

**TESIS DE GRADO
EN INGENIERÍA ELECTRÓNICA**

**ESTABILIZADOR DE CÁMARAS DE
FILMACIÓN**

Autores	Legajos
Joan Martorell	49294
Rodrigo Elías Lorenzo Leal	50577
Pablo Agustín Remedi	52125
Martín Biondi Sarra	51631

Índice general

I. Prefacio y agradecimientos	7
II. Resumen	9
III. Introducción	11
III.1. Historia y antecedentes	11
III.2. Definiciones y glosario	12
III.3. Justificación del proyecto	13
IV. Objetivos	15
IV.1. Finalidad del proyecto	15
IV.2. Planteamiento del problema a resolver	15
V. Definición del Producto	17
V.1. Requerimientos	17
V.1.1. Construcción de la Casa de calidad	18
V.2. Especificaciones funcionales y de diseño	19
V.2.1. Especificaciones del hardware	19
V.2.2. Especificaciones del software	20
VI. Análisis de Factibilidad	21
VI.1. Factibilidad tecnológica	21
VI.1.1. Estructura	21
VI.1.2. Motores	23
VI.1.3. Control remoto	25
VI.1.4. Elección de una solución	26
VI.1.5. FMEA	27
VI.2. Factibilidad de tiempos	31
VI.2.1. Planificación	31
VI.2.2. Programación	35
VI.3. Factibilidad económica	37
VI.3.1. El Negocio	37
VI.3.2. El Servicio	37
VI.3.3. El Mercado	37
VI.3.4. Plan de Marketing	38
VI.3.5. Análisis Financiero	39
VI.4. Factibilidad legal y responsabilidad civil	41

VII Ingeniería del detalle	43
VII.1 Hardware	43
VII.1.1. Diagrama en bloques	43
VII.1.2. Descripción de cada bloque	44
VII.1.3. Detalles de selección y cálculo de componentes	46
VII.1.4. Plan de pruebas de cada módulo	53
VII.2 Software	56
VII.2.1. Diagramas de estado y flujogramas	56
VII.2.2. Análisis de complejidad	68
VII.2.3. Descripción de subrutinas	69
VII.2.4. Listados comentados de código	71
VII.2.5. Plan de prueba de módulos y de depuración de software	81
VIII Construcción del prototipo	83
VIII.1 Definición de los módulos	83
VIII.1.1 Unidad Controladora Principal	83
VIII.1.2 Sensor de inclinación	83
VIII.1.3 Receptor RF	83
VIII.1.4 Control Remoto	83
VIII.1.5 Aplicación de PC	83
VIII.2 Diseño de los circuitos impresos	84
VIII.2.1 Microcontrolador	84
VIII.2.2 Interfaz USB-UART	85
VIII.2.3 Controladores de Motor	86
VIII.2.4 Fuente de alimentación	86
VIII.2.5 Diseño de la placa principal	86
VIII.2.6 Diseño de las placas del control remoto	88
VIII.3 Diseño mecánico	92
VIII.3.1 Diseño mecánico del control remoto	92
VIII.4 Detalles de construcción y precauciones especiales de montaje	93
VIII.4.1 Control remoto	93
IX. Validación del prototipo	95
IX.1. Validación de hardware	95
IX.1.1. Plan y protocolos especiales de medición	95
IX.1.2. Mediciones	97
IX.1.3. Evaluación	99
IX.1.4. Resultados	100
IX.2. Validación de software	100
IX.2.1. Plan y protocolos especiales de medición	100
IX.2.2. Mediciones	101
IX.2.3. Evaluación	102
IX.2.4. Resultados	102
X. Estudios de confiabilidad de hardware y de software	103
X.1. Hardware	103
X.2. Software	105

XI. Conclusiones	107
XI.1. Conclusiones excelencias y objetivos alcanzados	107
XI.2. Fallos. Recomendaciones para diseños futuros	107
XII Anexos	109
XII.1. Contrato modelo de locación de equipo	109
XII.2. Manual de Mantenimiento	112
XII.2.1. STATHEROS	112
XII.2.2. Seguridad	113
XII.2.3. Calibración Física	113
XII.2.4. Calibración Digital	119
XII.2.5. Recambio de un Motor de Pitch	121
XIII Bibliografía	127

I Prefacio y agradecimientos

A nuestras familias, por todo su apoyo durante la carrera, tanto en el ámbito académico como en el personal, sin ellos no hubiéramos llegado hasta aquí.

A nuestros amigos, por compartir con nosotros los momentos buenos, y ayudarnos en los momentos malos.

A nuestros profesores, por darnos las herramientas necesarias para realizar este proyecto.

A Francisco DiTaranto, por mostrarnos el verdadero camino del aprendizaje.

A Gabriel, Nelson y los muchachos del pañol, por ayudarnos día a día a hacer nuestro trabajo.

Agradecimientos especiales al Grandísimo Jorge Cáceres, ya que sin su ayuda no hubiéramos podido entregar a tiempo el proyecto.

Y finalmente a Pipino, algún día llegaremos a California.

II Resumen

La industria cinematográfica Argentina es un sector que experimentó un gran crecimiento en los últimos años. Una parte esencial de este rubro es la estabilización de cámaras, que está dominada por Steadicam. En este trabajo, se realiza un desarrollo íntegro de un sistema activo de tipo *gimbal*, que busca desplazar a la marca vigente en el sector de cámaras pequeñas (menor a 2kg) y medianas (entre 2kg y 6kg).

Para llevar a cabo este proyecto, se realiza un análisis de mercado entrevistando a directores de fotografía y operadores de cámaras, con el fin de crear un producto que se ajuste a sus necesidades. Posteriormente se adquiere una estructura prefabricada y se desarrolla una placa de control que se encarga de realizar la estabilización activa. El resultado es un producto que pesa 2kg, es decir, un 80% menos que la Steadicam, no requiere de conocimiento especializado para su operación y es muy compacto, permitiendo hacer tomas en espacios reducidos que serían imposibles para la competencia.

III Introducción

III.1 Historia y antecedentes

Uno de los más comunes retos a la hora de realizar una toma en movimiento, es el de minimizar las perturbaciones externas, usualmente generadas por temblores, que afectan la calidad de la toma. Antes de los sistemas para estabilización de cámaras, un director tenía básicamente dos opciones: la primera era usar un *dolly* (un carro montado sobre un riel, ver Figura III.1.1) para estabilizar la cámara, lo cual consumía demasiado tiempo, requería de instalaciones fijas, y no permitía flexibilidad en la dirección de la toma; la segunda era contratar un operador de cámara experimentado, que permitía mayor flexibilidad y menores tiempos de preparación, a costa de una menor calidad de la toma.



Figura III.1.1.: Toma en movimiento realizada con dolly

En 1971 el director de fotografía Garret Brown introduce el Steadicam, el primer sistema de estabilización de cámara portátil. Utilizando un sistema de contrapesos que se monta sobre un arnés, un operador de cámara puede moverse con cierto grado de libertad y lograr una toma con perturbaciones mínimas. En 1976 el sistema Steadicam hace su debut en Hollywood: el director Hal Ashby utiliza este sistema para realizar tomas largas y continuas en su película “*Bound for Glory*”. Posteriormente, en 1980, Brown introduce el sistema *low mode* (Figura III.1.2), el cual permitía al operador realizar tomas por debajo del nivel de su cintura; este sistema fue empleado el mismo año en la película “*The Shining*” de Stanley Kubrik.



Figura III.1.2.: Operador de Steadicam con low mode

A lo largo de los años se ha perfeccionado el Steadicam, pero en esencia sigue siendo un sistema de contrapesos. Adicionalmente, el uso de Steadicam se ha vuelto una práctica regulada, lo que implica que para manejar este sistema se requiere la realización de un curso con su consecuente certificación.

Con el desarrollo de los motores *brushless* y el auge de la electrónica digital, surgen sistemas activos de estabilización (Figura III.1.3) , que compensan el movimiento angular en los 3 ejes de la cámara. Esto se logra midiendo el ángulo de inclinación utilizando una IMU y realimentando la posición a los motores, con el fin de establecer un plano fijo sin perturbaciones. Este sistema trae apareadas ventajas notorias frente al Steadicam, como lo son la reducción del tamaño y peso del dispositivo, y la mayor facilidad de uso, que elimina la necesidad de cursos preparatorios. Sin embargo, su utilización en rodajes es limitada, debido a que se ha construido durante 30 años un mercado alrededor de la tecnología dominante, poniendo una barrera muy alta al ingreso de nuevos sistemas de estabilización.



Figura III.1.3.: Toma realizada con un gimbal

III.2 Definiciones y glosario

- *Gimbal*: estructura de suspensión que mediante apoyos de rotación permite el balanceo de un objeto.
- Steadicam: marca comercial de sistemas de estabilización de cámaras por contrapeso (ver sección Historia y antecedentes).
- *Pitch*: cabeceo, rotación sobre el eje del manubrio (rotación “vertical”).
- *Roll*: alabeo, rotación sobre el eje de la cámara.

- *Yaw*: guiñada, rotación sobre el eje Z (rotación “horizontal”).
- **COG**: *Center of Gravity*; este término refiere al centro de gravedad de un sistema u objeto, el cual se define como el punto respecto del cual el peso coordenado del sistema u objeto vale cero.
- **IMU**: el *Inertial Measurement Unit*, o Unidad de Medición Inercial, es un dispositivo que mide aceleración de la gravedad y velocidad angular.
- **MEMS**: los *Micro ElectroMechanical Systems*, o Sistemas Microelectromecánicos, es un tipo de tecnología que utiliza sistemas mecánicos implementados en componentes micrométricos.
- **Driver**: bloque circuital que realiza el pasaje de lógica a potencia para manejar los motores.
- **BLDC**: motor sin escobillas de poco desgaste (*BrushLess Direct Current motor*).
- **Tail-heavy**: literalmente, “pesado en la cola”. Expresión utilizada para indicar que, durante el proceso de calibración, la cámara tiene su peso concentrado por detrás del eje pivotante imaginario de *pitch*.
- **Nose-heavy**: literalmente, “pesado en la nariz”. Expresión utilizada para indicar que, durante el proceso de calibración, la cámara tiene su peso concentrado por delante del eje pivotante imaginario de *pitch*.
- **Quick-release plate**: Plataforma utilizada en fotografía, en donde se atornilla el equipo de filmación. Permite el ajuste rápido de posición adelante-atrás utilizando un tornillo manual.

III.3 Justificación del proyecto

Se identificó una potencial aplicación para estabilizadores de cámara en la industria cinematográfica Argentina, donde el mercado de estos dispositivos es inexistente. Buscando explorar esta veta, el grupo entró en contacto con un director de fotografía, usuario de sistemas Steadicam, que mostró un gran interés por el producto. Esta persona explicó que los elementos necesarios para la filmación se alquilan por jornadas de proveedores especializados. Con esta información, se decidió descartar la idea inicial de vender el *gimbal* y, en vez, brindar un servicio integral que contemple la calibración del dispositivo y soporte técnico. Asimismo, compartió requerimientos que ayudaron a establecer especificaciones que apuntan a satisfacer las necesidades del mercado.

De esta manera, la innovación que se propone es en la comercialización del *gimbal*, lo que implica un cambio en el enfoque inicial del proyecto, ya que al alquilar el producto, el costo del equipo no es un factor determinante, puesto que se planea amortizarlo con varias sesiones de filmación. Luego de una investigación orientada a factibilidad tecnológica, que incluyó el contacto con varios proveedores de motores y estructuras para *gimbal*, se concluyó que era factible desarrollar un producto funcional para satisfacer las necesidades de este mercado aliciente.

IV Objetivos

IV.1 Finalidad del proyecto

Este proyecto se orienta a satisfacer las necesidades de directores de fotografía o filmación, y a las productoras de cine y publicidad. El objetivo es lograr que estas entidades reduzcan sus costos, se liberen de la dependencia del operador de Steadicam, y logren mayor nivel de flexibilidad a la hora de realizar tomas en movimiento. Estos objetivos se logran principalmente simplificando el uso del sistema, de forma tal que cualquier persona familiarizada con el uso de cámaras logre utilizarlo sin dificultades. Así, el proyecto implica el diseño de un producto superior a Steadicam y con más prestaciones que los sistemas activos existentes al momento que luego sería comercializado mediante alquiler.

IV.2 Planteamiento del problema a resolver

En Argentina, cuando un director requiere una toma estabilizada, debe necesariamente recurrir a un operador certificado de Steadicam, quien a su vez es el propietario de este equipo. En el país existe sólo un reducido número de operadores certificados a los que se puede recurrir, y estos no se encuentran regulados por ningún gremio o entidad. Los problemas que surgen de esta dinámica son los siguientes:

- El director queda a disposición del operador, y no al revés; es decir, los tiempos de trabajo son los del operador, y las limitaciones también las impone este último.
- El director pierde control de la toma al no ser él quien la encuadra. Asimismo, no todo operador de Steadicam es efectivo para realizar esta tarea por lo cual una toma puede requerir varias repeticiones.
- El director termina, en muchos casos, aceptando y acostumbrándose a las limitaciones de un único operador, lo cual afecta su trabajo.
- En muchos casos se debe acondicionar el set de filmación para realizar las tomas con Steadicam.

La mayoría de estos problemas se solucionan permitiendo al director utilizar a su propio operador de cámara para las tomas que anteriormente hubieran requerido al operador de Steadicam.

V Definición del Producto

Luego de detectar la necesidad del mercado cinematográfico de contar con una alternativa más versátil en materia de estabilización de cámaras, se emprendió una recopilación de información de los posibles usuarios. En especial, se entrevistaron dos directores de fotografía, uno con experiencia en producciones cinematográficas y otro dedicado a la publicidad. Ambas personas, usuarias de Steadicam para realizar tomas en movimiento, expresaron un deseo de contar con una alternativa más versátil, dado que consideran que la tecnología existente impone ciertos obstáculos al proceso creativo (ver sección IV.2). Lo que se detalla a continuación son los requerimientos que emanaron de sendas entrevistas.

V.1 Requerimientos

Efectuando un análisis de los expresado por los interesados, se puede deducir que pretenden un funcionamiento al menos igual al del sistema existente, puesto que están acostumbrados a un régimen de trabajo determinado. En particular, requieren una autonomía de 6 horas, aceptando la posibilidad de recambiar baterías agotadas por nuevas. Por ejemplo, se pueden tener 3 que duren 2 horas cada una, con la condición que debe ser fácil reemplazarlas y lleguen a plena carga en menos de 4 horas. Por otro lado, ya tienen contemplado un tiempo de calibración que no debe superar los 45 minutos. Por último, necesitan que el sistema sea capaz de balancear una cámara Epic Scarlet con lente Carl Zeiss Ultra Prime o similar (peso equivalente de 5kg), con una efectividad similar a la de Steadicam.

Además de requerir una especie de compatibilidad con la metodología y desempeño al que están acostumbrados, indicaron mejoras deseadas al sistema existente. A saber, que se pueda realizar el encuadre de manera remota, que el aparato sea compacto y que pueda operarse por cualquier persona, sin necesidad de entrenamiento. Son estos requerimientos los que diferenciarán el producto con la competencia y, en definitiva, satisfacerán las necesidades existentes del mercado. Cabe destacar que este control remoto no controlara el enfoque o zoom de la cámara, ya que esas funciones se incluyen en el control remoto de la misma.

- Precisión de balanceo: capaz de corregir movimientos por trote del camarógrafo.
- Modos de operación:
 1. Seguimiento de movimiento suavizado para paneos.
 2. Fijación completa de la cámara.
 3. Fijación por control remoto.
- Tiempo de operación: sesiones de 6 horas.
- Tiempo de calibración: menor a 45 minutos.
- Tamaño: capaz de pasar por una ventana de $1,6m^2$.
- Condiciones ambientales de operación: capaz de mojarse, soportar vientos fuertes, y temperaturas extremas de tundra y desierto.

- Capacidad de carga: necesariamente capaz de balancear una Epic Scarlet con lente Carl Zeiss Ultra Prime (5kg).

V.1.1 Construcción de la Casa de calidad

Los requerimientos obtenidos mediante entrevistas deben ser interpretados debidamente para formar las especificaciones del producto. Sin embargo, previo a esto es conveniente determinar las relaciones entre las necesidades y los parámetros de diseño que se disponen para materializar una solución pertinente. Una manera de visualizar las conexiones es utilizando una casa de calidad, como la que se presenta en la Figura V.1.1. Este método de representación también cuenta con un análisis comparativo con la competencia. De aquí podemos ver, por ejemplo, que la elección de la batería está fuertemente relacionada con el peso de la estructura, el torque de los motores y el alcance de la comunicación inalámbrica. Estas, a su vez, están muy ligadas a la capacidad de carga del *gimbal*, que se destacó como una característica central del producto.

