base de pruebas que permita la calibración de los componentes, el ajuste de los parámetros y los ensayos de funcionamiento, puede facilitar y acelerar enormemente el desarrollo de sistemas quadrotores auto-controlados que pudieran proponerse en años posteriores. De esta manera, se podría invertir más esfuerzo y tiempo en mejorar los algoritmos o en implementar nuevas funciones, que dieran lugar a dispositivos mucho más complejos y completos.

10.2. Trabajos futuros

Tras la ejecución y conclusión de este trabajo, se plantean diversas ideas y propuestas interesantes para llevar a cabo en futuros proyectos. Algunas de ellas son:

- Inclusión de un sensor de ultrasonidos para poder realizar un control de altura en interiores con una fiabilidad aceptable.
- Implementación del control de posición respecto a los ejes 'x' e 'y' con ayuda de un dispositivo GPS para exteriores y/o un dispositivo láser para interiores.
- Mejorar la unión entre la base de pruebas y el quadrotor, bien buscando una nueva rótula más fiable, o bien diseñando un nuevo sistema más efectivo.
- Disponer, sobre la base de pruebas, sensores que permitan realizar mediciones precisas de los ángulos y la altura del quadrotor en ensayo, de forma que se pueda comprobar también que las lecturas de los sensores del dron son correctas.
- Sustituir la Raspberry Pi 2 utilizada en este trabajo por una Raspberry Pi 3.

Bibliografía

The History of Drones and Quadcopters. (s.f.). Quadcopter Arena. Extraído de <http://quadcopterarena.com/the-history-of-drones-and-quadcopters/>.

Drones: a history of flying robots. (s.f.). Nesta. Extraído de <http://www.nesta.org.uk/drones-history-flying-robots/>.

A Brief History of Drones. (2012). The Nation. Extraído de <http://www.thenation.com/article/brief-history-drones/>.

Real Decreto-ley 8/2014, de 4 de julio, de aprobación de medidas urgentes para el crecimiento, la competitividad y la eficiencia. (2014). Agencia Estatal Boletín Oficial del Estado [ES]. Extraído de https://www.boe.es/diario_boe/txt.php?id=BOE-A-2014-7064.

Luukkonen, Teppo. (2011). Modeling and control of quadcopter. Aalto University.

Serrano Iribarnegaray, L; Martínez Román, J.A. (2013) [2ª ed]. Máquinas eléctricas. Valencia: Editorial Universitat Politècnica de València.

Hélices (Teoría de hélices). (s.f.). Mundo aeronáutico EWM. Extraído de <http://www.alasewm.com.ar/helicewm.htm>.

Raspberry Pi 2 Model B. (s.f.) Raspberry Pi Foundation. Extraído de <https://www.raspberrypi.org/>.

How does a brushless electric motor work? (2006) How Stuff Work Tech. Extraído de <http://electronics.howstuffworks.com/brushless-motor.htm>.

Motor Brushless (2008) Quadruino.COM. Extraído de <http://www.quadruino.com/guia-2/materiales-necesarios-1/motores-brushless>.

ESC's (2008) Quadruino.COM. Extraído de <http://www.quadruino.com/guia-2/materiales-necesarios-1/esc>.

Raspberry Pi to Arduino Shields Connection Bridge. (s.f.) Cooking Hacks. Extraído de <https://www.cooking-hacks.com/documentation/tutorials/raspberry-pi-to-arduino-shields-connection-bridge/>.

Inertial Measurement Unit (IMU) (1998) Space System Laboratory (University of Meryland). Extraído de <http://www.ssl.umd.edu/projects/RangerNBV/thesis/2-4-1.htm>.

Riu, J.; Boqué, R.; Maroto, A.; Rius, F. X.; Laso, J. (2006). Calibración de equipos de medida. Tarragona. Universitat Rovira i Virgili.

Ors-Carot, R.; Martí-Campoy, A.; Pérez-Jiménez, A. J.; Gracia-Morán, J.; Saiz-Adalid L. J. (2012). Apuntes de informática industrial. Valencia: Editorial Universitat Politècnica de València.

Comunicación – Bus I2C. (2007) Robots. Extraído de http://robots-argentina.com.ar/Comunicacion_busI2C.htm.

Configuración I2C. (2012) Adafruit. Extraído de <https://learn.adafruit.com/adafruit-raspberry-pi-lesson-4-gpio-setup/configuring-i2c>.

The User Datagram Protocol (2008) Electronics Research Group. Extraído de <http://www.erg.abdn.ac.uk/users/gorry/course/inet-pages/udp.html>.

Intro to Real-Time Linux for Embedded Developers (2013) LINUX.COM. Extraído de <https://www.linux.com/blog/intro-real-time-linux-embedded-developers>.

Hägglund, Tore; Åström, Karl J. (2009). Control PID avanzado. Madrid: Pearson Prentice Hall.

3 DOF Hover. (s.f.). Quanser Innovate Educate. Extraído de http://www.quanser.com/products/3dof_hover.

Anexos

Anexo I

Para la modelización CAD 3D de componentes electrónicos complejos se ha acudido a la comunidad de ingenieros GRABCAD. Esta comunidad, accesible en la red, permite que diseñadores y desarrolladores compartan sus archivos y modelados, con el fin de ofrecer ayuda, de forma totalmente desinteresada, a los compañeros de profesión y a todo aquel que pueda tener interés en hacer uso de esos archivos.

Para el modelado de la Raspberry Pi se ha utilizado el CAD “Raspberry Pi 2 model B, 3D CAD assembly model”, publicado el 21 de mayo de 2015 por Robèrt Akòpov. (Ilustración 59).