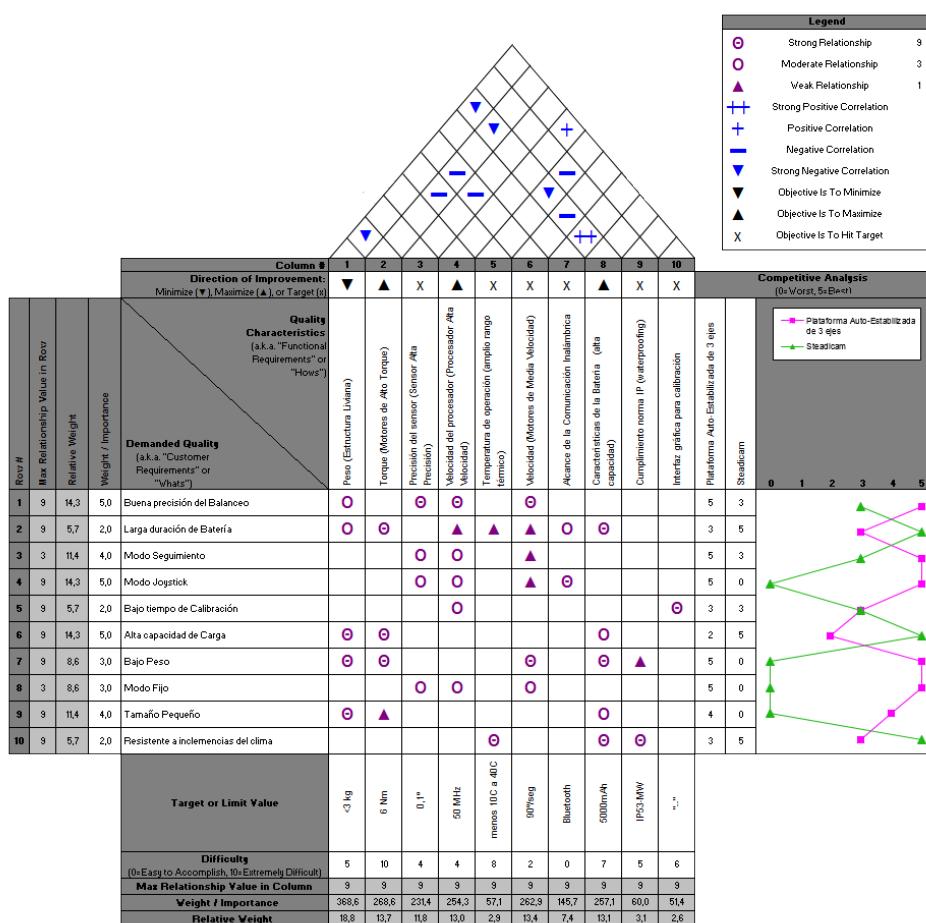


Figura V.1.1.: Casa de calidad

La comparación con la competencia realizada indica que la plataforma estabilizada de 3 ejes supera a la Steadicam en casi todos los aspectos, excepto en la capacidad de carga y duración de la batería. Esto se debe a que el sistema de contrapesos es más robusto dado que no posee componentes electrónicos y, por lo tanto, no necesita fuente de alimentación. No obstante, esta aparente debilidad es en realidad el punto donde el producto que se ofrece brinda una ventaja competitiva; al incorporar electrónica, se puede contar con la posibilidad de controlar el encuadre de forma remota y reducir considerablemente el peso y las dimensiones del estabilizador.

V.2 Especificaciones funcionales y de diseño

La tarea de formular las especificaciones de diseño se apoya en el análisis de los requerimientos establecidos previamente y constituye un aspecto vital en el proceso de diseño. Debido a que derivan de una interpretación de los requerimientos, se tuvo que emplear una mezcla de conocimiento de ingeniería e imaginación.

V.2.1 Especificaciones del hardware

Módulo	Detalle	Especificación
Motores	Velocidad máxima	90 °/seg
	Torque (<i>yaw</i> y <i>roll</i>)	6 N/m ± 20%
	Resistencia interna (<i>yaw</i> y <i>roll</i>)	19.3 Ω ± 10%
	Torque (<i>pitch</i>)	4 N/m ± 20%
	Resistencia interna (<i>pitch</i>)	10.6 Ω ± 10%
	Tensión nominal	12V ± 20%
Batería	Tensión	12V ± 10%
	Capacidad	≥ 5000 mAh
	Corriente	≥ 5C
MCU	Canales	9 x PWM
		2 x SCI
		1 x I2C
		10 x GPIO
Sensor	Resolución angular	± 1°
	Tasa de refresco	≥ 1kHz
Control remoto	Alcance mínimo	50 m
	Cantidad de palancas	2
	Precisión de las palancas	± 1°
	Cantidad de botones	4
	Indicación Visual	LED RGB
	Batería	5V @ 500 mAh
General	Temperatura	-10 °C a 40 °C
Conectores	PC	Mini USB
	Batería	Polarizado DIN
	Motores	3 Terminales Normal
	Sensor	4 Terminales Polarizado
	Módulo inalámbrico	4 Terminales Polarizado

Figura V.2.1.: Especificaciones técnicas del hardware

Motores

Para determinar la velocidad máxima de corrección de ángulo se ensayaron movimientos bruscos con un tubo horizontal con manubrios, similar a lo que se tendrá en el producto final. Estos movimientos fueron mucho más violentos que los detectados en la prueba de trote, donde se registraron movimientos de 20°/seg.

En cuanto al torque, se previó un desalineamiento máximo del centro de masa con respecto al eje de rotación de 10cm que, con una cámara de 5kg sumada a la mitad del peso de la estructura (1kg), conforman la especificación. Comercialmente se obtienen motores de 6N/m y de 4N/m, de allí la elección del torque de los más pequeños (*pitch*). Como el mercado de BLDC de estas características es de *hobbyistas*, la tensión de alimentación preferencial es de 12V y admite amplias variaciones, siempre que se mantenga la temperatura de los bobinados por debajo del máximo especificado por el fabricante.

Batería

Las especificaciones del motor hacen que el consumo promedioSe refiere al consumo de corriente del motor en condiciones que la cámara menos de 5° de la posición de equilibrio del *gimbal* sea de 1A.

V.2.2 Especificaciones del software

Módulo	Detalle	Descripción
Interfaz Grafica	USB	El programa debe permitir poder recibir y enviar datos vía USB con la placa controladora.
	Cajas de Texto	El programa debe poder variar los parámetros de calibración de la placa a través de cajas de texto.
	Sliders	El programa debe poder variar los parámetros de calibración de la placa a través de Sliders.
	Habilitación y Des-habilitación de Motores	El programa debe poder habilitar y deshabilitar cualquiera de los motores a través de un botón o pulsador.
	Graficador	El programa debe poder mostrar en tiempo real, y con fluidez los valores que recibe del puerto de comunicación seleccionado por el usuario.
Portabilidad	Portabilidad	El programa debe ser ejecutable en las principales plataformas de computadoras personales disponibles en la actualidad.

Figura V.2.2.: Especificaciones técnicas de software

VI Análisis de Factibilidad

VI.1 Factibilidad tecnológica

Se han planteado diferentes alternativas de diseño para los diferentes componentes principales del proyecto. A continuación se detallará cada uno de estos componentes, así como las alternativas de diseño y sus ventajas y desventajas.

VI.1.1 Estructura

VI.1.1.1 Forma mecánica

Los criterios a tener en cuenta para seleccionar la forma de la estructura son:

- Cantidad de material: a mayor cantidad de material no sólo aumenta el costo del producto, sino que también el peso de la estructura, lo cual va en contra de los requerimientos.
- Tamaño: los requerimientos imponen una cota superior para el tamaño de la estructura, por lo que se debe trabajar para mantenerse por debajo de las dimensiones máximas.
- Requerimiento de motores: ciertas estructuras tienen la capacidad de aprovechar su geometría de forma tal que el torque requerido por parte de los motores sea mínimo; una estructura sin esta característica requerirá de motores de mayor potencia, más costosos, voluminosos y pesados.
- Maniobrabilidad: una característica muy importante para el diseño, determina la facilidad con la que puede manejarse la estructura en conjunto con la cámara.
- Tiempo de desarrollo o adquisición: para ajustarse de forma fiel a los diagramas de tiempos realizados para este proyecto, es importante que el tiempo de desarrollo o adquisición de la solución sea lo más acotado posible.

En cuanto a la forma de la estructura, surgen tres potenciales soluciones, algunas de las cuales exploran la idea de un sistema pasivo. Estas son:

Gimbal

La estructura *gimbal* (ver Figura VI.1.1) tiene ventajas desde el punto de vista técnico y de interacción con el usuario. Los manubrios laterales y el superior propician un manejo intuitivo, que se traduce en facilidad para realizar tomas de todo tipo. Por otro lado, la forma de la estructura permite hacer coincidir el centro de gravedad de la cámara con los tres ejes rotacionales de los motores. En el caso idealizado, la cámara permanecería quieta sin necesidad de motores, pero el rozamiento hace que rote alrededor de los ejes. Entonces, los motores sólo deben proveer el torque necesario para vencer estas fuerzas, que es mucho menor al que deberían proporcionar si tuvieran que mover la cámara en una posición arbitraria. Es decir, el diseño de la estructura desplaza la complejidad de los motores y la sitúa en el proceso de calibración:

cuanto mejor esté balanceada, tanto menor será el tamaño de los motores. Consecuentemente, se debe emplear tiempo en perfeccionar las técnicas de balanceo y calibración del dispositivo previo a su uso.



Figura VI.1.1.: Estructura gimbal para cámaras estabilizadas

Brazo con plataforma

Consiste en un brazo mecanizado de tres articulaciones con una bandeja en uno de sus extremos, como el que se muestra en la Figura 1. Este diseño tiene un mayor parentesco con la Steadicam, porque está pensado para acoplarse a un arnés y no para manipular con las manos. Por lo tanto, hereda las restricciones de espacio de este sistema y no es apto para tomas en espacio reducidos. Desde el punto de vista técnico, es importante remarcar que dos de los tres motores deben poder mover el peso bruto de la cámara (el de yaw puede simplificarse haciendo coincidir el COG de la cámara con el eje de rotación). Asimismo, los esfuerzos no son simétricos en los dos motores más exigidos porque se encuentran a distancias distintas del COG de la cámara; de hecho, la distancia varía con el mismo movimiento de los motores.

Las características de este diseño hace que no sea el apropiado para la aplicación, debido a que no cumplen con las especificaciones propuestas y dificultan la implementación.

Suspensión Cardán

La estabilización por suspensión de cardán emplea una serie de anillos concéntricos pivoteados (ver Figura VI.1.2) que mantienen al elemento a estabilizar en una posición fija. Fue implementado ampliamente durante la Era de los Descubrimientos para estabilizar brújulas en los barcos de los exploradores europeos. Este sistema es el precursor del *gimbal* moderno, pero no incluye motores para compensación y, por lo tanto, no permite el encuadre remoto. El mayor problema de este sistema es que los anillos pueden pasar por delante de la lente de la cámara y arruinar la toma, por lo que esta alternativa queda descartada.



Figura VI.1.2.: Estructura estabilizadora por suspensión de cardán

VI.1.1.2 Material de construcción

Habiendo definido la forma de la estructura, se plantea otra disyuntiva, relacionada con el material de construcción. Los criterios a tener en cuenta para decidir sobre este aspecto serán los siguientes:

- Costo: por motivos obvios, es preferible un material de menor costo.
- Peso: a menor peso, la estructura es más liviana, lo cual no sólo hace las especificaciones más fáciles de cumplir, sino que además ayuda a ubicar el centro de masa de todo el sistema en la cámara, simplificando así el proceso de calibración.
- Disponibilidad.
- Resistencia: sería altamente favorable que el material resista golpes, dado que, si bien no se espera que el producto sufra impactos, los accidentes suelen ocurrir.
- Durabilidad: este criterio es de vital importancia, debido a que, al comercializar el producto en forma de alquiler, se requiere que tenga la mayor durabilidad posible para garantizar la rentabilidad del negocio.

A pesar de que hay muchos tipos de materiales, se reduce la elección a sólo 2, debido a su extensa utilización en el ámbito del cine y publicidad:

Aluminio

Las aleaciones de aluminio son fáciles de moldear, baratas (excepto variantes como Duraluminium), es fácil encontrar proveedores y son resistentes a la corrosión. Sin embargo, se deforman con relativa facilidad frente a impactos, lo cual constituye una desventaja porque el *gimbal* debe estar preparado a soportar golpes y caídas de poca altura (1.5m). Por otro lado, son más pesados que su equivalente en fibra de carbono y menos resistentes a rotura por corte.

Fibra de carbono

Los tubos hechos de este material son muy resistentes a esfuerzos en una dirección (soportan mucha tensión al corte) y muy livianos, ideales para cumplir las especificaciones de peso. En el caso de impacto no presentan abolladuras, pero son susceptibles a deformación cuando se las expone a esfuerzos de compresión (escenario muy poco común). Sin embargo, este material es caro y no tiene mucha disponibilidad si se tratan de piezas aisladas, aunque sí se consiguen estructuras *gimbal* prediseñadas. Estas características, en especial la última, hacen que esta opción sea la más recomendable para el proyecto.

VI.1.2 Motores

VI.1.2.1 Tipo de motores

Actualmente se cuenta con distintos tipos de motores de corriente continua que podrían ser elegibles para esta aplicación. Los criterios empleados para determinar el motor ideal para este proyecto son los siguientes:

- Realimentación: un motor con realimentación existente facilita el diseño y acelera los tiempos de desarrollo.
- Tamaño: un motor de menor tamaño ayuda a cumplir las especificaciones.

- Torque: para garantizar un producto escalable, resulta favorable que el motor seleccionado cuente con versiones de mayor y menor torque, de forma tal de poder diseñar productos apuntados a estabilizar cámaras más grandes y pequeñas, respectivamente.
- Granularidad del posicionamiento: la precisión con la que se puede fijar la posición angular del motor no depende únicamente del software, sino también del motor en sí; considerando esto, ciertos motores no cumplirían este requerimiento.
- Durabilidad: el desgaste mecánico en engranajes es el principal enemigo de la durabilidad en un motor de corriente continua, y, como se mencionó en reiteradas ocasiones, la durabilidad es uno de los parámetros de mayor importancia en este proyecto; por esto, un motor de poco desgaste y, por ende, alta durabilidad será un formidable candidato.

Para el tipo de motor a emplear, dado el trabajo a realizar, se analizaron, básicamente, tres opciones:

Servo

La principal ventaja de este modelo es el hecho de que el sistema de realimentación ya viene integrado, esto lo hace además fácil de implementar y controlar. Sin embargo, este tipo de motores resultan muy voluminosos en comparación a los otros, y presenta un alto grado de vibraciones mecánicas y desgaste (en caso de requerir caja reductora). Este último aspecto se vuelve aún más crítico en servicio intermitente.

Motor CC con cerdas

Estos motores son baratos y compactos, pero requieren necesariamente de una caja de reducción para poder entregar el torque requerido para esta aplicación, con lo que se pierde una de las ventajas, la del volumen. Al igual que el servo, el desgaste mecánico, acentuado por el servicio intermitente de la aplicación, conforma una gran desventaja para este modelo.

BLDC

La principal ventaja de este modelo es el hecho de que, al no contener piezas mecánicas que ejercen fricción entre sí, el desgaste mecánico del sistema es nulo. Como desventajas, se pueden denotar el alto costo que tienen los motores de este modelo y la dificultad relacionada con desarrollar un sistema de control.

VI.1.2.2 Número de motores y potencia requerida

El número de motores, se refiere puntualmente a la rotación en *pitch*, donde se puede utilizar uno o dos motores. Este parámetro se evalúa junto con la potencia debido a que uno depende del otro: las alternativas son pocos motores de mucha potencia, o varios motores de poca potencia. Los criterios a utilizar para determinar esta variable son:

- Costo: dos motores de poca potencia no necesariamente cuesten más que un motor de alta potencia; es importante considerar esto, pero siempre recordando que el costo del producto no es tan importante como la durabilidad que este tenga.
- Consumo: la descarga de la batería se da, principalmente, por el consumo de los motores, por lo que minimizar este criterio ayuda a maximizar la duración de la batería.
- Durabilidad: dos motores trabajando en paralelo reparten mucho mejor la carga que uno sólo.
- Tamaño: nuevamente, es necesario cumplir esta especificación.

- Peso: nuevamente, es necesario cumplir esta especificación.