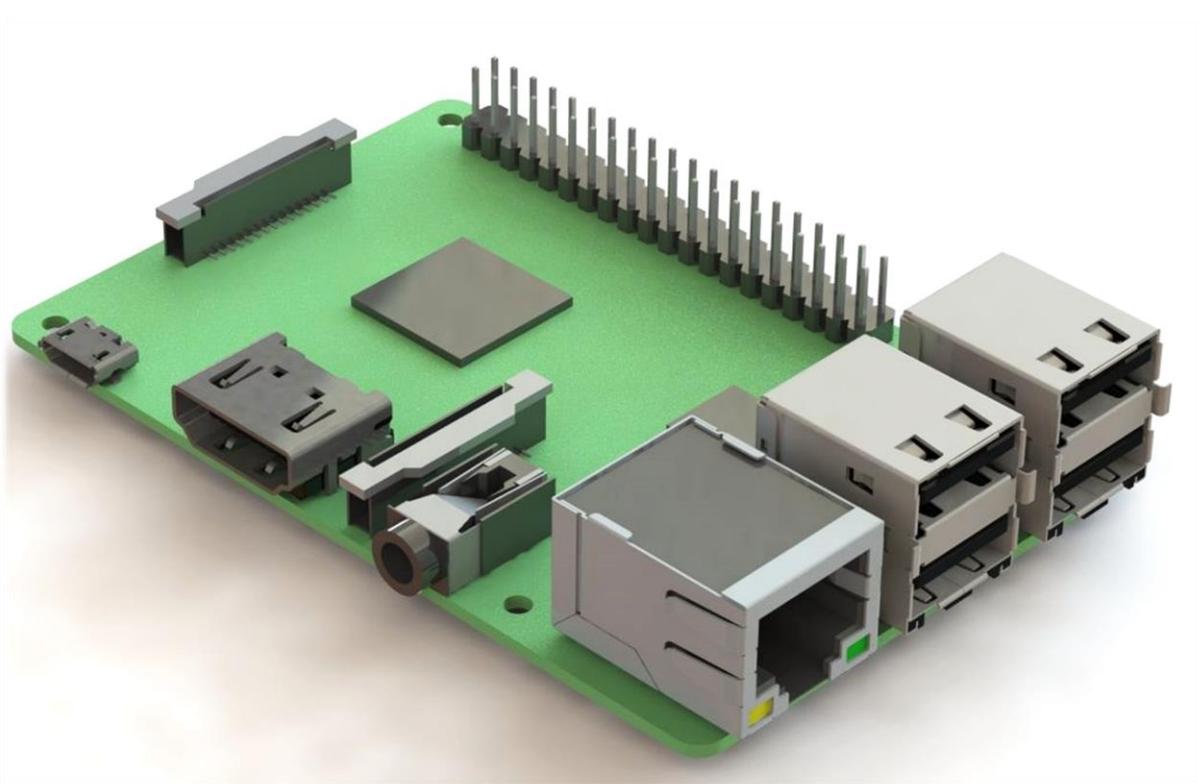


Ilustración 59. Modelo CAD de la Raspberry Pi 2 Model B.

Para el modelado del convertidor de tensiones se ha empleado el archivo CAD "lm2596 stepdown", publicado el 1 de enero de 2015 por Ray Brok. (Ilustración 60).

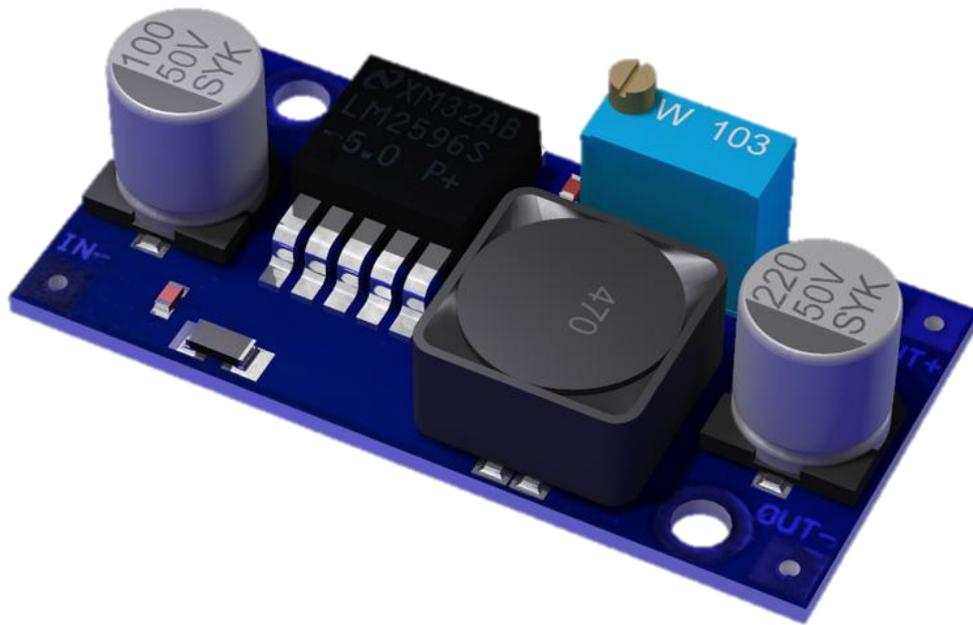


Ilustración 60. Modelado CAD del conversor de tensiones LM2596.