En cuanto al número de motores surgen dos alternativas:

Tres motores

Empleando sólo un motor para cada dirección radial (*yaw*, *roll*, y *pitch*) se minimizan los puntos de falla y se facilita el intercambio en caso de daño si se utiliza el mismo modelo para cada eje de rotación. Sin embargo, es más propenso al desgaste puesto que tiene que realizar más esfuerzo que en el caso de 2 motores en el eje *pitch*. Por otro lado, los motores más grandes tienen menor granularidad de la posición que los pequeños, debido a que necesitan menos polos para alcanzar más torque. El movimiento más accidentado se lo denomina *cogging*.

Cuatro motores

En esta configuración se emplean dos motores para *pitch*, lo que no es extraño ya que esta es la dirección más exigida, con el fin de repartir el desgaste entre dos motores. Esto es posible porque es el único par que puede tener simetría sin estorbar la lente de la cámara. A pesar de que esta configuración agrega un punto de falla, también agrega redundancia que puede resultar útil si se avería uno de los dos motores; si la cámara está bien balanceada, un motor puede ser suficiente para realizar el trabajo, aunque bajo un régimen de mayor exigencia. Este último punto debe cuidarse, ya que este mayor esfuerzo puede terminar dañando el motor restante.

Por otro lado, un motor más pequeño también implica un menor número de polos, que a su vez se traduce en una menor resolución angular. De todos modos, esta no es una desventaja importante porque la precisión puede mejorarse notablemente desde software, como se verá en la sección Ingeniería del Detalle del Software.

VI.1.3 Control remoto

En esta sección se analizan los diversos tipos de controles remotos que pueden utilizarse y se los compara según los siguientes criterios:

- Cantidad de señales de control: se deben poder controlar al menos seis variables del sistema, por lo cual se requieren al menos seis señales de control.
- Protocolo de comunicaciones: debe poder permitir comunicación bi-direccional y ser interpretado por el microprocesador fácilmente.
- Alcance: se determinó mediante la entrevista que el alcance debe ser entre 50 y 100 metros.
- Indicación al Usuario: debe tener un sistema que permita informar al usuario cual es el estado del sistema.
- Costo: por motivos obvios, es preferible uno de bajo costo.
- Disponibilidad: debe poder conseguirse en el país.

Controles RC

Estos se encuentran disponibles en una amplia gama de precios y prestaciones, sin embargo todos presentan características similares ya que siguen un estándar que fue optimizado para la industria de modelos a escala. Poseen al menos seis señales de control, las cuales pueden ser analógicas o digitales. Este protocolo

está hecho para el control directo de servos y motores, y consiste en enviar señales (en la mayoría de los casos uni-direccionales) moduladas en FM de 72MHz o PPM de 2,4GHz. La salida del dispositivo receptor consiste en señales PWM de 500Hz. El alcance de estos controles es excepcional, pudiendo ser de hasta un kilómetro. Si bien algunos modelos poseen un *display*, estos se usan para monitorear las señales de control del Joystick. El precio de estos controles varía a entre AR\$1000 y AR\$20000, y su disponibilidad es muy limitada.

Joystick inalámbrico de Xbox® 360

Estos son controles inalámbricos de la popular consola de videojuegos de Microsoft®. Estos poseen 19 señales de control, de los cuales 6 son analógicos y el resto digitales. El control se comunica con el receptor a través de señales RF de 2,4GHz, usando un protocolo de comunicaciones solo disponible para desarrolladores de videojuegos. El receptor funciona mediante protocolo USB HID. Su alcance máximo es de 50 metros. No posee *display* integrado. Su precio se encuentra entre AR\$200 y AR\$600, y se encuentra disponible en todo el país en cualquier casa de electrodomésticos, de computación, o de videojuegos.

Joystick inalámbrico de PlayStation® 3

Estos son controles inalámbricos de la popular consola de videojuegos de Sony®. Estos poseen 19 señales de control, de los cuales 6 son analógicos y el resto digitales. El control se comunica con el receptor a través de señales Bluetooth clase 3, usando un protocolo de comunicaciones libre. El receptor debe poder ser capaz de recibir e interpretar señales Bluetooth con perfil HID. Su alcance máximo es de 50 metros. No posee *display* integrado. Su precio se encuentra entre AR\$200 y AR\$600, y se encuentra disponible en todo el país en cualquier casa de electrodomésticos, de computación, o de videojuegos.

Joystick personalizado

Puede adecuarse a los requerimientos que sean necesarios.

VI.1.4 Elección de una solución

Tipo de estructura

Habiendo analizado las alternativas, se concluye que el *gimbal* es la mejor opción para la estructura del estabilizador.

Material de construcción

Evaluando estas dos alternativas, y considerando que no se pretende vender el producto sino alquilarlo, la segunda opción resulta más favorable. A pesar de esto, una estructura de fibra de carbono sería poco viable si se optara por un diseño propio para la estructura, principalmente debido al difícil acceso al material en el país, por lo que también se opta por adquirir una estructura pre-diseñada. Esta segunda decisión se hubiera tomado de todas formas debido a que el grupo de trabajo carece de los conocimientos o experiencia necesarios para diseñar una estructura con funcionalidad y durabilidad comparable a una pre-diseñada.

Tecnología de motores

Luego de considerar las tres opciones, se determinó emplear motores de tipo BLDC debido a dos factores: el primero, es que el mercado se ha acostumbrado a sistemas *gimbal* empleando motores BLDC; y el segundo, de mayor peso, es el hecho de que al proponer un sistema de comercialización por alquiler, el

costo del producto no es tan importante como su durabilidad, y es en este aspecto que el BLDC se posiciona por encima de los otros motores, siendo la elección más clara.

Número de motores

Si bien la segunda opción agrega un componente adicional que puede fallar, resulta conveniente extender la durabilidad del sistema de *pitch*, incluso si esto implica un motor adicional. Cabe resaltar que, debido a que la configuración de trabajo de estos dos motores permitiría que sus torques se adicionen, los motores de *pitch* trabajando de a dos requerirían un torque mucho menor que los otros.

Control remoto inalámbrico

Debido a que ningún control remoto disponible comercialmente se adecuaba a todos nuestros criterios, se decidió el desarrollo e implementación de nuestro propio control remoto inalámbrico que posea siete señales de control (tres analógicas y cuatro digitales), que utilice el protocolo de comunicaciones RF, y que posea un LED de estado RGB. Esto permite un grado de control mayor sobre modos de funcionamiento deseados.

Si bien el precio estimado del desarrollo se acerca mucho al del control inalámbrico de PLAYSTATION® 3, los módulos RF que se encuentran disponibles en el mercado funcionan solo mediante el perfil SPP.

Resumen

Elemento	Detalle	Alternativa Seleccionada	Ventajas
Estructura	Forma Mecánica	Gimbal	<ul style="list-style-type: none"> • Manejo intuitivo y facilidad de uso. • Permite alinear el eje de acción de los motores con el centro de masa de la cámara. • Independiza mecánicamente los motores.
	Material de Construcción	Fibra de carbono	<ul style="list-style-type: none"> • Alternativa más liviana y resistente. • Presentan muchas estructuras pre – diseñadas.
Motores	Tecnología	BLDC	<ul style="list-style-type: none"> • No contiene partes móviles, en consecuencia no tiene desgaste mecánico. • Buena relación peso/potencia. • Permiten sistemas de control de alta precisión
	Cantidad de Motores	Cuatro Motores	<ul style="list-style-type: none"> • Permiten un menor esfuerzo, y menor desgaste por uso que la configuración de tres motores para el eje de pitch.
Control Remoto	Control Remoto	Control Personalizado	<ul style="list-style-type: none"> • Puede ser elaborado para adaptarse de la mejor forma posible a las especificaciones requeridas.

Figura VI.1.3.: Resumen de las alternativas de diseño

VI.1.5 FMEA

Subdivisión en bloques

Se identifican seis bloques principales que componen el producto: el sensor, el microprocesador, los drivers, los motores, la estructura, y el control remoto. Cada uno de estos bloques puede fallar en formas diferentes y con consecuencias de magnitud y gravedad variadas.

A continuación se presenta el FMEA que incorpora un análisis de las causas y las acciones preventivas y reactivas cuando ocurren los siniestros.

CAPÍTULO VI. ANÁLISIS DE FACTIBILIDAD

Módulo	Modo de falla	Efecto	SVD	Causa	OCR	Controles de prevención	Controles de detección	DTC	RPN	Acciones
Sensor	Presenta offset	La cámara queda estabilizada en una posición indeseada	2	Falta calibración	1	Calibrar el sensor cada vez que se usa el equipo	Medir la salida del sensor cuando está completamente quieto	1	2	Calibrar el giroscopio y el acelerómetro
	Produce salida nula	La cámara queda fija en una posición	3	Desconexión de los cables de I2C	1	Utilizar conectores con encastre mecánico	Realizar un sondeo de dispositivos I2C por software	3	9	Reconectar los cables de I2C
	Produce salida no nula errática			Falla interna del sensor	1	Verificar la salida del sensor antes de su uso	Medir las señales de I2C en busca de ausencia de pulsos	3	9	Cambiar el sensor
		La cámara se mueve erráticamente	9	Interferencia en el canal I2C	1	Hacer mediciones controladas cuando el sensor está quieto	Medir las señales de I2C en busca de picos de tensión	3	9	Usar un <i>level shifter</i>
MCU	Se "cuelga"			Falla interna del sensor	1	Verificar el funcionamiento del sensor antes de su uso	Medir las señales de I2C en busca de ausencia de pulsos	3	27	Cambiar el sensor
	Se apaga	La plataforma queda completamente quieta, con los periféricos apagados	9	Interferencia de las señales de I2C	2	Agregar condición de <i>timeout</i> para la lectura y escritura	Mostrar mensaje de <i>timeout</i> en pantalla	2	36	Reiniciar el MCU
				Sobrepico de tensión de alimentación	2	Agregar diodos de protección	Medir la tensión de alimentación	2	36	Cambiar el MCU
Drivers	Se produce cortocircuito en la señal de control	Se queman los puertos PWM del MCU		Las pistas de señal de control se tocan				2	3	Separar las pistas con una cuchilla
	Se produce cortocircuito en la señal de salida	La batería se descarga violentamente (riesgo de explosión severa)	8	Las pistas de señal de control se tocan	1	Diseñar el PCB con pistas debidamente separadas	Examinar las pistas con un microscopio		5	
				Los MOSFETs se quemaron por corriente	1	Poner un circuito limitador de corriente	Medir la corriente de salida	5	40	
	Deja de generar señal de salida	Los motores se detienen	9	Los MOSFETs se quemaron por tensión	1	Poner un circuito limitador de tensión	Medir la tensión de salida	5	45	Cambiar el driver
	La señal de control queda fija en estado activo	Los motores consumen mucha corriente y se calientan	6	La señal de enable no desabilita la salida	1	Utilizar resistores de <i>pull-down</i> en el enable (falla apagado)	Medir las señales a la salida del driver	3	18	
Motores	Se detienen y quedan apagados	La plataforma no opone resistencia frente al movimiento	10	Se quemaron los bobinados del motor	1	Poner un circuito limitador de corriente	Medir la corriente de salida	20	30	Cambiar el bobinado del motor
Estructura	Se produce una fractura	Caen la cámara o alguno de los motores (depende el lugar de la fractura)	10	Se excedió el peso máximo permitido	1	Hacer chequeos periódicos para evaluar la integridad de la estructura		10	100	Reemplazar las partes dañadas
	Se deforma	Balanceo defectuoso y nocivo para el resto de los componentes mecánicos	2		3		Examinar la estructura en busca de fallas		5	
Control remoto	Se corta la comunicación	La plataforma no obedece al control remoto	1	Se dañó el transmisor y/o el receptor	1	Hacer pruebas periódicas de transmisión de datos	Medir el canal SCI en busca de señal nula	1	1	Reemplazar el transmisor y/o receptor
	El display no muestra caracteres	No pueden visualizarse los cambios	1	Se golpeó el control remoto	1	Manejar el control con cuidado	Encender el control remoto y verificar que muestre caracteres	1	1	Reemplazar el display
				Se quemó el display	1	Agregar diodos de protección por sobretensión	Medir continuidad entre cada uno de los terminales	1	1	

Figura VI.1.4.: Modos de falla del gimbal

Modos de falla

Los modos de falla para los distintos bloques son:

- Sensor

- Presenta offset: Este problema se manifestaría como un error de valor constante en la salida del sistema de sensado de posición. El efecto producido por este error sería un desalineamiento del plano referencia de estabilización, por lo que la cámara seguiría estabilizándose pero respecto de una posición inclinada. La detección y corrección de este problema puede efectuarse de forma externa al microprocesador, mediante software en el momento de la calibración. Este problema, así como los dos que siguen, serían causados por desgaste acumulado o fallas eléctricas.

- Produce salida nula: Este problema se manifestaría como una salida nula en todo momento de la medición. El efecto producido por este error sería que el microprocesador no percibiría ninguna perturbación en la cámara, asumiendo que está perfectamente estabilizada en todo momento, resultando en una cámara fija en una posición por los motores, sin estabilización. La detección y corrección de este problema puede efectuarse mediante un módulo de seguridad implementado en el software del microprocesador.
- Produce salida no nula errática: Este problema se manifestaría como una salida fija y distinta de cero en todo momento de la medición. El efecto producido por este error sería que el microprocesador percibiría una perturbación en la cámara en todo momento, independientemente del movimiento o la posición de los motores; de esta forma, el microprocesador buscaría estabilizar la cámara, lo que, en estas circunstancias, produciría un movimiento constante y descontrolado de los motores, teniendo efectos potencialmente destructivos para el producto. La detección y corrección de este problema puede efectuarse mediante un módulo de seguridad implementado en el software del microprocesador

■ Mircoprosesador

- Se “cuelga”: en caso de que el microprocesador deje de funcionar de forma repentina, pueden darse dos situaciones. Por un lado, los motores podrían apagarse y el sistema quedaría como si estuviera desconectado de la alimentación, lo cual no tendría consecuencias negativas para el producto y requeriría sólo un reinicio para solucionarse. Por otro lado, los motores podrían seguir girando sin una forma de decodificar las mediciones de posición, lo que resultaría en un escenario similar al de “produce salida no nula errática”. El segundo escenario es mucho más peligroso que el primero, pero, por motivos obvios, ambos son imposibles de solucionar por software. En lugar de esto, se busca erradicar la causa del problema, que sería un problema con la alimentación del microprocesador.

■ Drivers

- Se queman los MOS: se identifican dos casos en este modo de falla, siendo estos que los MOS se quemén por sobre-tensión en inversa, y que se quemén por corriente.
 - Tensión: en caso de superarse la tensión en inversa nominal del MOS, el mismo realiza un circuito abierto; esto implica que la señal de control que se modula a través del *half-bridge* no podría alcanzar los niveles requeridos, por lo que el comportamiento del sistema sería uno totalmente errático. Al no poder predecir con exactitud el comportamiento del sistema bajo esta falla, los efectos podrían ser potencialmente destructivos para el producto. Este inconveniente se detectaría a través de pruebas de prototipo, es decir previas a la finalización del producto, debido a que un problema circuital de esta especie no debería existir en el sistema final.
 - Corriente: en caso de superarse la corriente directa nominal del MOS; el mismo realiza un corto circuito; esto implica que la batería quedará cortocircuitada. La descarga súbita y descontrolada de corriente que esto produciría tendría efectos destructivos para la batería y, si el efecto resultase explosivo, para el producto en sí. Al igual que el escenario anterior, pruebas extensivas deberían ser capaces de identificar la causa del problema y permitir su definitiva y permanente corrección.
- Cortocircuito: en caso de que los dos MOS de un mismo *half-bridge* se enciendan a la vez, la batería quedaría en un cortocircuito con sí misma, lo que causaría, al igual que en el escenario anterior, una violenta descarga de corriente con las mismas consecuencias. Afortunadamente,

este escenario se soluciona sencillamente modificando el software que maneja el encendido de los MOS en el driver, y su detección sería, al igual que todos los escenarios para el *driver*, fácilmente detectable a través de pruebas de prototipo.

■ Motores

- Dejan de funcionar: en caso de que los motores dejen de funcionar, el sistema estaría como desconectado de alimentación, lo cual no tendría consecuencias negativas para la estructura y solamente habría que reemplazar los motores. Nuevamente, una falla de esta índole podría ser causado, casi unívocamente, por un problema con la alimentación.
- Rotura: se prevé la posibilidad de que una rotura física, no eléctrica, en los motores ocurra. En este caso, probablemente, estos queden fijos en una posición determinada, lo cual deja al sistema en condiciones similares al caso “Zero output”. Este modo de falla no tiene consecuencias potencialmente destructivas para el producto, más allá de los motores que habría que reemplazar.

■ Estructura

- Fractura: el continuo desgaste que conlleva el uso de la estructura podría, eventualmente, producir una fractura. Este modo de falla, dependiendo de dónde se produzca, podría causar un daño irreparable en la cámara a estabilizar, siendo así el escenario más peligroso de todos. Sin embargo, bajo las condiciones de trabajo estipuladas para el producto, sería difícil producir un desgaste tal que genere este modo de falla, por lo que la causa más probable sería un exceso en las condiciones de trabajo. Es prudente resaltar que este tipo de fracturas generalmente nacen de lesiones menores en la estructura que actúan como concentradores de tensiones, por lo que la temprana detección de estas lesiones podría ayudar en la prevención de una fractura mayor.
- Deformación: este modo de falla tiene mayores probabilidades de darse sin exceder las condiciones de trabajo que el anterior, pero a su vez no trae aparejado consecuencias de la misma magnitud, sino mucho menores. En el peor de los casos, la cámara quedaría estabilizada respecto de un plano inclinado (modo de falla “Offset”), que es fácil de solucionar. Este modo de falla, sin embargo, indicaría que se ha superado la durabilidad del producto y, en dicho momento, se debería evaluar si la vida útil del producto puede extenderse mediante algún medio, o si se debe reemplazar.

■ Control Remoto

- Corte de comunicación: en este caso, el sistema de estabilización debería seguir funcionando normalmente en la última configuración guardada o, en caso de que la falla se produzca durante el proceso de configuración, que el sistema de estabilización vuelva a la ultima configuración guardada. La ruptura de la antena se encuentra contemplada dentro de esto modo de fallas, ya que de romperse, la comunicación se perderá.
- Falla de alguna señal de control: en ese caso, las funciones que requieran el uso de dicha señal de control quedarían imposibilitadas, pero no afectarían directamente al sistema de estabilización.
- Desgaste o sobrecarga de baterías: En este caso el control remoto se apagaría, cesando las comunicaciones, con lo cual ocurriría un caso similar a una falla de comunicaciones. De producirse una sobretensión, podrían quemarse los componentes más sensibles del control remoto (el regulador de tensión, el microprocesador, o el módulo RF)
- Rotura del LED de estado: En este caso, el LED no se encendería o solo encendería algunos de los colores, en ambos casos, no afectaría al sistema de estabilización.

VI.2 Factibilidad de tiempos

VI.2.1 Planificación

PERT

La planificación del proyecto está dividida en 2 etapas: la primera consiste en el desarrollo de un primer prototipo funcional y la segunda está dedicada a crear un producto terminado. A continuación se presenta un resumen de las tareas a realizar, seguidas por el diagrama PERT. Una consideración muy importante es que los días se toman como jornadas de 5 horas ininterrumpidas, menor a las 8 usuales debido a los obstáculos que presenta la facultad: parciales, trabajos prácticos y clases. Es por esta razón que las tareas tienen, por momentos, duraciones demasiado largas.

De las 5 tareas iniciales, 4 de ellas son de adquisición de dispositivos que conforman el *gimbal* que, debido a las barreras a la importación de Argentina, tienen tiempos extensos y con mucha dispersión en las estimaciones de duración esperada. Por otro lado, el desarrollo de la interfaz entre el MCU y la computadora es una tarea que tiene corta duración y que sólo requiere una idea aproximada del diseño deseado para la versión final.

Luego de disponer de los materiales necesarios, nos encontramos con un cuello de botella, que es la adaptación al MCU elegido. Esto implica el aprendizaje de la interfaz de desarrollo, familiarización con la arquitectura y acostumbramiento a la metodología para hacer y ejecutar programas sobre la plataforma elegida. Todas las tareas que la preceden implican programar funciones en el procesador o dependen directamente de programas para realizar el nexo entre el *hardware* y el *software*. Cabe destacar que el tiempo elegido para esta tarea también tiene grandes fluctuaciones, ligadas principalmente a cuán intuitiva es la IDE, el grado de documentación y el soporte disponible en Internet.

Una tarea muy importante es la fusión de datos del sensor, proceso por el cual se combinan las mediciones del acelerómetro y giroscopio para brindar un ángulo de inclinación preciso en el tiempo más corto posible. Debido a que el desempeño general del *gimbal* está determinado por cuán precisa es la medición, esta tarea es de máxima importancia, por lo que se le dio una duración considerable. La interfaz SCI se encarga de transmitir datos del MCU a la computadora y es muy útil para ver los cambios de inclinación en tiempo real. Aunque no está explícito en el PERT por un tema de simplicidad y orden, es posible que se necesite un ajuste de parámetros en la detección del ángulo, que deberá realizarse junto con la estabilización. Por otra parte, también se debe tener un control preciso sobre los motores, que a su vez depende de la exactitud de las especificaciones.

Una vez que se consigue la estabilización se pasa a incluir el comando inalámbrico del *joystick* en el lazo de control, para contar con la modalidad de encuadre remoto. Finalmente, se utiliza el controlador terminado para estabilizar la estructura *gimbal*, proceso en el cual se prueban distintas cámaras y se corrigen algunos problemas menores que puedan surgir. En este punto, se cuenta con un prototipo funcional del estabilizador y se puede identificar el camino crítico en rojo.

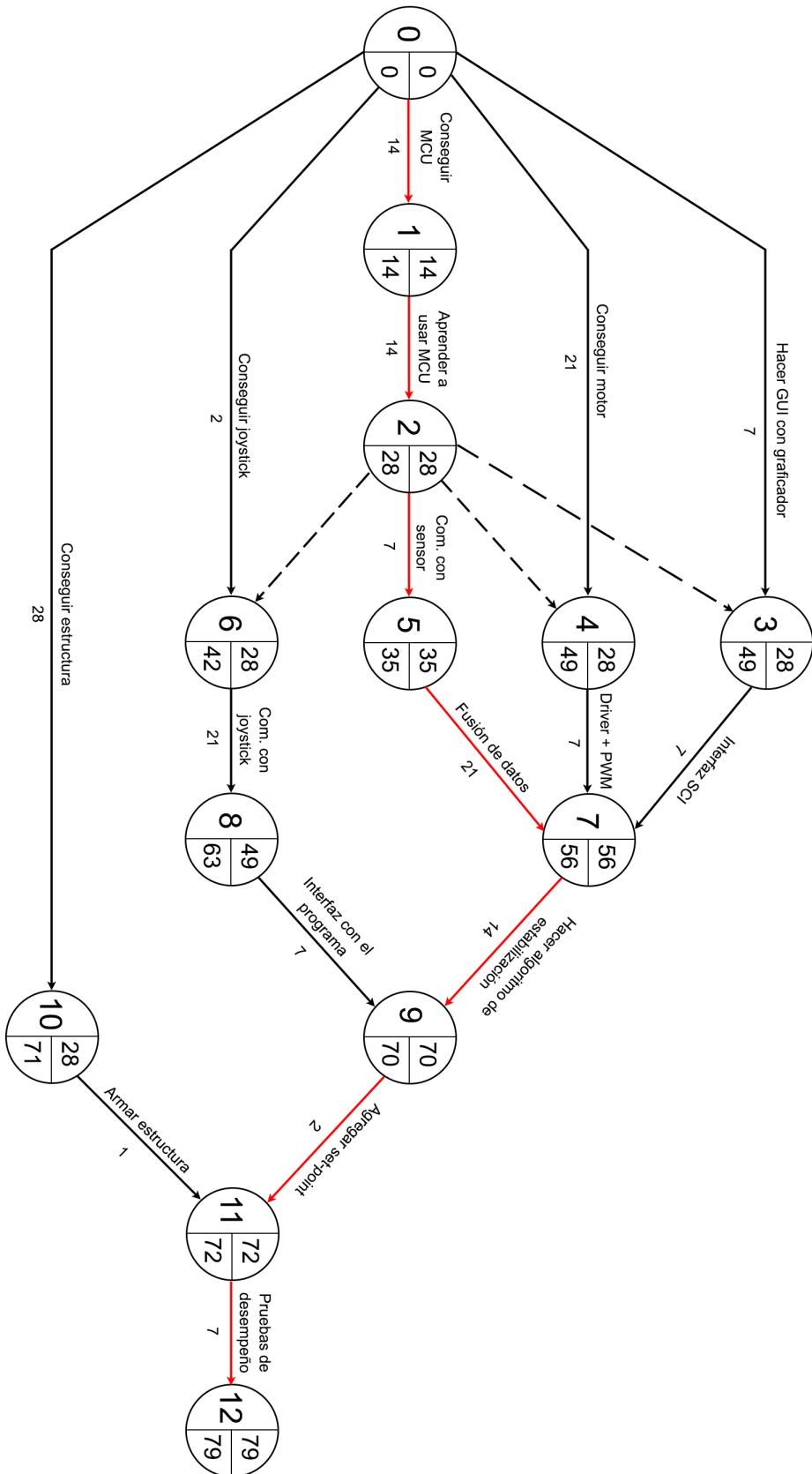


Figura VI.2.1.: Diagrama de flujo PERT de la Etapa I

La segunda etapa es mucho más lineal que la primera porque implica el diseño de la placa y el gabinete y su posterior materialización. Luego del ensamblado se introduce el nuevo código, que se nutre de las ideas

que surgieron durante la realización del proyecto, para luego realizar las últimas pruebas de desempeño y calidad. En color rojo se marca el camino crítico, que depende de la manufactura y no de la programación.

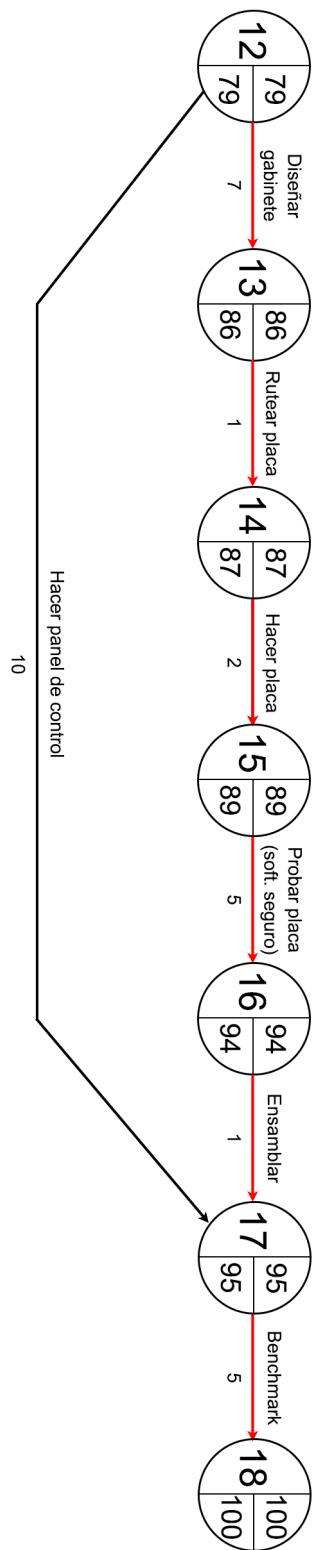


Figura VI.2.2.: Diagrama de flujo PERT de la Etapa II

ID	Tarea	Optimista	Más probable	Pesimista	Esperada	Comienzo temprano	Finalización tardía
0 - 0	Inicio	0	0	0	0	0	0
0 - 1	Conseguir MCU	10	13	20	14	0	0
0 - 3	Hacer GUI + graficador	5	6	10	7	0	0
0 - 4	Conseguir motor	18	21	27	22	0	0
0 - 6	Conseguir joystick	1	2	3	2	0	0
0 - 10	Conseguir estructura	20	28	35	28	0	0
0 - 1	Aprender a usar MCU	10	13	20	14	14	14
3 - 7	Interfaz con SCI	4	6	12	7	28	49
4 - 7	Driver + PWM	5	7	9	7	28	49
2 - 5	Comunicación con el sensor	3	6	12	7	28	28
5 - 7	Fusión de datos	15	21	25	21	35	35
6 - 8	Comunicación con el joystick	15	21	27	21	28	42
8 - 9	Interfaz con el programa	4	6	11	7	49	63
7 - 9	Hacer algoritmo de estabilización	10	14	18	14	56	56
9 - 11	Agregar set-point	1	2	3	2	70	70
10 - 11	Armar estructura	0,5	1	1,5	1	28	71
11 - 12	Pruebas de desempeño	4	7	8	7	72	72
12 - 13	Diseñar gabinete	4	6	12	7	79	79
12 - 17	Hacer panel de control	4	10	15	10	79	79
13 - 14	Rutear placa	0,5	1	1,5	1	86	86
14 - 15	Hacer placa	0,5	1	2	2	87	87
15 - 16	Probar placa	2	4	8	5	89	89
16 - 17	Ensamblar	0,5	1	1,5	1	94	94
17 - 18	Benchmark	2	5	7	5	95	95
18 - 18	Fin	0	0	0	0	100	100

Figura VI.2.3.: Tiempos estimados y calculados para las tareas PERT

Monte Carlo

Debido a que la estimación de tiempo de finalización del proyecto obtenida mediante PERT suele ser poco representativa, se decidió realizar una simulación Monte Carlo a partir de los datos de la Tabla VI.2.3. Para eso se modeló el tiempo esperado de cada tarea con una distribución triangular cuyos parámetros son los tiempos optimista, más probable y pesimista de PERT, y se realizaron 50,000 ejecuciones de la simulación. La Figura VI.2.4 muestra que la duración más probable del proyecto son 120 días, es decir, un 20% mayor que la predicción obtenida anteriormente. De hecho, sólo el 10% de las ejecuciones indicaron que la duración es de 100 días, de donde podemos concluir que la simulación de Monte Carlo provee una estimación mucho más fiel a la realidad.

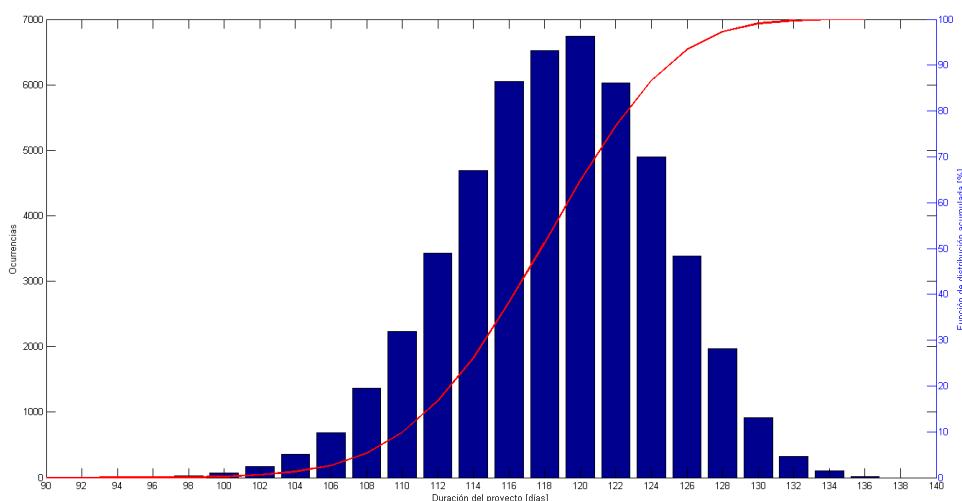


Figura VI.2.4.: Resultados de la simulación de Monte Carlo

VI.2.2 Programación

La planificación del proyecto utilizando PERT es útil para ver las dependencias de las tareas y asignar recursos, pero no es efectiva para hacer un seguimiento más detallado del proyecto. Para solucionar esto, se realizó el diagrama de Gantt que se muestra en la Figura VI.2.5, donde podemos ver, nuevamente, el camino crítico resaltado en rojo. Las duraciones de las tareas son los valores esperados de PERT por lo que, en principio, la duración del proyecto sería 100 días. Sin embargo, gracias a la simulación Monte Carlo sabemos que las tareas pueden retrasarse, por lo que será necesario ajustar el Gantt a medida que se progrese en el proyecto.



Figura VI.2.5.: Resultados de la simulación de Monte Carlo

VI.3 Factibilidad económica

A continuación se presenta el modelo de negocios de Framing S.A.

VI.3.1 El Negocio

Introducción

El objetivo de Framing S.A. es convertirse en el principal proveedor de soluciones de estabilización para filmación en Argentina.

La empresa tendrá una estructura de Sociedad Anónima, conformada por Martín Biondi Sarra, Rodrigo Lorenzo Leal, Joan Martorell y Pablo Remedi.

Origen

El negocio surge desde el desarrollo de una nueva tecnología, los estabilizadores basados en motores *brushless*, mejor conocidos como *Brushless Gimbal*. El estricto control sobre las importaciones en el país, como también el alto costo de estos equipos hace que su importación a la Argentina sea difícil, y poco rentable. Por otro lado, debido a que las partes que conforman el sistema están al alcance en Argentina, el desarrollo de una versión de bajo costo que cubra el mercado local es una posibilidad real.