Anexo II

Para los controles de orientación y altura se han implementado cuatro PID, uno para cada variable. A partir de las lecturas de la IMU se procesan los valores de Roll, Pitch, Yaw y altura, obteniéndose valores de acciones independientes. Después, estas acciones se combinan para proporcionar a cada motor la potencia que debe desarrollar con el fin de reestablecer o alcanzar las referencias de cada variable.

```
void RaspberryPi::ReadyState(void) {

    float refmotor1,refmotor2,refmotor3,refmotor4;

    GetMeasures();

    t += Ts;

    if (fabs(roll)>600.0 || fabs(pitch)>600.0 ){
        motors->on=0;
        StopControl();
        printf("STOP - Angulos de orientacion demasiado elevados\n");
    }

    if(On){

        switch (iSubstate) {

            case RASPBERRY_PI_IDLE:

                motors->on = false;
                break;

            case RASPBERRY_PI_REMOTE_CONTROL:

                motors->on = true;
                motors->mode = CONTROL_MODE_AUTO;

                ///! CONTROL PID RASPBERRY Configurable desde HMI

                wzp = Saturation(xpaltura*(refaltura-altitude),0.15,-0.15); //Antes 0.08
                wzd = Saturation(kdaltura*(-valtitude),0.06,-0.06);
                wzi = Saturation((wzi+kialtura*0.001*(refaltura-altitude)), kiy, -kiy);

                wz = wzd+wzp+wzi;

                wpp = Saturation(kppitch*(refpitch-pitch), 0.2, -0.2);
                wpd = Saturation(kdpitch*(-vpitch), 0.15, -0.15);
                wpi = Saturation(wpi+(refpitch-pitch)*kipitch, 0.1, -0.1);

                wp = (wpd+wpp+wpi);

                wrp = Saturation(xproll*(refroll-roll), 0.2, -0.2);
                wrd = Saturation(kdroll*(-vroll), 0.15, -0.15);
                wri = Saturation(wri+(refroll-roll)*kiroll, 0.1, -0.1);

                wr = (wrd+wrp+wri);

                wyp = Saturation(kpyaw*(refyaw-yaw),0.1, -0.1);
                wyd = Saturation(kdyaw*(-vyaw), 0.075, -0.075);
                wyi = Saturation(wyi+(refyaw-yaw)*kiyaw, 0.05, -0.05);

                wy = wyd+wyp+wyi;


```

Obtención de las acciones de control independientes para cada variable.

```

//!SUMA MOTORES
refmotor1 = Saturation(umotor+wr-wp+wy+wz, 1, 0);
refmotor2 = Saturation(umotor+wr+wp-wy+wz, 1, 0);
refmotor3 = Saturation(umotor-wr+wp+wy+wz, 1, 0);
refmotor4 = Saturation(umotor-wr-wp-wy+wz, 1, 0);

if (archivo_raspberrypi_Abierto){
    fprintf(archivo_raspberrypi, "%f %f %f %f\n", t, refmotor1, refmotor2, refmotor3, refmotor4);
}

motors->SetMotors(refmotor1,refmotor2,refmotor3,refmotor4); // Pas a xxxxxx

break;

case RASPBERRY_PI_MOTOR_REF:

    motors->on = true;
    motors->mode = CONTROL_MODE_MANUAL;

    motors->SetControlActions(wfrd, whck, wift, wrgt);

    break;

}

}

void RaspberryPi::ShutDownState(void) {

    StopQuad();
    fclose(archivo_raspberrypi);

}

void RaspberryPi::StartQuad(void){

    sleep(5);

    sleep(5);

    Filter->Setup();
    motors->Setup();

    Filter->Start();
    sleep(2);
    motors->Start();

    sleep(5);

    archivo_raspberrypi = fopen("/home/pi/programs/RaspberryPi.txt", "w");
    archivo_raspberrypi_Abierto = true;

    t=0;

    On = true;

    StartControl(RASPBERRY_PI_IDLE);

}

```

Combinación de las acciones de control para cada motor acorde a su distribución.

Como se puede ver en el código, la asignación de los valores de las acciones no es directa, si no que se pasa por un filtro de saturación. De modo que en la asignación de acciones del PID de cada variable se aplican unos valores máximos y mínimos que se le permite adquirir a cada parámetro. Por ejemplo, para la acción proporcional del Pitch, tan solo se permiten valores entre -0.2 y 0.2. De esta forma se evita que la respuesta del controlador sea demasiado brusca, ya que, al tratarse de un sistema de naturaleza inestable, cualquier exceso en alguna acción de control podría desestabilizar al dispositivo. También se satura la acción que finalmente será enviada a cada motor, de este modo, aunque el control lo requiera, no se someterá a los motores a sobrecargas que pudieran deteriorar los devanados o los componentes mecánicos

PRESUPUESTO

Índice del presupuesto

1. Introducción	3
2. Cuadro de precios unitarios	4
2.1. Capítulo 1: Diseño	4
2.2. Capítulo 2: Construcción	4
2.3. Capítulo 3: Software.....	4
2.4. Capítulo 4: Ensayos	4
3. Cuadro de precios descompuestos	5
3.1. Capítulo 1: Diseño	5
3.2. Capítulo 2: Construcción	5
3.3. Capítulo 3: Software.....	8
3.4. Capítulo 4: Ensayos	8
4. Presupuesto parcial	9
4.1. Capítulo 1: Diseño	9
4.2. Capítulo 2: Construcción	9
4.3. Capítulo 3: Software.....	9
4.4. Capítulo 4: Ensayos	9
5. Presupuesto total	10

1. Introducción

El contenido de este documento se centra en exponer el coste económico que supone la ejecución del trabajo al cual pertenece: “Desarrollo de una plataforma de control basada en Raspberry Pi para la implementación y ensayo del control de orientación mediante soporte de pruebas indoor”; junto con el trabajo “Diseño de una plataforma software para el control de orientación 3D de un sistema quadrotor basado en Raspberry Pi 2” realizado por el alumno Rui Pedro Machado da Silva. El motivo de presentar un presupuesto común a ambos trabajos se debe a la inevitable relación entre ambos y a su continua convergencia en los diferentes aspectos tecnológicos, además de la estrecha colaboración entre las dos partes durante realización de todo el proyecto.

En este presupuesto se desglosa todo el desarrollo en cuatro capítulos principales, a saber: diseño, construcción, software y ensayos. Cada uno de ellos, a su vez, posee unidades de obra que también son estudiadas por separado:

Capítulo 1: Diseño

UO.1.01 Diseño del quadrotor: Concreción de las características del quadrotor, selección de los componentes acorde a las características y planteamiento de la distribución de cada elemento.

UO.1.02 Diseño de la base de pruebas: Diseño de la base de pruebas, selección de materiales y estudio del modo de ejecución del montaje.

Capítulo 2: Construcción

UO.2.01 Adecuación de conexiones electrónicas y eléctricas: realización de los circuitos y conexiones necesarias, tanto para la etapa de potencia como para la transferencia de datos.

UO.2.02 Montaje del quadrotor: ensamblaje de las piezas y elementos que conjugan la totalidad del quadrotor funcional.

UO.2.03 Adecuación de las piezas de la base de pruebas: adaptación y tratamiento de las partes que conforman la base de pruebas, para su posterior montaje.

UO.2.04 Construcción de la base de pruebas: unión de todas las piezas que conforman la base de pruebas.

Capítulo 3: Software

UO.3.01 Desarrollo del software: escritura del código necesario para gestionar las lecturas de los sensores y el procesado de los datos, con el fin implementar un algoritmo de control capaz de accionar los actuadores.

Capítulo 4: Ensayos

UO.4.01 Ensayos del quadrotor: pruebas y testeos del sistema global (hardware/software) para la comprobación de su correcto funcionamiento.

2. Cuadro de precios unitarios

2.1. Capítulo 1: Diseño

Código	Concepto	Precio
UO.1.01	Diseño del quadrotor	2.549,75 €
UO.1.02	Diseño de la base de pruebas	1.380,47 €

2.2. Capítulo 2: Construcción

Código	Concepto	Precio
UO.2.01	Adecuación de conexiones electrónicas y eléctricas	2.271,69 €
UO.2.02	Montaje del quadrotor	1.200,75 €
UO.2.03	Adecuación de las piezas de la base de pruebas	127,60 €
UO.2.04	Construcción de la base de pruebas	173,44 €

2.3. Capítulo 3: Software

Código	Concepto	Precio
UO.3.01	Desarrollo del software	5.814,24 €

2.4. Capítulo 4: Ensayos

Código	Concepto	Precio
UO.4.01	Ensayos del quadrotor	2.012,14 €

3. Cuadro de precios descompuestos

3.1. Capítulo 1: Diseño

UO.1.01 Diseño del quadrotor					
Código	Descripción	Unidades	Rendimiento	Precio	Importe
MO GITI	Graduado en Tecnologías Industriales	h	90	25	2250
MAQ OP	Ordenador portátil	h	75	3	225
	Costes directos complementarios	%	0,01	2475,00	24,75
	Costes indirectos	%	0,02	2499,75	50,00
	Precio total UO				2.549,75 €

UO.1.02 Diseño de la base de pruebas					
Código	Descripción	Unidades	Rendimiento	Precio	Importe
MO GITI	Graduado en Tecnologías Industriales	h	50	25	1250
MAQ OP	Ordenador portátil	h	30	3	90
	Costes directos complementarios	%	0,01	1340,00	13,40
	Costes indirectos	%	0,02	1353,40	27,07
	Precio total UO				1.380,47 €

3.2. Capítulo 2: Construcción

UO.2.01 Adecuación de conexiones electrónicas y eléctricas					
Código	Descripción	Unidades	Rendimiento	Precio	Importe
MAT CAC	Conector alta capacidad T macho	Ud	1	0,9	0,9
MAT CBO	Conector batería oro M+H 3,5mm	Ud	24	0,96	23,04
MAT CCP	Conector para cable de potencia	Ud	2	0,304	0,608
MAT USB	Cable micro USB	Ud	1	3,75	3,75
MAT RET	Tubo retractil P/Baterías 45mm	Ud	1	2,08	2,08
MAT REG	Regleta 6mm transparente	Ud	2	0,117	0,234
MAT CPT	Cable de potencia	m	1	2,32	2,32
MAT CCM	Cable de comunicación	Ud	4	0,094	0,376
MAT EST	Bobina de estaño	Ud	1	4,54	4,54
MAQ SSN	Soldador de estaño	h	4,5	11,9	53,55
MAQ FA	Fuente de alimentación de 40A	h	10	1,47	14,7

MAQ OSC	Osciloscopio	h	106,5	6	639
MO GITI	Graduado en Tecnologías Industriales	h	40	35	1400
MO TEC	Técnico de laboratorio	h	4	15	60
Costes directos complementarios		%	0,01	2205,10	22,05
Costes indirectos		%	0,02	2227,15	44,54
Precio total UO					2.271,69 €