VI.3.2 El Servicio

Framing S.A. provee soluciones de estabilización para cámaras de filmación de hasta 6kg. Dicha solución consta de un sistema tipo *Brushless Gimbal*, acompañada de un control remoto para su manejo.

La Competencia

En la actualidad en Argentina los únicos sistemas de estabilización equivalentes son los implementados con Steadicam. Estos sistemas funcionan con contrapeso, por lo que son pesados y requieren de la habilidad del operador para su correcto manejo.

En Argentina los sistemas tipo Steadicam son manejados por técnicos certificados los cuales no solo son dueños de los equipos, sino también son los únicos calificados para su operación.

Estos sistemas, si bien cumplen con su tarea, obligan a la producción a exponerse al error humano del operario, y adicionalmente no permiten que el director general o de fotografía tengan control sobre el encuadre de la toma.

Nuestra Ventaja

Debido a que no requiere ningún tipo de contrapeso, los sistemas *Brushless Gimbal* son mucho más livianos que los tipo Steadicam. Adicionalmente debido a que la estabilización es realizada de forma inteligente y automatizada, el mismo camarógrafo de la filmación puede realizar la toma en movimiento.

Finalmente, con el uso de un control inalámbrico se puede controlar la posición de la cámara de manera remota, pudiendo corregir el encuadre en tiempo real, lo cual no es posible con otro tipo de sistemas.

VI.3.3 El Mercado

El mercado a cubrir es conformado por dos grandes agentes: producciones cinematográficas y producciones publicitarias.

El Mercado del Cine en Argentina

A continuación se listan la cantidad de películas desarrolladas en Argentina que fueron estrenada los últimos 4 años. Los datos fueron extraídos del Anuario de la Industria del Cine, elaborado por el Instituto Nacional de Cine y Artes Audiovisuales (INCAA).

Año	Cantidad de Largometrajes
2010	110
2011	119
2012	145
2013	166

Figura VI.3.1.: *Cantidad de largometrajes desarrollados en Argentina en los últimos 4 años*

El Mercado de la Publicidad en Argentina

Se solicitó un estudio de mercado con el objetivo de estimar la cantidad de publicidades filmadas en Argentina. El estudio fue realizado de forma honoraria por el presidente de la Agencia Wunderman Argentina, Juan Pablo Jurado. Dicho estudio concluyó con lo siguiente: “La averiguación que hicimos, es que se filman aproximadamente 1,520 comerciales por año en la Argentina. En esta cantidad, vas a encontrar todo tipo de comerciales, de presupuestos que pueden rondar los 5 millones de pesos hasta comerciales de 30,000 pesos.”

Mercado Objetivo

Proyectando que para 2015 las producciones cinematográficas no descenderán de las 160, y que las producciones de publicidad permanecerán relativamente constantes, se apunta a cubrir el 10% del mercado de cine y el 1% del mercado de la publicidad dentro de los primeros dos años, con posibilidades de expansión a 20% y 5% respectivamente a lo largo de 5 años.

Esto se traduce a una cantidad de filmaciones de 16 y 15 respectivamente para un total anual de 31 filmaciones en los primeros dos años y un total de 107 filmaciones anuales a los 5 años.

VI.3.4 Plan de Marketing

Los objetivos de Framing S.A. al nivel marketing son los siguientes:

- Establecerse como alternativa a Steadicam en el mercado del cine y la publicidad en Argentina.
- Obtener un volumen de ventas que permita establecer un negocio rentable.
- Establecer una relación positiva con nuestros clientes.

Para lograr dichos objetivos nos enfocaremos en los siguientes aspectos:

El Servicio

Consistirá en el alquiler de sistemas tipo *Brushless Gimbal*. El servicio comenzará a funcionar en una primera prueba piloto en el verano de 2015.

Si bien en principio no se contempla ofrecer otro tipo de servicios, no se descarta la posibilidad de desarrollar nuevos productos siempre y cuando comparten mercado con nuestro servicio.

El Lugar

Framing S.A. se establecerá dentro de los límites de la Ciudad Autónoma de Buenos Aires, como también en el conurbano de la Provincia de Buenos Aires.

Se escogió esta localidad debido a que las principales productoras de cine y publicidad del país están establecidas en dicha región. Adicionalmente todos los miembros de la Sociedad Anónima residen en esta zona.

El Precio

Se establecerá un precio inicial de ARS\$5,000 por jornada de uso del equipo, contemplando que cada jornada será de 8 horas.

Averiguaciones realizadas permitieron investigar que el servicio de un técnico de Steadicam cuesta entre ARS\$7,000 y ARS\$8,000 por día de trabajo. Consideramos que un precio inicial de ARS\$5,000 por jornada es una diferencia significativa para hacer aún más atractivo el producto, y es un precio que nos permitirá establecer un negocio rentable.

La Promoción

La promoción se hará de forma directa, esto es, ofreciendo nuestro servicios a productoras de cine y publicidad. Haremos esto a través de correo electrónico, reuniones y demostraciones en vivo en las oficinas de los posibles clientes.

Adicionalmente se hará promoción del servicio en conferencias como la Exposición de Equipamiento Profesional y Servicios para Radio, Televisión, Cable, Cine, Satélite, Iluminación & Audio organizada por la La Cámara Argentina de Proveedores y Fabricantes de Equipos de Radiodifusión (CAPER).

VI.3.5 Análisis Financiero

Los Costos

Se realizó el análisis de los costos que genera la empresa, separándolos en cuatro tipos, que serán explicados a lo largo de esta sección:

- Inversión Inicial.
- Costos Fijos.
- Costos Variables.
- Costos de Capital de Trabajo.

La inversión inicial se considera como el costo de las partes que componen el proyecto, así como el de crear una página web para la promoción del negocio. En el siguiente cuadro pueden verse los costos de las partes que componen la Inversión Inicial:

Inversiones		
	U\$D	AR\$
Microporcesador	5	41
Motores	270	2190
Bateria	150	1217
Cargador de bateria	100	811
Estructura	1175	9529
Joystick	50	406
Página web	617	5000
Total	2367	19193

Figura VI.3.2.: Cuadro de costos de la Inversión Inicial del proyecto

Cabe destacar que en este cuadro y los posteriores se resaltó en verde la moneda utilizada para realizar las transacciones. En este caso, debido a que los componentes no se producen en el país, se tuvieron que abonar en Dólares Americanos, mientras que la única inversión que ha de abonarse en Pesos Argentinos es la creación de la página web, que puede ser realizada un tercero o por nosotros mismos (en este caso, se debe considerar el costo de oportunidad que significa destinar tiempo a esta tarea).

Con respecto a los gastos fijos, se consideró que se alquilará una oficina de $40m^2$ en la zona de Puerto Madero, que ha de cumplir también el rol de lugar de mantenimiento y desarrollo de nuevos equipos. Los costos del mantenimiento, tanto del equipo, como la oficina, se juntaron en una sola casilla por razones de conveniencia. Se distinguen las categorías de servicios telefónicos e informáticos, debido a que este último representa los costos de mantenimiento de la página web. Por último, por razones de conveniencia se juntaron los servicios básicos en la casilla de “otros servicios”. En el siguiente cuadro pueden verse los costos fijos que componen el proyecto:

Costos Fijos Mensuales		
	U\$D	AR\$
Mantenimiento	308	2500
Servicios Telefónicos	25	200
Servicios Informáticos	123	1000
Alquiler de Oficina	617	5000
Otros Servicios(Luz, Agua y Gas)	105	850
Total	1178	9550

Figura VI.3.3.: Cuadro de costos fijos del proyecto

Un costo variable es aquel que se modifica de acuerdo a variaciones del volumen de producción. Debido a que la meta del proyecto es alquilar el *gimbal*, este proyecto no presenta costos variables en el sentido estricto. Más bien, se incurrirá en gastos de producción cuando la demanda de equipos sobrepase la disponibilidad, que se verá reflejado en saltos de inversión.

Los costos de Capital de Trabajo son los sueldos de los empleados. Como se cuenta sólo con los cuatro miembros en este proyecto, que figuran como asociados, los costos de capital de trabajo son cuatro sueldos. En el siguiente cuadro pueden verse los costos de Capital de Trabajo que componen el proyecto:

Capital de trabajo		
	U\$D	AR\$
Asociados	2466	20000
Total	2466	20000

Figura VI.3.4.: Cuadro de costos de Capital de Trabajo del proyecto

Análisis económico-financiero

En la siguiente tabla puede verse el cuadro de resultados del proyecto realizado a un año, teniendo en cuenta que se alquila el equipo por dos semanas de filmación por mes. Para realizar este análisis, se tuvo en cuenta que la inversión inicial se amortiza en un año.

Cuadro Resultados	0	1	2	3	4	5	6	7	8	9	10	11	12
Ingresos	0	56000	56000	56000	56000	56000	56000	56000	56000	56000	56000	56000	56000
Costos Fijos	0	-9550	-9550	-9550	-9550	-9550	-9550	-9550	-9550	-9550	-9550	-9550	-9550
Costos Variables	0	0	0	0	0	0	0	0	0	0	0	0	0
Amortización	0	-1599	-1599	-1599	-1599	-1599	-1599	-1599	-1599	-1599	-1599	-1599	-1599
EBIT	0	44851	44851	44851	44851	44851	44851	44851	44851	44851	44851	44851	44851
Impuesto a las ganancias	0	-21528	-21528	-21528	-21528	-21528	-21528	-21528	-21528	-21528	-21528	-21528	-21528
UN	0	23322	23322	23322	23322	23322	23322	23322	23322	23322	23322	23322	23322

Figura VI.3.5.: Cuadro de resultados a un año

A partir del cuadro resultados, se calculó el cuadro de Cash Flow del proyecto, dando como resultado el siguiente:

Flujo de fondos	0	1	2	3	4	5	6	7	8	9	10	11	12
Ingresos	0	56000	56000	56000	56000	56000	56000	56000	56000	56000	56000	56000	56000
Costos	0	-9550	-9550	-9550	-9550	-9550	-9550	-9550	-9550	-9550	-9550	-9550	-9550
Inversión	-19193	0	1	2	3	4	5	6	7	8	9	10	11
Capital de trabajo	0	-20000	-20000	-20000	-20000	-20000	-20000	-20000	-20000	-20000	-20000	-20000	-20000
Impuesto a las ganancias	0	-21528	-21528	-21528	-21528	-21528	-21528	-21528	-21528	-21528	-21528	-21528	-21528
Cash Flow	-19193	4922	4923	4924	4925	4926	4927	4928	4929	4930	4931	4932	4933

Figura VI.3.6.: Cuadro de Cash Flow a un año

Resultados

A partir del cuadro de flujo de caja mostrado en el cuadro anterior, se calculó el Valor Actual Neto (VAN) con una tasa de interés efectiva anual del 30 %, dando como resultado un valor positivo de \$15710. También se calculó la Tasa Interna de Retorno (TIR) en un 24 %. Por último, se obtuvo un Período de Retorno de la Inversión de 10 meses. Habiendo hecho el análisis de factibilidad y considerando todos los factores, se concluye que el proyecto es viable económicoamente. En la siguiente figura se pueden ver estos resultados.

VAN	\$ 15.710
TIR	24%
PRI	10

Figura VI.3.7.: VAN, TIR y PRI del proyecto

VI.4 Factibilidad legal y responsabilidad civil

En virtud de ser un aspecto ajeno a la Ingeniería se ha consultado con especialistas en la materia, a fin de determinar la factibilidad legal del producto, así como el alcance de la responsabilidad civil. De acuerdo con las características reseñadas del producto, el experto concluyó que el equipo está alcanzado por la Resolución N° 92/1998 de la ex Secretaría de Industria, Comercio y Minería del ex Ministerio de Economía y Obras y Servicios Públicos. Asimismo, el producto queda encuadrado en la Ley N° 24.240 de Defensa del Consumidor y la Ley 11.723 “Régimen Legal de la Propiedad Intelectual”.

En primer lugar, toda vez que el *Gimbal* es un aparato electrónico de baja tensión (se alimenta con 14 VDC) se encuentra incluido en la previsión legal de la mencionada Resolución, que establece en su **Artículo 1º**: “Sólo se podrá comercializar en el país el equipamiento eléctrico de baja tensión que cumpla con los requisitos esenciales de seguridad que se detallan en el Anexo I en DOS (2) planillas que forman parte de la presente resolución, considerándose comercialización toda transferencia aún como parte de un bien mayor”.

Agregando en el **Artículo 2º**: “A los fines de la presente resolución se entiende por equipamiento eléctrico de baja tensión a los artefactos, aparatos o materiales eléctricos destinados a una instalación eléctrica o formando parte de ella, que tenga una tensión nominal de hasta MIL (1000) Volt en corriente alterna eficaz o hasta MIL QUINIENTOS (1500) Volt en corriente continua.”

Por otra parte, en el Anexo II de la resolución se establecen los procedimientos y plazos a cumplir para la certificación de los productos:

Primera Etapa:

El productor deberá presentar ante la Dirección Nacional de Comercio Interior, una declaración de conformidad del producto con los "requisitos esenciales de seguridad del equipamiento eléctrico de baja tensión", según Anexo I de la Resolución, teniendo la misma carácter de declaración jurada. La declaración debe basarse en ensayos o evaluaciones documentados por el productor o terceros y los mismos deben dar confianza en que el producto ofrecido cumple con los requisitos mencionados. Dicha presentación deberá realizarse dentro de los 8 meses posteriores al inicio de la comercialización.

Segunda Etapa:

Finalizada la etapa anterior y por el término de UN (1) año se podrá presentar ante la DIRECCION NACIONAL DE COMERCIO INTERIOR una Certificación de Conformidad de Tipo para los Requisitos Esenciales de Seguridad otorgada por un organismo de certificación acreditado ante O.A.A., de acuerdo a los procedimientos de certificación que para esta etapa éste establezca, en base a ensayos realizados por un laboratorio acreditado o reconocido, sobre especímenes representativos de la producción normal que serán seleccionados por el organismo de certificación. Periódicamente, el organismo de certificación tomará muestras en fábrica o en el mercado para determinar el cumplimiento con la Certificación de Conformidad de Tipo original.

No obstante ello, dado que se decidió comprar el cargador de la batería de litio, no se requerirán certificaciones adicionales puesto que tanto el cargador como la batería ya vienen homologadas y certificadas por su fabricante.

Por su parte, la Ley 24.240 de “Defensa del Consumidor” establece, en su Artículo 1º - Objeto: “La presente ley tiene por objeto la defensa de los consumidores o usuarios. Se consideran consumidores o usuarios, las personas físicas o jurídicas que contratan a título oneroso para su consumo final o beneficio propio o de su grupo familiar o social: a) La adquisición o locación de cosas muebles; ...” Sobre la base de esta previsión, y en virtud de que lo que se ofrece es la locación del producto, puede afirmarse que el servicio que se va a prestar está regulado en el Capítulo V – De la Prestación de los Servicios que, entre otras cosas prevé que “... Quienes presten servicios de cualquier naturaleza están obligados a respetar los términos, plazos, condiciones, modalidades, reservas y demás circunstancias conforme a las cuales hayan sido ofrecidos, publicitados o convenidos.”

Con respecto al software a utilizar para llevar a cabo este proyecto, se utilizará la versión especial del CodeWarrior 10.6 IDE para programar el microprocesador, que es gratuita. Por otro lado, la interfaz gráfica y el graficador dinámico con comunicación por puerto serie serán desarrollados en Java. Por lo tanto, no se requieren consideraciones respecto de la protección de la propiedad intelectual.

Estos son los puntos de mayor relevancia. No obstante, pueden encontrarse copias completas de las Leyes y la Resolución en el Anexo, al igual que un contrato tipo de locación del producto que se firmará con el locatario.

VII Ingeniería del detalle

VII.1 Hardware

VII.1.1 Diagrama en bloques

El *gimbal* puede descomponerse en 6 bloques de hardware, como se muestra en la Figura VII.1.1. De aquí podemos ver que la MCU principal mantendrá 2 canales de comunicación serie: uno de ellos se utilizará para intercambiar datos en tiempo real con un control remoto inalámbrico. El otro servirá para conectarse mediante USB con una computadora que corre el software propietario, y posee más funcionalidades que el control remoto, como la posibilidad de graficar las señales del sensor. Así, la conexión con la computadora está orientada a la puesta a punto inicial *off-line*, mientras que el control permite realizar el encuadre y cambiar algunos parámetros *on-line* para lograr el mejor funcionamiento.

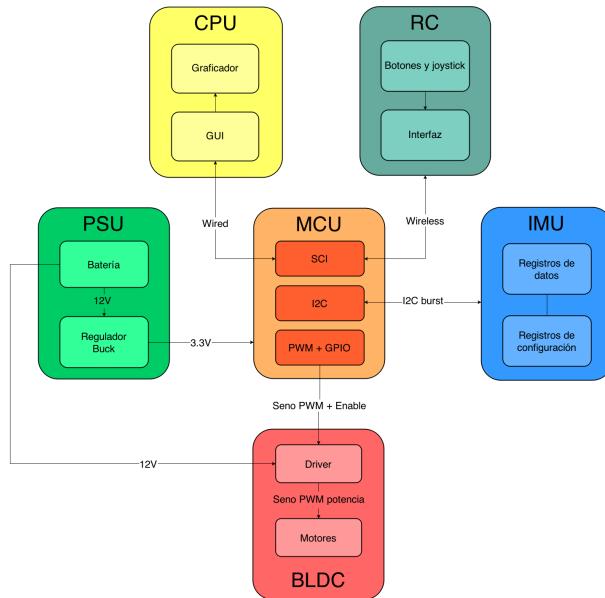


Figura VII.1.1.: Diagrama en bloques general del Statheros

En cuanto a la corrección del ángulo propiamente dicha, destacamos que el MCU actualiza la medición del sensor periódicamente y envía las señales PWM de control al *driver* de los motores, junto con la señal de *enable*. Utilizando la tensión de la batería, este último genera una señal amplificada que mueve los motores. Un regulador *buck* también toma los 12V y los convierte en 3,3V y 5V que alimentan al MCU y al IMU (los 5V se utilizan para los *level-shifters*).

VII.1.2 Descripción de cada bloque

VII.1.2.1 MCU

El microcontrolador debe ser capaz de controlar todos los bloques descritos anteriormente, por lo que debe contar con las siguientes funcionalidades:

- 9 canales PWM.
- 3 canales de GPIO.
- 2 canales de comunicación UART.
- 1 canal ADC.
- Interfaz de programación de aplicación (para cargar el programa en la placa final).

VII.1.2.2 Conversor USB-UART

Para realizar la transferencia de datos entre el microcontrolador y la computadora se utiliza comunicación serie por su simplicidad. Sin embargo, debido a que cada vez menos computadoras poseen un puerto serie, se decidió emplear un conversor para admitir una conexión por USB. El único requerimiento de este dispositivo es que sea capaz de mantener una comunicación a una tasa de 115,200 baud/s como mínimo, para garantizar latencias mínimas en el programa principal.

VII.1.2.3 IMU

El integrado IMU a utilizar debe tener, como mínimo, las siguientes prestaciones:

- Acelerómetro (tres ejes).
- Giroscopio (tres ejes).
- Amplificación de la señal.
- ADC integrado.
- Protocolo de comunicación compatible con el microprocesador.
- Precisión angular de al menos 1° y 1°/s.

De esta forma, los acelerómetros y giroscopios se encargan de realizar la medición y luego las señales analógicas son amplificadas, digitalizadas y finalmente transmitidas al microprocesador para ser procesadas.

VII.1.2.4 Driver

El sub-circuito de *driver* a implementar debe poder cumplir con los siguientes requerimientos:

- Entrada lógica de 5V.
- Salida de potencia de 12V con 0,7A pico por fase.
- Frecuencia de conmutación de 1kHz.
- *Auto dead-time compensation*.

El sub-circuito podría, desde luego, superar estos requerimientos, pero es necesario que los alcance. Los primeros dos son características de los motores a emplear, la frecuencia de conmutación es un resultado combinado de los requerimientos de corrección angular y la estrategia de control definida para los BLDC, y el *dead-time compensation* es un factor de seguridad cuyo empleo se recomienda en todo *half-bridge driver*.

VII.1.2.5 Control Remoto

En el siguiente diagrama en bloques se ven los distintos módulos que conforman el control remoto inalámbrico. Una breve descripción de cada uno de los bloques sigue al diagrama.

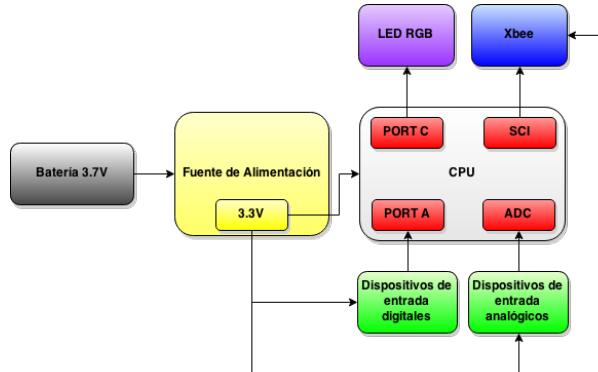


Figura VII.1.2.: Diagrama en bloques del control remoto inalámbrico

CPU

Es el bloque de control del *joystick*; se encarga de analizar las señales de control provenientes de los periféricos de entrada y dar la información para actualizar el LED de estados. Los módulos que se utilizan del microcontrolador son los siguientes:

- SCI: para comunicarse con el módulo RF.
- ADC: para comunicarse con los puertos de entrada analógicos.
- PORT A: para comunicarse con los puertos de entrada digital.
- PORT C: para controlar el LED de estados.

Se decidió usar este microcontrolador debido a su bajo precio, sus buenas prestaciones, y porque es el usado para controlar la plataforma.

Módulo RF

Este bloque se encarga de establecer la comunicación inalámbrica entre el *joystick* y el sistema estabilizado. Se comunica con el microcontrolador mediante el protocolo SCI usando una serie de comandos que permiten modificar las opciones de configuración del módulo RF y enviar datos.

LED de estado RGB

El microcontrolador se comunica con este dispositivo mediante el PORT C. Esto encender el LED en alguno de los colores disponibles indicando el estado donde se encuentre el control.

Fuente de alimentación

Consiste de una batería de 3,7V, y un *regulador lineal* que baje la tensión que provee la batería de 3,7V a 3,3V.

Dispositivos de entrada digitales

Son pulsadores con circuitos que mitigan los rebotes, que se comunican con el PORTA del microcontrolador.

Dispositivos de entrada analógicos

En este módulo se utiliza un potenciómetro doble colocado en configuración *joystick*, que permite el control de *pitch* y *yaw*, y un potenciómetro rotativo, para el control de *roll*. Se conectan al microcontrolador mediante los módulos ADC.

VII.1.3 Detalles de selección y cálculo de componentes

VII.1.3.1 MCU

Debido a la gran cantidad de integrados que cumplen con los requerimientos, se debió refinar la búsqueda exigiendo al microcontrolador un entorno de programación amigable y una base de soporte fuerte. Asimismo, se requiere que esté disponible una placa de desarrollo para el prototipado inicial, que también sirva para programar el microcontrolador de aplicación. Estos requerimientos fueron alcanzadas por el MK20DX128VLH6, un procesador de 50MHz basado en ARM Cortex-M4 que, además, está disponible en una placa de desarrollo FRDM-K20D50M, que cuenta con una interfaz openSDA para la programación con la computadora y SWD para cargar programas a otros dispositivos.

VII.1.3.2 Interfaz USB-UART

Las dos modelos más populares en este rubro son el FTDI RLxxx y el CP210x, ambos con *drivers* de computadora extensamente probados en los Windows, Linux y Mac. La decisión de utilizar el CP2104 se basó en el tamaño reducido (viene en paquete QFN-24) y por su bajo costo.

VII.1.3.3 IMU

Selección del Circuito Integrado

En la Figura VII.1.3, que se muestra a continuación, se presentan algunos de los circuitos integrados que se consideraron para la IMU.

Fabricante	Integrado	Sensibilidad de Acelerometro	Sensibilidad de Gyroscopio	Tipo de Comunicación	Precio (USD\$)
InvenSense	MPU6050	$\pm 2/\pm 4/\pm 8/\pm 16\text{g}$	$\pm 250/\pm 500/\pm 1000/\pm 2000$	I2C	10.7
InvenSense	MPU9250	$\pm 2/\pm 4/\pm 8/\pm 16\text{g}$	$\pm 250/\pm 500/\pm 1000/\pm 2001$	I2C; SPI	12.7
ST Microelectronics	LSM6DS0TR	$\pm 2/\pm 4/\pm 8\text{ g}$	$\pm 245/\pm 500/\pm 2000$	I2C; SPI	5.46

Figura VII.1.3.: Comparación entre IMU

De estos integrados, se descartó el MPU9250 debido a que este no sólo es el mas costoso sino que además cuenta con un magnetómetro de 3 ejes. Si bien esto representa una ventaja aparente frente a los otros dos, si se realiza un buen procesamiento de los datos del sensor, se pueden alcanzar prestaciones similares a la de este integrado sin necesidad de utilizar la información del magnetómetro.

De las opciones restantes, se escogió el MPU6050 por encima del LSM6DS0TR. Si bien este ultimo es menos costoso, el primero está mucho mas difundido, presenta una documentación de software extensa, y en consecuencia es mucho mas confiable. Adicionalmente este sensor se encuentra disponible en placas de prueba en el país, por lo que su adquisición es mucho mas fácil.

Análisis por Bloques

En Figura VII.1.4 vemos el diagrama en bloques del integrado elegido, provisto por Invensense, que explica los módulos que conforman el MPU6050. Una breve descripción de cada uno de los bloques sigue al diagrama.

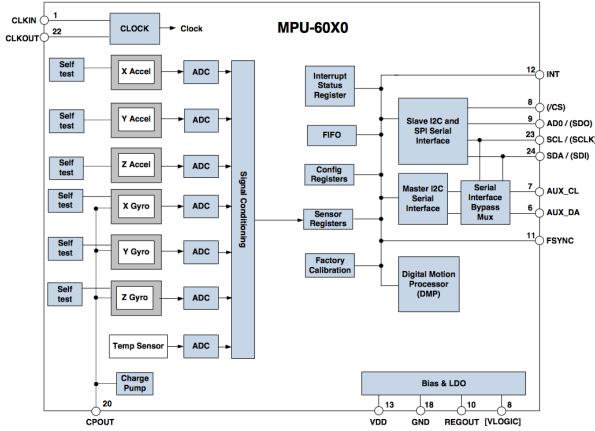


Figura VII.1.4.: Diagrama en bloques del elemento sensor

Self test Este módulo se emplea para realizar una prueba de los sensores; si se habilita el *self test*, entonces el módulo activará los sensores en forma controlada, produciendo así una señal de salida conocida que se evalúa para determinar la presencia de fallas en el sistema.

Gyroscope and signal conditioning Existen tres giroscopios individuales de tecnología MEMS, cada uno dedicado a realizar mediciones en un eje rotacional diferente. Cuando el giroscopio se rota, el efecto Coriolis causa una vibración que es detectada por un sensor capacitivo, generando una señal que es amplificada, demodulada y filtrada para obtener un nivel de tensión directamente proporcional a la velocidad angular de la rotación. La salida del sensor se mantiene (mediante un *latch*) en los registros de sensores (*sensor registers*). El giroscopio requiere de una calibración externa, que incluye la programación del valor de fondo de escala de la medición, para funcionar correctamente.

Accelerometer and signal conditioning Existen tres acelerómetros individuales de tecnología MEMS, cada uno dedicado a realizar mediciones en un eje rotacional diferente. Cuando el acelerómetro se rota (o sencillamente cuando la gravedad actúa sobre él), una masa de prueba interna se desplaza, y sensores capacitivos perciben este desplazamiento en forma diferencial; este proceso resulta en una medición de las aceleraciones experimentadas por el integrado. La salida del sensor se mantiene (mediante un *latch*) en los registros de sensores (*sensor registers*). El acelerómetro requiere de una calibración externa, que incluye la programación del valor de fondo de escala de la medición, para funcionar correctamente.

Temperature sensor Este sensor realiza mediciones de temperatura para poder compensar los efectos que esta tiene en la medición de los giroscopios y acelerómetros. Esta información puede recuperarse a través de los registros de sensores (*sensor registers*) o el registro FIFO.

ADC 16-bits El MPU6050 cuenta con seis ADCs sigma-delta de 16-bits de resolución, uno para cada uno de los giroscopios y acelerómetros empleados en la medición. Se puede programar, entre otras cosas, la frecuencia de muestreo de los ADCs, el tipo de filtro pasa-bajo a aplicar, y la frecuencia de corte del mismo.

I2C communication interface Este módulo estaría compuesto por los bloques *Slave I2C* y *Master I2C* (todo lo referente a comunicación serial está sólo presente en el MPU6000, no en el MPU6050). En forma predeterminada, el IMU actúa como *slave* en la comunicación I2C, pero también puede funcionar como *master* si así se lo requiere. El nivel de tensión para la lógica comunicacional se determina a través del pin VLOGIC.

Digital Motion Processor (DMP) Este bloque ofrece algoritmos de procesamiento digital de movimiento; la función principal de este bloque es reemplazar los filtros complementarios o de Kalman implementados por el usuario para el procesamiento de las mediciones. La correcta utilización de este bloque implicaría acondicionamiento digital programado específicamente para las señales características del integrado, obteniendo así una compatibilidad única entre hardware y software, además de alivianar el trabajo del procesador principal. El MPU6050 ofrece un pin de interrupciones relacionado con este módulo.

Interrupt status register Este integrado cuenta con un registro configurable para generar interrupciones asincrónicas en el procesador principal. Este servicio puede ser utilizado en la comunicación I2C, serial, o para funcionar junto con el DMP.

Configuration registers Los registros empleados para configurar parámetros de funcionamiento del integrado, como ser características de la comunicación I2C, características de la medición, configuración de los ADCs, registro FIFO, etc.

Bias and LDO Este bloque genera las tensiones y corrientes internas que requiere el integrado para funcionar.

Charge Pump Este bloque genera las altas tensiones que requieren los osciladores de los giroscopios para realizar la medición.

FIFO Este bloque se emplea para facilitar la comunicación a través de interfaz serial. Se puede configurar qué datos se volcarán en el mismo, si los del acelerómetro, los del giroscopio, la medición de temperatura, etc.

Level shifter El objetivo principal de este bloque es recibir de forma fiel las lecturas del giroscopio y el acelerómetro provistas por el IMU. En la Figura VII.1.5 presentada a continuación se muestra el circuito normalmente utilizado para implementar un *level-shifter*; este, básicamente, convierte una señal de 5V en una de 3,3V y viceversa.

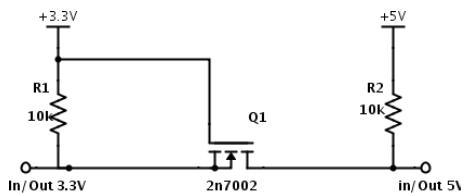


Figura VII.1.5.: Circuito de un *level-shifter* genérico

La tarea de recepción de datos se hace a través del módulo de comunicación I2C del microprocesador. Como el MCU trabaja con lógica de 3,3V de bajo consumo, la línea de comunicación es vulnerable frente a fuentes de ruido externas, como las que se presentan en el *gimbal*. En particular, se tiene ruido eléctrico de alta frecuencia y alta energía producido por la commutación de 12V/1A a 32kHz del PWM que controla

los motores, y fuertes campo magnéticos generados por los imanes permanentes de los BLDC. Por estas razones, se decidió implementar *level-shifters* en el lado de la MCU (*step-up*) y de la IMU (*step-down*) para aumentar la tensión de referencia del mensaje de I2C a 5V y así incrementar la inmunidad frente al ruido del sistema.

VII.1.3.4 Driver

Alternativas de Diseño

La primera decisión en cuanto a la implementación del *driver* viene dada por la forma de fabricación. Una posible implementación de este subsistema consiste en el desarrollo de un *driver* discreto, que consiste en utilizar componentes como compuertas lógicas y transistores de potencia para construirlo. Si bien esta alternativa permite tener un mayor control sobre las prestaciones del dispositivo, es una opción bastante compleja y que demanda mucho tiempo de desarrollo. Si se considera que el proyecto, en su totalidad, es de por sí complejo, se decidió que el desarrollo de un *driver* discreto no formará parte de los problemas a resolver.

La segunda posibilidad consiste en utilizar un circuito integrado especialmente diseñado para motores BLDC. Estos circuitos requieren el uso de pocos componentes adicionales, tienen especificaciones confiables y ocupan menos espacio físico en el PCB. La desventaja frente a una implementación discreta radica en que las prestaciones del *driver* son impuestas por el fabricante y que suelen ser más costosos.

Tomando en cuenta todo lo anterior, se optó por implementar un *driver* con circuito integrado.

Selección del Circuito Integrado

En la tabla VII.1.6, que se muestra a continuación, se presentan alguno de los circuitos integrados que se consideraron para la implementación del *Driver*.