UO.2.02 Montaje del quadrotor

Código	Descripción	Unidades	Rendimiento	Precio	Importe
MAT RP2B	Raspberry Pi 2 Modelo B	Ud	1	40,5	40,5
MAT RPA	Raspberry Pi to Arduino Shields Connection Bridge	Ud	1	40	40
MAT IMU	IMU 10DOF (MPU6050 + HMC5883 + MS5611)	Ud	1	18,9	18,9
MAT CT	Convertor de tensión	Ud	1	4,21	4,21
MAT SD	Tarjeta Micro-SD 16GB	Ud	1	5,55	5,55
MAT WF	Wifi USB	Ud	1	16,36	16,36
MAT MOT	Motor ele. Brushless Trainer 1500	Ud	4	26,6	106,4
MAT ESC	TURNIGY Plush 25amp Speed Controller w/BEC	Ud	4	11,97	47,88
MAT CW	Turnigy Slowfly Propeller 8x4.5 Black (CW)	Ud	2	1,09	2,18
MAT CCW	Turnigy Slowfly Propeller 8x4.5 Black (CCW)	Ud	2	1,09	2,18
MAT CHA	Chasis de mikrokopter mk40	Ud	1	62,05	62,05
MAT PCC	Placa Central para Chasis MK-30, 40 o 50	Ud	2	6,7	13,4
MAT POL	Poliestireno expandido 1m2	Ud	1	2	2
MAT BPN	Bote de pintura negra	Ud	1	1,5	1,5
MAT BPV	Bote de pintura verde	Ud	1	1,5	1,5
MAT IN	Imán de neodimio N52 8 x 1mm	Ud	8	0,12	0,96
MAT TUE	Tuerca DIN EN 24032 M2,5	Ud	24	0,03	0,72
MAT TOR20	ISO 7045 H M2,5 x 20 - 4.8 - H	Ud	9	0,04	0,36
MAT TOR8	ISO 7045 H M2,5 x 8 - 4.8 - H	Ud	2	0,03	0,06
MAT E15	Espaciadores 5.6mm x 15mm M2,5 Nylon	Ud	10	0,084	0,84
MAT E10	Espaciadores 5.6mm x 10mm M2,5 Nylon	Ud	12	0,078	0,936
MAT PBQ	Placa de baquelita 10 x 10cm	Ud	1	13,3	13,3
MAT PEG	Pegamento instantáneo (50ml)	Ud	1	2	2
MAT CAD	Cinta aislante	Ud	1	2,3	2,3
MAQ FA	Fuente de alimentación 40A	h	3	1,47	4,41
MAQ KIT	Kit de herramientas	h	15	0,53	7,95
MAQ TAL	Taladradora	h	1	2,1	2,1
MO GITI	Graduado en Tecnologías Industriales	h	30	25	750
MO TEC	Técnico de laboratorio	h	1	15	15
Costes directos complementarios		%	0,01	1165,55	11,66
Costes indirectos		%	0,02	1177,20	23,54
Precio total UO					1.200,75 €

UO.2.03 Adecuación de las piezas de la base de pruebas

Código	Descripción	Unidades	Rendimiento	Precio	Importe
MAT MDF	Tablero MDF 560x400x10mm	Ud	1	4,6	4,6
MAT LCA	Listón cepillado abeto 56x520mm	Ud	1	6,25	6,25
MAT TA45	Tornillo avell poz a.cromo cr3 4x45	Ud	4	0,0716	0,2864
MAT PTN	Bote de pintura negra	Ud	1	1,5	1,5
MAQ SRB	Sierra radial de banco	h	0,15	1,5	0,225
MAQ FM	Fresadora para madera	h	0,5	2	1
MO CAR	Carpintero	h	1	10	10
MO GITI	Graduado en Tecnologías Industriales	h	4	25	100
	Costes directos complementarios	%	0,01	123,86	1,24
	Costes indirectos	%	0,02	125,10	2,51
	Precio total UO				127,60 €

UO.2.04 Construcción de la base de pruebas

Código	Descripción	Unidades	Rendimiento	Precio	Importe
MAT TA20	Tornillo avell poz a.cromo cr3 4,5x20	Ud	8	0,0476	0,3808
MAT TA30	Tornillo avell poz a.cromo cr3 4x30	Ud	12	0,0572	0,6864
MAT APA	Arandela plana ancha a.zinc D.5	Ud	20	0,046	0,92
MAT EAI	Escuadra angulo inox 60x60mm	Ud	4	0,75	3
MAT TC	Tubo de carbono 8x6x1000mm	Ud	1	12,4	12,4
MAT VC	Varilla de carbono 2,5x1000	Ud	1	2,7	2,7
MAT ROT	Rótula Igubal GFSM-08-IG	Ud	1	6,84	6,84
MAT PFV	Placa de fibra de vidrio 1,5x100x100mm	Ud	1	2,63	2,63
MAT TOR12	ISO 7045 H M2,5 x 12 - 4.8 - H	Ud	4	0,035	0,14
MAT TUE	Tuerca DIN EN 24032 M2,5	Ud	4	0,03	0,12
MAT PE	Pegamento epoxi 5 min 128g	Ud	1	11,49	11,49
MAQ SIE	Sierra	h	0,25	0,8	0,2
MAQ KIT	Kit de herramientas	h	1,5	0,53	0,795
MAQ TAL	Taladradora	h	0,5	2,1	1,05
MO GITI	Graduado en Tecnologías Industriales	h	5	25	125
	Costes directos complementarios	%	0,01	168,35	1,68
	Costes indirectos	%	0,02	170,04	3,40
	Precio total UO				173,44 €

3.3. Capítulo 3: Software

UO.3.01 Desarrollo del software					
Código	Descripción	Unidades	Rendimiento	Precio	Importe
MAQ FA	Fuente de alimentación de 40A	h	40	1,47	58,8
MAQ OP	Ordenador portátil	h	80	3	240
MAQ OS	Ordenador de sobremesa	h	100	3,45	345
MO GITI	Graduado en Tecnologías Industriales	h	200	25	5000
	Costes directos complementarios	%	0,01	5643,80	56,44
	Costes indirectos	%	0,02	5700,24	114,01
	Precio total UO				5.814,24 €

3.4. Capítulo 4: Ensayos

UO.4.01 Ensayos del quadrotor					
Código	Descripción	Unidades	Rendimiento	Precio	Importe
MAQ FA	Fuente de alimentación de 40A	h	45	1,47	66,15
MAQ OP	Ordenador portátil	h	60	3	180
MAQ OS	Ordenador de sobremesa	h	60	3,45	207
MO GITI	Graduado en Tecnologías Industriales	h	60	25	1500
	Costes directos complementarios	%	0,01	1953,15	19,53
	Costes indirectos	%	0,02	1972,68	39,45
	Precio total UO				2.012,14 €