Fabricante	Integrado	Corriente de Salida	Tensión de Entrada	Tipo de Comunicación	Precio (USD\$)
Texas Instruments	DRV10983	2A	8V - 28V	PWM; I2C	3
Microchip	TC4451	2.6A	4.5V - 18V	PWM	1.84
ST Microelectronics	L6234	4A	7V - 52V	PWM	8.69

Figura VII.1.6.: Comparación entre Circuitos de Driver

De los circuitos integrados presentados, se descartó el TC4451 debido a que sólo puede manejar una sola fase y, debido a que cada motor tiene tres, generaría complicaciones en el diseño final de la placa de control, como también mayor consumo y espacio.

Finalmente se escogió el L6234 por encima del DRV10983 porque, si bien este es más costoso, se ha usado en otros proyectos similares, y su abarcado uso y difusión lo hacen más confiable. Adicionalmente, al tener mayor corriente máxima de salida, permite que el proyecto sea escalable en el sentido de que puede ser usado con motores de mayor potencia, y en consecuencia, manejar cámaras más pesadas si así se decidiese.

Análisis por Bloques

En la Figura VII.1.7 se muestra el diagrama de bloques interno del integrado. Una breve descripción de cada uno de los bloques sigue al diagrama.

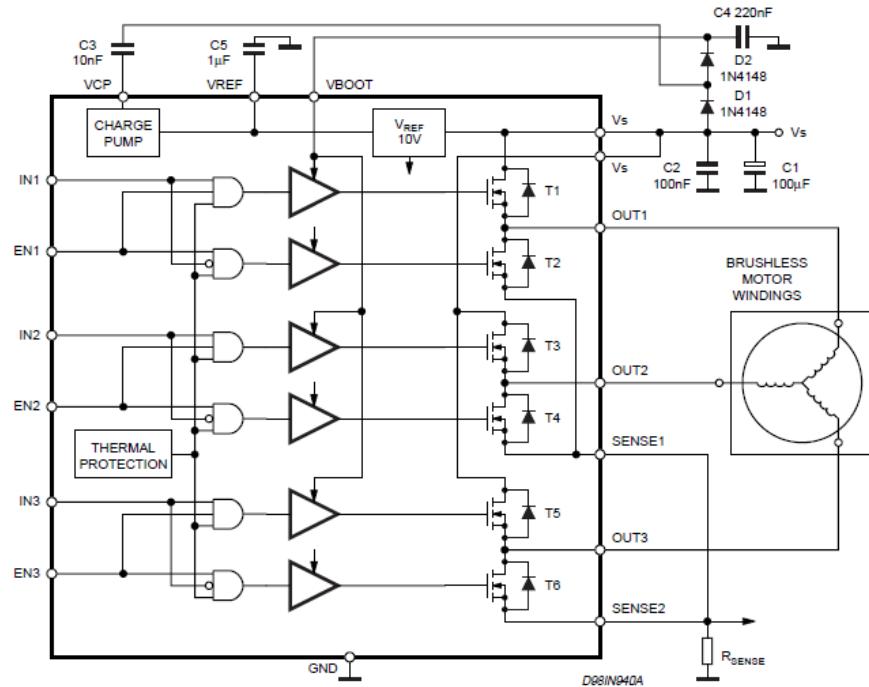


Figura VII.1.7.: Diagrama en bloques del driver

Bloque Lógico Está conformado por las compuertas AND. Tienen como función servir de interfaz entre el control externo y el control del *driver*.

Bloque Buffer Está conformado por los *buffers*. Tienen como función aislar el *Bloque Lógico* del *Bloque de Potencia*.

Bloque de potencia Está conformado por los MOSFET y los diodos. Su función es recibir la señal lógica por parte del control y reproducirla a la salida, a la tensión y corriente necesarias para mover el motor.

Thermal protection Este bloque consiste en una protección interna frente a las altas temperaturas. Cuando el integrado supera cierta temperatura, a partir de la cual se considera que entra en peligro, la protección deshabilita el acceso al control, eliminando así el peligro.

Charge Pump y VRef Este bloque se encarga de tomar como referencia la tensión de entrada V_s , y proveer al circuito todas las tensiones necesarias (referidas a esta).

Selección de Componentes

En la Figura VII.1.8 se muestra el esquemático utilizado en la placa del driver. Los componentes seleccionados (diódos y capacitores) fueron elegidos según la recomendación del fabricante (ver FiguraVII.1.7)

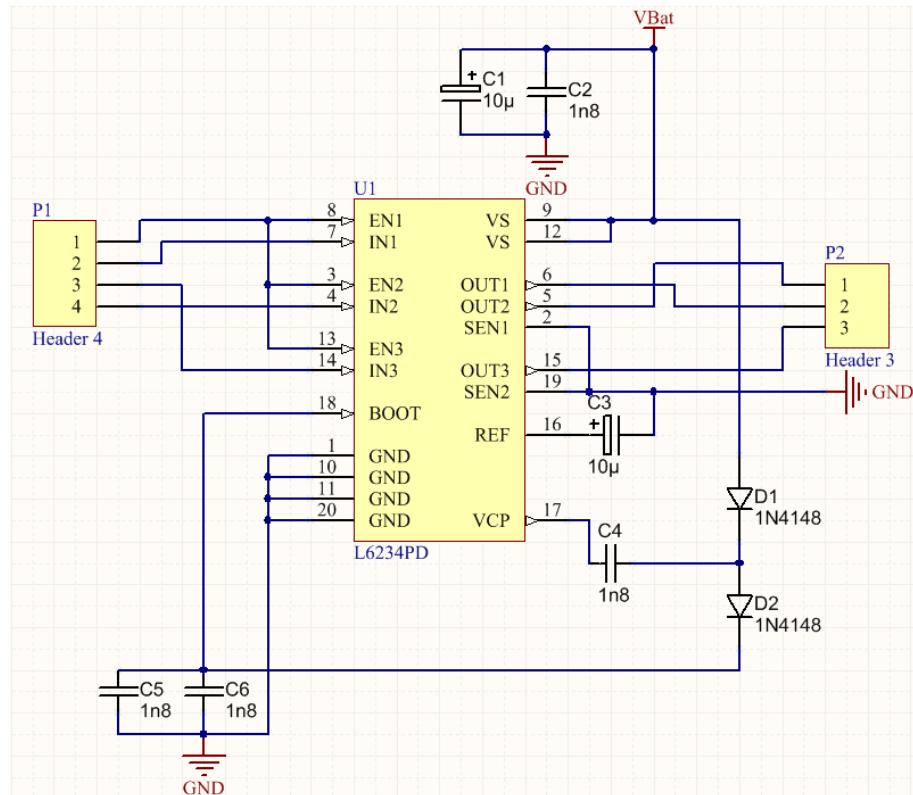


Figura VII.1.8.: Esquemático del Circuito Driver

VII.1.3.5 Control remoto

En todos los bloques que se verán, el factor determinante en la elección de los componentes es la disponibilidad local de los mismos.

MCU

El microcontrolador usado es el Freescale K20DX128LVH5, el mismo utilizado en el controlador principal del gimbal. Se decidió usar este microcontrolador debido a su bajo costo, buenas prestaciones, y porque es el usado para controlar la plataforma.

Módulo RF

En la Figura VII.1.9 presentada a continuación, se muestran alguno de los módulos que se consideraron para el bloque de transmisión por RF.

Fabricante	Nombre del Módulo	Protocolo de Comunicación	Frecuencia de Transmisión (GHz)	Tipo de Comunicación	Tasa de Transmisión	Alcance (m)	Precio (USD\$)
Wenshing	TWS-BS y RWS - 371	AM - ASK	0.45592	Paralela (8 bits)	4.8 kbps	50 - 100	8.9
iTea Studio	HC - 05	Bluetooth V2.0	2.4	UART; SPI	9600 - 460800 baud/s	50	5
Digi	Xbee	Zigbee	2.4	UART; SPI	9600 - 460800 baud/s	100-200	20

Figura VII.1.9.: Comparación entre módulos RF

El primero en ser descartado fue el modulo común por AM. La razón mas influyente en esto fue su baja velocidad de transmisión, como también el hecho de que la implementación de la comunicación con el microcontrolador debe ser desarrollada enteramente y no es ningún protocolo de comunicación conocido. Posteriormente se descarto la opción del modulo por Bluetooth. Si bien es cierto que es la mas económica, el poco alcance, y el hecho de que se requiere desarrollar ciertos conocimiento del protocolo Bluetooth para

su correcta configuración (lo cual no forma parte del objetivo del proyecto) lo hacen una opción menos eficiente frente a la tercera.

Finalmente, y en consecuencia, se decidió implementar la comunicación con Módulos Xbee. Su principal ventaja es que si bien esta basado en el protocolo Zigbee, el Xbee no requiere desarrollar conocimientos del mismo ya que se pueden configurar sus parámetros a vía SCI/UART con comando AT. La combinación de prestaciones, facilidad de implementación y poco tiempo de desarrollo terminaron siendo los factores mas influyentes en la selección de este dispositivo como alternativa, a pesar de ser la mas costosa.

En la Figura VII.1.10 presentada a continuación se muestra un modulo Xbee.



Figura VII.1.10.: Módulo Xbee

LED de estados RGB

Se usa un LED RGB through hole con resistencias de 150Ω para limitar la corriente. La terminal común del LED se conecta directamente a 3,3V y las otras terminales se conectan al PORT C mediante las resistencias.

Fuente de alimentación

Se utiliza una batería de litio-ion recargable de 3,7V para alimentar todos los componentes del control remoto. Se utiliza un reguladores lineal para reducir los 3,7V a 3,3V. Se eligio el mismo regulador lineal de baja caida que para el sistema de estabilización.

Dispositivos de entrada digitales

En la Figura VII.1.11 vemos el diagrama del circuito anti-rebote usado en los botones del control remoto. Los valores de los componentes se calcularon de forma tal que el tiempo de carga del capacitor baste para evitar el rebote del pulsador, pero lo suficientemente chico para que sea despreciable. Además, se buscó que los valores de los componentes necesarios estén disponibles en encapsulados SMD 1206. Con estos criterios, se eligieron los valores que se presentan junto al diagrama circuitual de la FiguraVII.1.11.

El diodo elegido es un BAT54, que posee una caída de 240 mV, por lo que deja la tensión de salida dentro de los valores mínimos admisibles por el microprocesador.

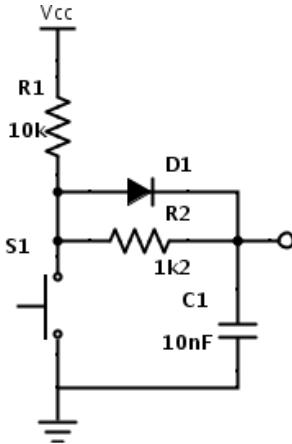


Figura VII.1.11.: Diagrama del circuito anti-rebote

Dispositivos de entrada analógicos

Para los dispositivos de entrada analógicos que controlan los movimientos en tres dimensiones se utilizaron tres potenciómetros, cada uno conectado a una entradas ADC del microprocesador. Para controlar el *yaw* y el *pitch* se eligió un potenciómetro doble de $10\text{k}\Omega$, en configuración de *joystick*, y para controlar el *roll* se optó por uno radial de $10\text{k}\Omega$.

El ADC se está configurado para tener una resolución de 8 bits, ya que no se requiere mucha precisión (e.g. 16 bits) para medir los valores de posición. Los potenciómetros se alimentan con la tensión del microprocesador (3,3V).

Basado en todo lo anterior, podemos calcular la resolución del ADC según:

$$\text{Res}_{\text{ADC}} = \frac{3,3\text{V} - 0\text{V}}{2^8} = 12,89\text{mV}$$

VII.1.4 Plan de pruebas de cada módulo

VII.1.4.1 IMU

Para verificar el correcto funcionamiento del IMU se realizarán una serie de ensayos:

Ensayo de level shifter

Este control es necesario para garantizar que no hayan daños en el integrado ni en el microprocesador; sólo se verifica tener una señal de la tensión correspondiente a cada lado del *level-shifter*. Esto es tan simple como fundamental, dado que una variación de tensión, incluso pequeña, podría causar daños irreparables en los componentes previamente mencionados.

Ensayo de módulo I2C

Este control se avoca a verificar el correcto funcionamiento del módulo I2C; enviando el comando *whoAmI* al IMU desde el microprocesador se procede a evaluar la respuesta del sensor. Este comando tiene respuesta conocida, por lo cual si la actual difiriese de la obtenida podrían estar ocurriendo tres situaciones: la primera, que el módulo I2C del microprocesador haya dejado de funcionar; la segunda, que el módulo I2C del IMU ha dejado de funcionar; y la tercera, que los cables que conectan los dispositivos estén generando interferencia.

Basta verificar las señales enviadas desde el microprocesador correspondientes a un mensaje conocido para descartar la primera opción; esto se haría empleando un osciloscopio.

Basta cambiar los cables existentes por unos cortos y que no se crucen entre sí para descartar la tercera opción.

La segunda opción puede sólo descartarse luego de haber descartado las otras dos.

Ensayo de *self-test*

Este control verifica el hardware del IMU mediante el protocolo interno de *self-test*.

Ensayo de medición

Este control evalúa la correcta configuración del filtro complementario y/o el DMP. Se coloca el IMU en una inclinación conocida y se verifican las lecturas; si estas difiriesen de lo esperado, si se realizó el *self-test* previamente en forma exitosa, se debe re-configurar el sistema de procesamiento digital y repetir la prueba hasta que no hayan diferencias mayores.

VII.1.4.2 Driver

Para verificar el correcto funcionamiento del driver, se realizarán dos ensayos: en vacío y con carga.

Ensayo en vacío

El objetivo de la prueba en vacío es verificar que las señales de control se estén transmitiendo de la manera correcta al driver. Para realizar este ensayo se debe conectar el circuito de la Figura VII.1.8 de la siguiente manera: Las entradas IN1, IN2, e IN3 se deben conectar a las salidas lógicas de PWM del MCU. A continuación, el pin que conecta EN1, EN2 y EN3, se debe conectar a una de las salidas lógicas del MCU. Finalmente se debe suministrar alimentación (identificada en el circuito como VBat) mediante una fuente externa regulada a 12V.

Una vez realizadas las conexiones, se coloca una punta de osciloscopio en una de las entradas y otra en la salida correspondiente (EN1 con OUT1 y así sucesivamente). Finalmente se pasa a transmitir una secuencia de PWM de prueba (frecuencia y DC arbitrarias). Si la salida muestra una onda idéntica a la de la entrada (salvo un retraso en el tiempo) significa que dicha fase del driver está funcionando de manera correcta. Una vez verificado que las tres fases del driver funcionan adecuadamente, se puede proceder al ensayo con carga.

Ensayo con carga

El objetivo de la prueba con carga es verificar que el driver es capaz de proveerle al motor la corriente y tensión necesarias, como también verificar que el MCU es capaz de controlar el movimiento del motor, a través del driver.

Para realizar dicho ensayo, se debe conectar el MCU y el driver de la misma forma que en el ensayo en vacío. Adicionalmente se deben conectar las tres fases del motor a las salidas OUT1, OUT2 y OUT3.

Una vez conectado el circuito se coloca una punta de osciloscopio en una de las entradas y otra en la salida correspondiente (EN1 con OUT1 y así sucesivamente). Finalmente se pasa a transmitir una secuencia de PWM de rotación positiva, y rotación negativa. Dicha secuencia, provista por el fabricante, se muestra en la Figura VII.1.12.