4. Presupuesto parcial

4.1. Capítulo 1: Diseño

Código	Concepto	Precio
UO.1.01	Diseño del quadrotor	2.549,75 €
UO.1.02	Diseño de la base de pruebas	1.380,47 €
Total Capítulo:		3.930,21 €

4.2. Capítulo 2: Construcción

Código	Concepto	Precio
UO.2.01	Adecuación de conexiones electrónicas y eléctricas	2.271,69 €
UO.2.02	Montaje del quadrotor	1.200,75 €
UO.2.03	Adecuación de las piezas de la base de pruebas	127,60 €
UO.2.04	Construcción de la base de pruebas	173,44 €
Total Capítulo:		3.773,48 €

4.3. Capítulo 3: Software

Código	Concepto	Precio
UO.3.01	Desarrollo del software	5.814,24 €
Total Capítulo:		5.814,24 €

4.4. Capítulo 4: Ensayos

Código	Concepto	Precio
UO.4.01	Ensayos del quadrotor	2.012,14 €
Total Capítulo:		2.012,14 €

5. Presupuesto total

Capítulo	Concepto	Precio
Cap. 1	Diseño	3.930,21 €
Cap. 2	Construcción	3.773,48 €
Cap. 3	Software	5.814,24 €
Cap. 4	Ensayos	2.012,14 €
PRESUPUESTO DE EJECUCIÓN MATERIAL		15.530,07 €
	15% Gastos generales	2.329,51 €
	6% Beneficio industrial	931,80 €
PRESUPUESTO DE EJECUCIÓN POR CONTRATA		18.791,38 €
	21% IVA	3.946,19 €
PRESUPUESTO BASE DE LICITACIÓN		22.737,57 €

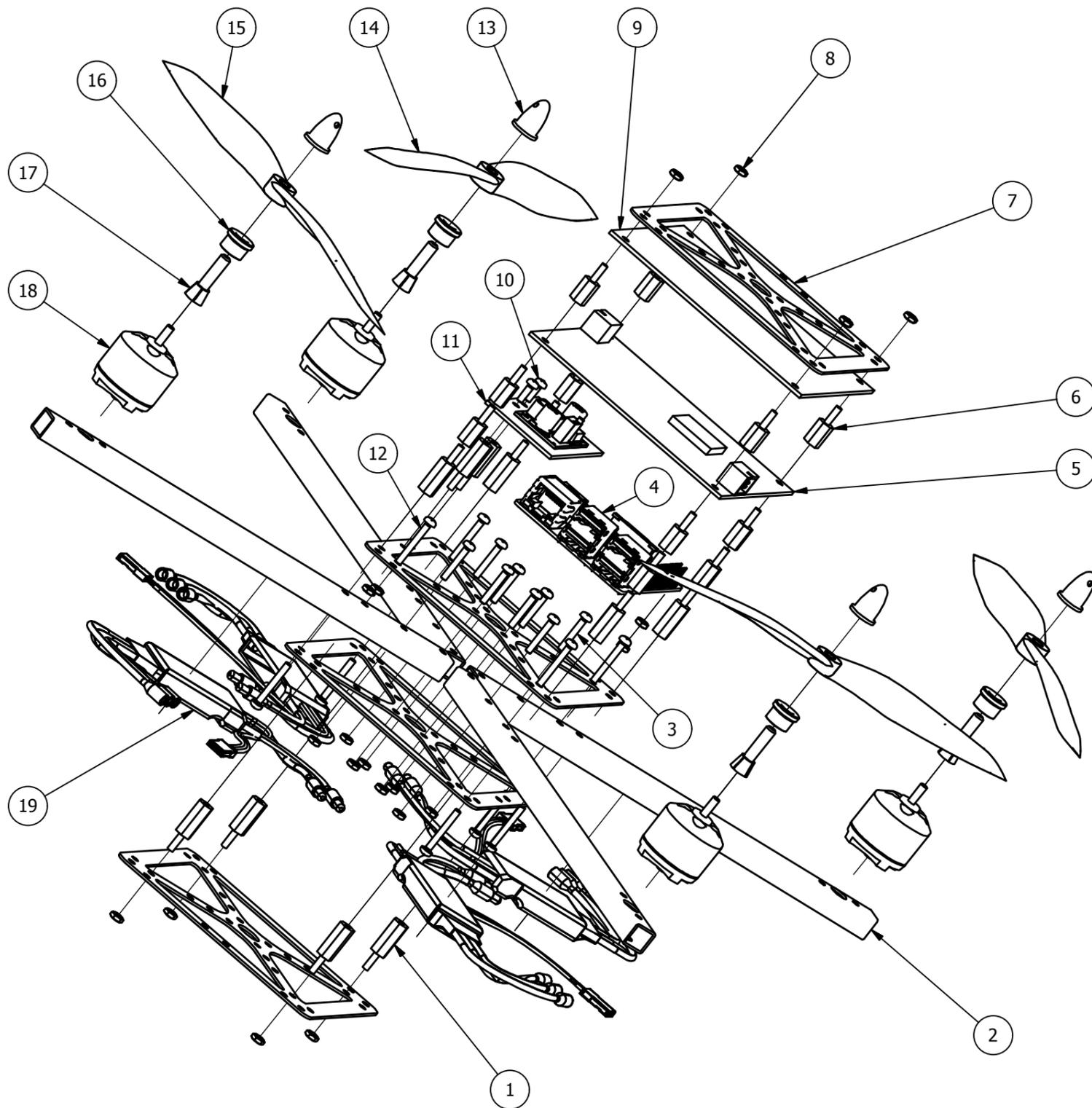
Asciende el presupuesto base de licitación a la cantidad de:

VEINTIDOS MIL SETECIENTOS TREINTA Y SIETE EUROS CON CINCUENTA Y SIETE CÉNTIMOS.

PLANOS

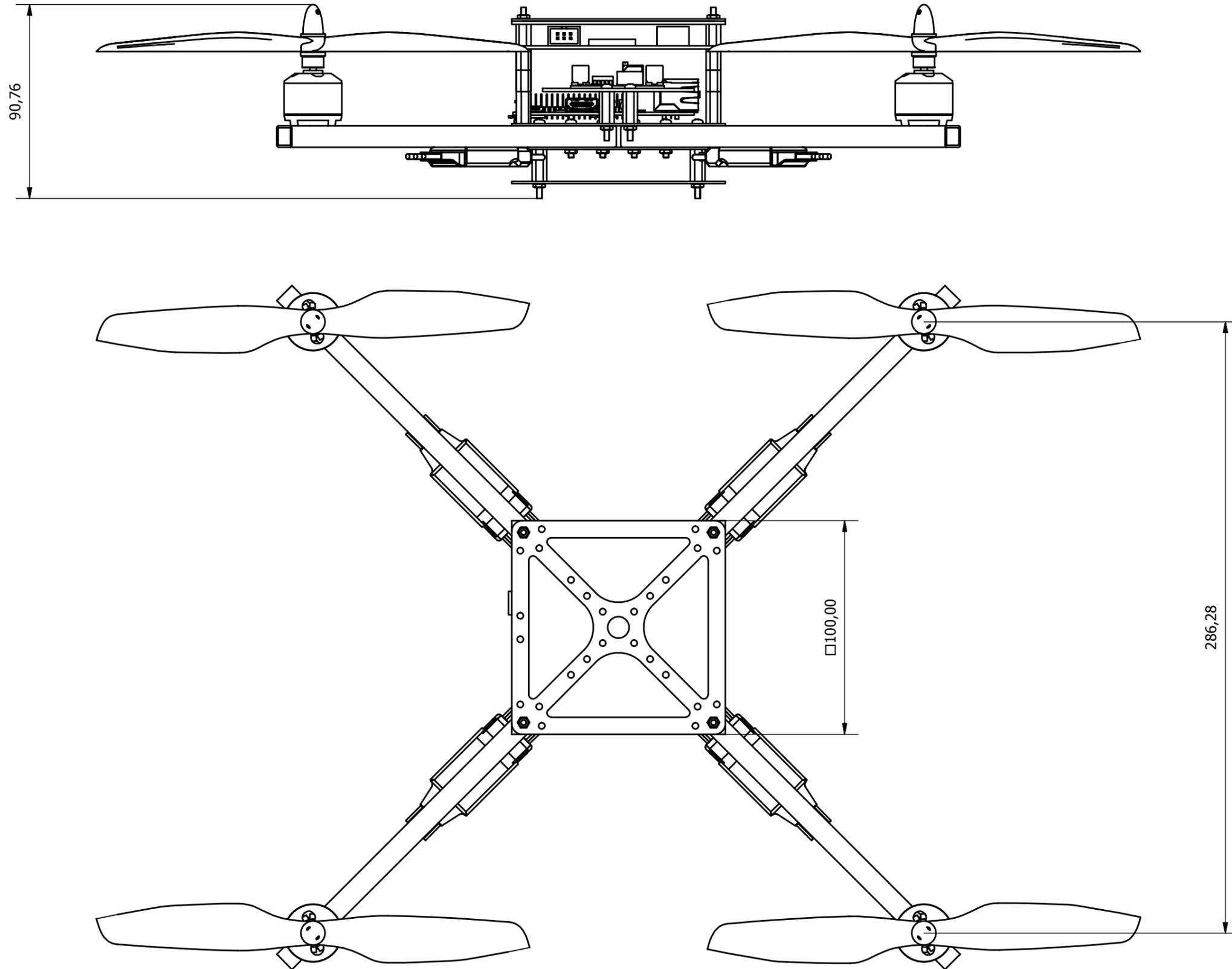
Índice de planos

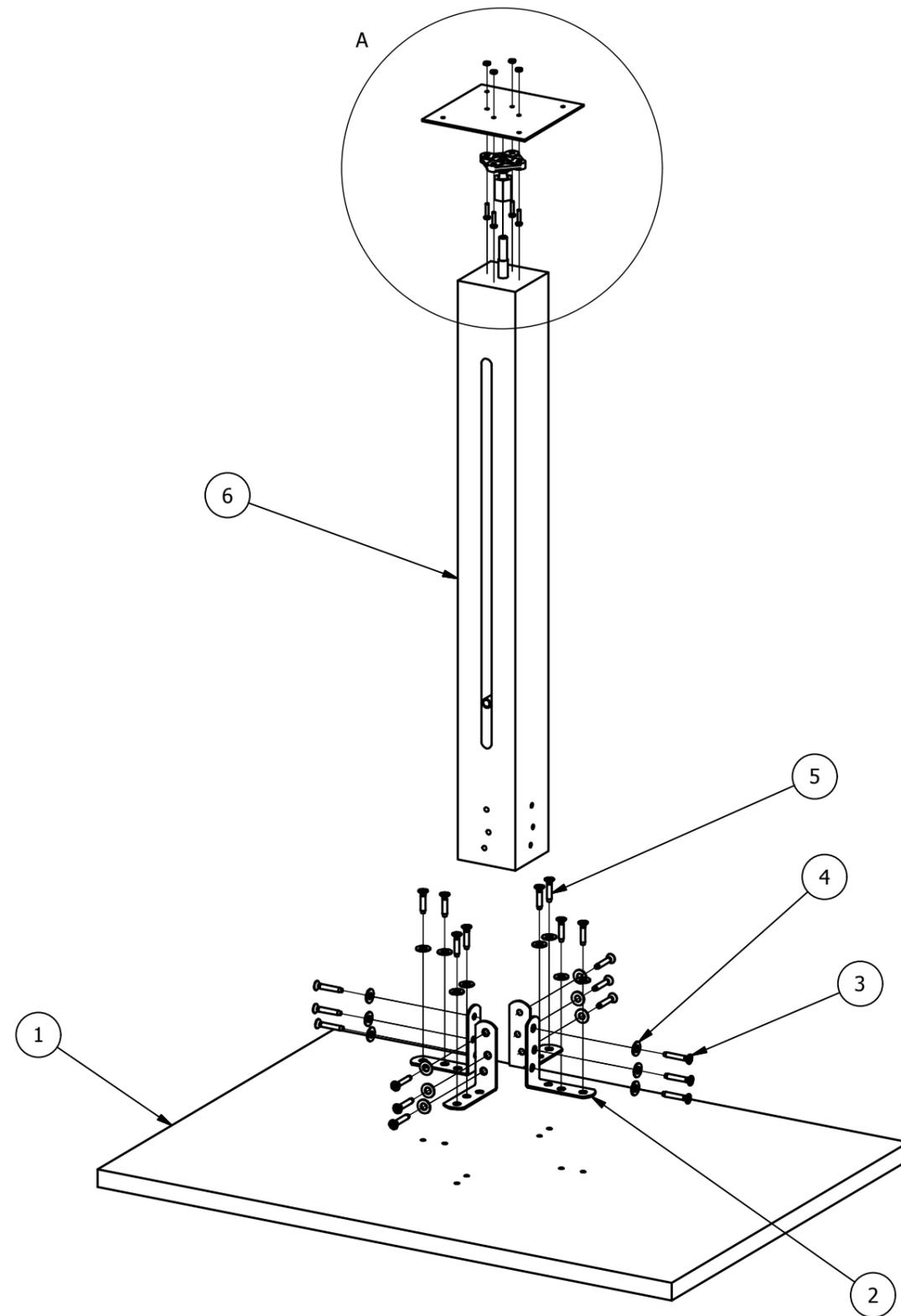
- Plano de conjunto en explosión del quadrotor..... 1**
- Plano de conjunto del quadrotor 2**
- Plano de conjunto en explosión de la base de pruebas 3**
- Plano de conjunto de la base de pruebas 4**



LISTA DE PIEZAS

ELEMENTO	CTDAD	Nº DE PIEZA	DESCRIPCIÓN
1	10	Separador 15 mm	Separador de nylon de 15 mm con saliente y hueco roscados
2	3	Brazo	Tubo de aluminio de sección cuadrada con perforaciones para sujeción
3	8	ISO 7045 - M2,5 x 16 - 4.8 - H	Tornillo de cabeza cilíndrica redondeada con hueco cruciforme tipo H - producto de clase A
4	1	Raspberry Pi 2	Placa Raspberry Pi 2 Modelo B
5	1	IMU	Placa soporte para IMU con puertos de conexión
6	12	Separador 10 mm	Separador de nylon de 10 mm con saliente y hueco roscados
7	4	Placa radial	Placa perforada para chasis
8	20	ISO 4032 - M2,5	Tuerca hexagonal
9	1	Placa de protección	Placa de fibra de vidrio
10	2	ISO 7045 - M2,5 x 8 - 4.8 - H	Tornillo de cabeza cilíndrica redondeada con hueco cruciforme tipo H - producto de clase A
11	1	Convertidor de tensión	Convertidor de tensiones acoplado a una placa con conexiones de entrada y salida de corriente
12	9	ISO 7045 - M2,5 x 20 - 4.8 - H	Tornillo de cabeza cilíndrica redondeada con hueco cruciforme tipo H - producto de clase A
13	4	Porta-hélice sup.	Tope de aluminio roscado para la sujeción de la hélice
14	2	Hélice 8x4.5 CCW	Hélice de resina ABS de 8 pulgadas de largo y 4.5 pulgadas de paso con giro antihorario
15	2	Hélice 8x4.5 CW	Hélice de resina ABS de 8 pulgadas de largo y 4.5 pulgadas de paso con giro horario
16	4	Porta-hélice inf.	Anillo de aluminio para el apoyo de la hélice
17	4	Eje porta-hélice	Adaptador de eje para hélice de aluminio
18	4	Motor	Motor brushless Yellow Trainer 1500 de 16 A
19	4	Driver	Driver ESC Turnigy Basic 25 A





LISTA DE PIEZAS			
ELEMENTO	CTDAD	Nº DE PIEZA	DESCRIPCIÓN
1	1	Pie	Tablero MDF
2	4	Escuadra	Escuadra angulo inox.
3	12	ANSI B18.6.5M - M4x0,7 x 25	Tornillo de cabeza plana avellanada con hueco cruciforme - Tipo F - Tipo IA - Métrico
4	20	Arandela	Arandela plana ancha a.zinc D.5
5	8	ANSI B18.6.5M - M4x0,7 x 20	Tornillo de cabeza plana avellanada con hueco cruciforme - Tipo F - Tipo IA - Métrico
6	1	Mástil	Listón cepillado de abeto
7	1	Tubo de carbono	Tubo de carbono de
8	1	Rótula	Rótula metálica inox. Igubal GFSM-08-IG
9	1	Alojamiento de rótula	Alojamiento polimérico de rótula Igubal GFSM-08-IG
10	4	ISO 7045 - M2,5 x 12 - 4.8 - H	Tornillo de cabeza cilíndrica redondeada con hueco cruciforme tipo H - producto de clase A
11	1	Placa de acople	Placa de fibra de vidrio con agujeros para el acople del quadrotor
12	8	ISO 4032 - M2,5	Tuerca hexagonal