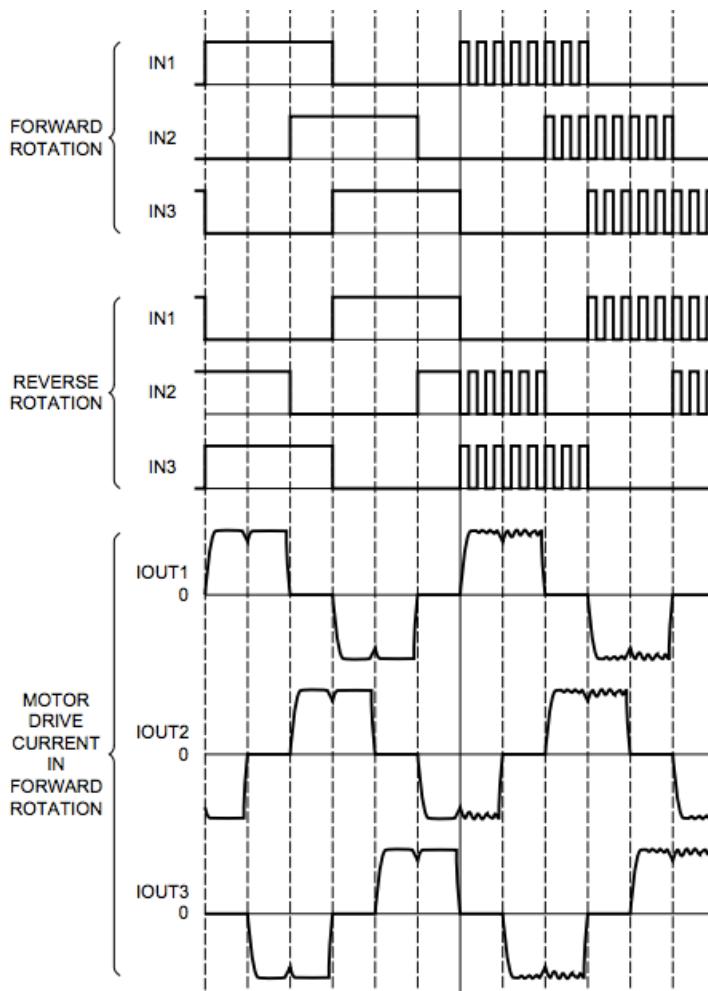


Figura VII.1.12.: Secuencia de PWM de rotación

Si la salida muestra una onda similar a la de la Figura VII.1.12 (salvo un retraso en el tiempo) significa que dicha fase del driver está funcionando de manera correcta. Una vez verificada que las tres fases del driver funcionan adecuadamente, podemos asegurar que el driver está funcionando correctamente.

VII.1.4.3 Control Remoto

Ensayo de batería

Se realiza de la misma forma que en el caso de la batería del sistema estabilizado, salvo que se descarga a una corriente constante de 0.5A, representativo del consumo máximo del sistema. Se calculó este consumo sumando los máximos de corrientes según las hojas de datos de cada componente. Con esto se contrastará la duración máxima de la batería con la que provee el fabricante.

Ensayo de fuentes de alimentación

Este ensayo consiste en medir las tensiones de salida de los reguladores con un osciloscopio para determinar si la salida posee *ripple* de tensión despreciable. Para determinarlo, se investigó el componente más sensible a estas fluctuaciones, que resultó ser el *display*, con una tolerancia de hasta 5%. Esta restricción debe cumplirse para pasar la prueba, en caso contrario, se debe cambiar el integrado y volver a ensayar.

Ensayo del LED de estado RGB

Esta prueba consiste en mandar una combinación de bits a los periféricos del PORT C de modo que se prendan por separado los colores del LED. En caso contrario se deberá revisar el código del PORT C o el estado de las conexiones de las terminales del LED.

Ensayo de módulo Xbee

Este ensayo consiste en enviar cadenas de caracteres entre ambos dispositivos (receptor y transmisor) usando la interfaz SCI y el modo consola del *debugger* del microprocesador. De esta forma, el sistema funciona como una conexión serie inalámbrica y cada dispositivo debería recibir todo lo enviado por el otro. En caso contrario deberá inspeccionarse si el dispositivo se encuentra bien conectado y configurado en los canales correctos mediante el uso de comandos AT.

Ensayo de circuitos de entrada analógicos

Este ensayo consiste en cambiar la intensidad de un LED mediante el uso del módulo PWM del microprocesador, el cual recibe los datos directamente de uno de los canales de entrada del conversor AD. Si el brillo del LED se corresponde con la posición del potenciómetro, el sistema funciona correctamente.

Ensayo de circuitos de entrada digitales

Este ensayo es similar al anterior, solo que en vez de controlar el brillo del LED, se verifica que se prenda cuando se apriete uno de los botones.

VII.2 Software

VII.2.1 Diagramas de estado y flujoogramas

VII.2.1.1 Control de inclinación

El programa desarrollado para el control de inclinación se compone de módulos independientes. Se los denomina como independientes debido a que cada uno cumple una tarea puntual, pero cada módulo se alimenta con los resultados del anterior, encargados de desarrollar diferentes tareas. En la Figura VII.2.1 presentada a continuación se ilustra un sencillo diagrama de bloques representativo del algoritmo; cada bloque se detallará a largo de esta sección.

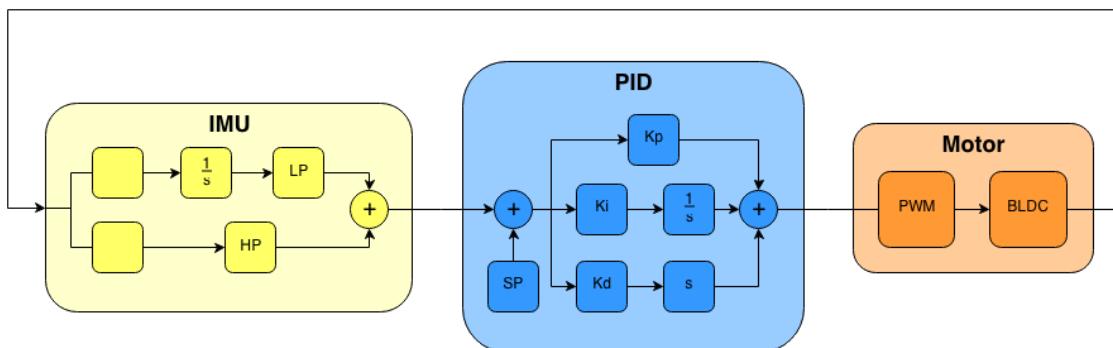


Figura VII.2.1.: Diagrama de bloques del control de posición

Procesamiento digital

El objetivo principal de este bloque es refinar y adaptar los datos obtenidos en el bloque anterior para poder usarlos en el algoritmo de control.

Una vez obtenidos los datos del acelerómetro y el giroscopio es aconsejable un procesamiento digital para aumentar la fidelidad de las lecturas. Para lograrlo se utiliza un filtro complementario cuyo funcionamiento se resume en el diagrama de la Figura VII.2.2.

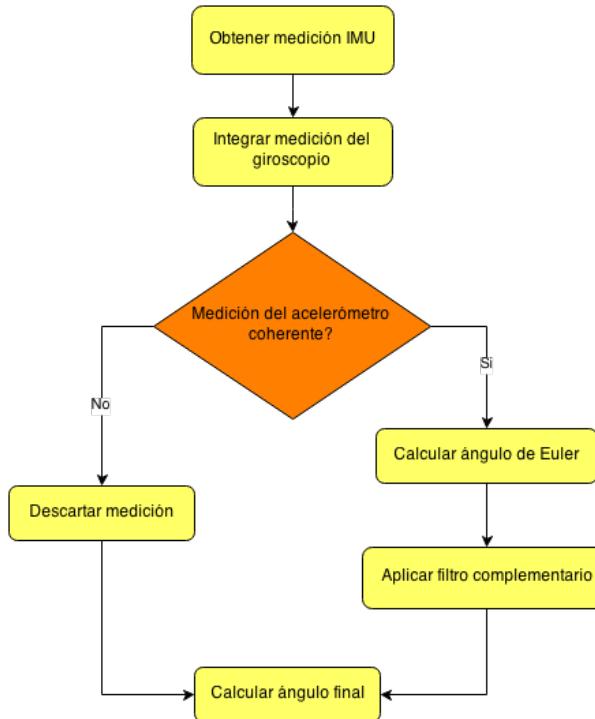


Figura VII.2.2.: Flujograma del filtrado de la medición angular

Inicialmente, se deben acondicionar los datos que proporciona la IMU utilizando los valores de las sensibilidades y escala que se establecieron en la inicialización del sensor. Los valores de la velocidad angular (en $^{\circ}/s$) y aceleración (en g) se obtienen a partir de las ecuaciones VII.2.1 y VII.2.2, respectivamente.

$$\omega_i = \frac{\Omega_i - offset_i}{\frac{2^{16}-1}{FSR}} \quad (\text{VII.2.1})$$

$$a_i = \frac{A_i - offset_i}{\frac{2^{16}-1}{FSR}} \quad (\text{VII.2.2})$$

Donde Ω_i y A_i son los enteros de 16 bits sin procesar del eje i que se obtienen de la IMU y FSR es la escala a rango completo. Este último parámetro puede tomar valores predefinidos, que se detallan en la hoja de datos del fabricante, adjunta en Anexos.

Una vez procesados los datos, se procede a realizar la integración numérica simple de la velocidad angular, según la ecuación VII.2.3, donde $\omega_{[k+1]}$ es la medición de velocidad angular actual, $\theta_{[k]}$ es el último valor de la posición angular, dt es el período de muestreo y $\theta_{[k+1]}$ es el ángulo actual.

$$\theta_{[k+1]} = \theta_{[k]} + \omega_{[k+1]} \cdot dt \quad (\text{VII.2.3})$$

Debido al *offset* inevitable del giroscopio y los errores de medición, siempre habrá error en $\omega_{[k+1]}$ y se acumulará sin control, si no se aplica un método correctivo. Es aquí donde el acelerómetro juega un papel

importante, ya que provee mediciones de ángulo absolutas. Sin embargo, debido a su principio de funcionamiento, es inherentemente sensible al ruido, por lo suele producir mediciones erróneas espontáneamente. Para eliminar estas variaciones se utiliza la identidad física de la ecuación VII.2.4, que se demuestra planteando cosenos directores a vectores genéricos en 3D.

$$F = \sqrt{a_x^2 + a_y^2 + a_z^2} \quad (\text{VII.2.4})$$

$$a_i = g_i \Rightarrow F = 1 \quad (\text{VII.2.5})$$

La igualdad VII.2.5 se cumple siempre que el cuerpo esté sometido únicamente a la aceleración de la gravedad; en el caso que se produjeran aceleraciones importantes en cualquiera de los ejes, esta regla de decisión perdería validez. Esto suele ocurrir en *drones*, pero no en un *gimbal* como el que se trata en este trabajo. Una vez que se tiene la medición apropiada del acelerómetro, se utilizan las ecuaciones VII.2.6 y VII.2.7 para pasar a ángulos de Euler. Es importante destacar que con este método sólo se pueden obtener mediciones de *pitch* (θ) y *roll* (ϕ), puesto que la gravedad en el eje z es invariante frente a cambios en la guiñada (ψ).

$$\theta_{acc} = -\text{atan2}(a_x, \sqrt{a_y^2 + a_z^2}) \quad (\text{VII.2.6})$$

$$\phi_{acc} = -\text{atan2}(a_y, \sqrt{a_x^2 + a_z^2}) \quad (\text{VII.2.7})$$

Finalmente, se aplica el filtro complementario sobre las dos mediciones angulares utilizando las ecuaciones VII.2.8 y VII.2.9, donde α es la variable de ajuste que determina cuánto peso relativo se le asigna a cada medición. Una manera de expresar este valor es con la ecuación VII.2.10, donde τ representa el tiempo luego del cual el acelerómetro corrige la medición del giroscopio, luego de una perturbación. Está claro que, cuanto mayor el valor de α , menor será el ruido de la señal (porque el acelerómetro está atenuado) pero mayor será el tiempo de establecimiento (el acelerómetro va a corregir el ángulo con pequeños aportes).

$$\theta_{final} = \alpha \cdot \theta_{[k+1]} + (1 - \alpha) \cdot \theta_{acc} \quad (\text{VII.2.8})$$

$$\phi_{final} = \alpha \cdot \phi_{[k+1]} + (1 - \alpha) \cdot \phi_{acc} \quad (\text{VII.2.9})$$

$$\alpha = \frac{\tau}{\tau + dt} \quad (\text{VII.2.10})$$

Cálculo de la corrección

El objetivo principal de este bloque es utilizar las medidas obtenidas en el bloque anterior para generar parámetros de movimiento para los motores. El flujo de la Figura VII.2.3 describe el funcionamiento del cálculo del ángulo de corrección.

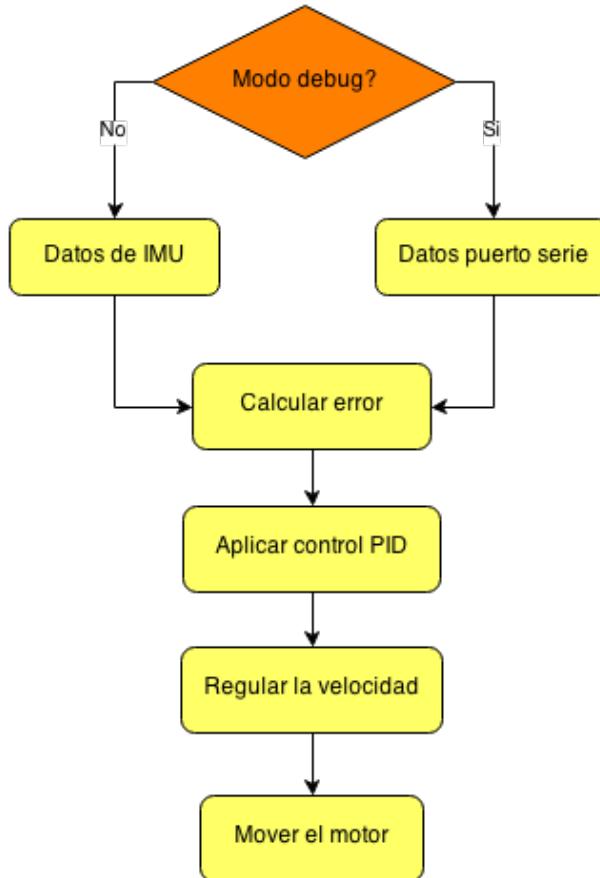


Figura VII.2.3.: Flujograma de la corrección de la posición

Lo primero que evalúa la función es el estado de operación. El modo *debug* sirve para poder probar el funcionamiento de los motores (ver Plan de pruebas del hardware). Con cualquiera de las dos entradas, se calcula la señal error entre el punto de trabajo deseado y la medición.

$$\text{error} = \text{setPoint} - \text{medición} \quad (\text{VII.2.11})$$

Esta señal sirve de entrada para el algoritmo de control PID, caracterizado por el diagrama de flujo de la Figura VII.2.4.

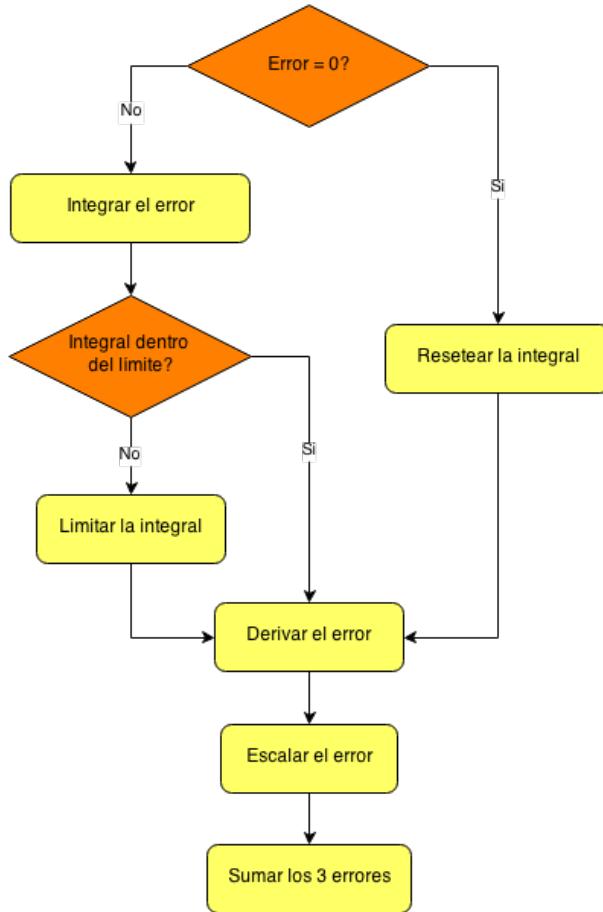


Figura VII.2.4.: Flujograma del algoritmo PID

A la salida se obtiene un nuevo error que alimenta el bloque regulador de velocidad, que se implementa con un control proporcional de la forma $k \cdot \theta$, donde k es una constante y θ es el ángulo comandado. Este método admite movimientos fluidos en un amplio rango de ángulos de rotación y es muy fácil de implementar. Sin embargo, debido a que el control de posición se realiza a lazo abierto, hay límites para la máxima velocidad de movimiento que asegura que no se pierda sincronismo entre el movimiento y la señal.

Algoritmo de movimiento de los motores

Tomando como entrada la velocidad y posición deseadas del bloque anterior, este bloque se encarga de mover los motores para cumplir las órdenes impuestas. Para lograrlo, se hace variar el *duty cycle* (DC) de tres canales de PWM para energizar las fases de los motores BLDC. Estas señales, desfasadas en 120° , alimentan los tres bobinados del motor, convirtiéndolos en electroimanes que atraen progresivamente el rotor de imán permanente. La forma de las señales de salida presenta dos alternativas posibles de conmutación de las fases: cuadrada o senoidal.

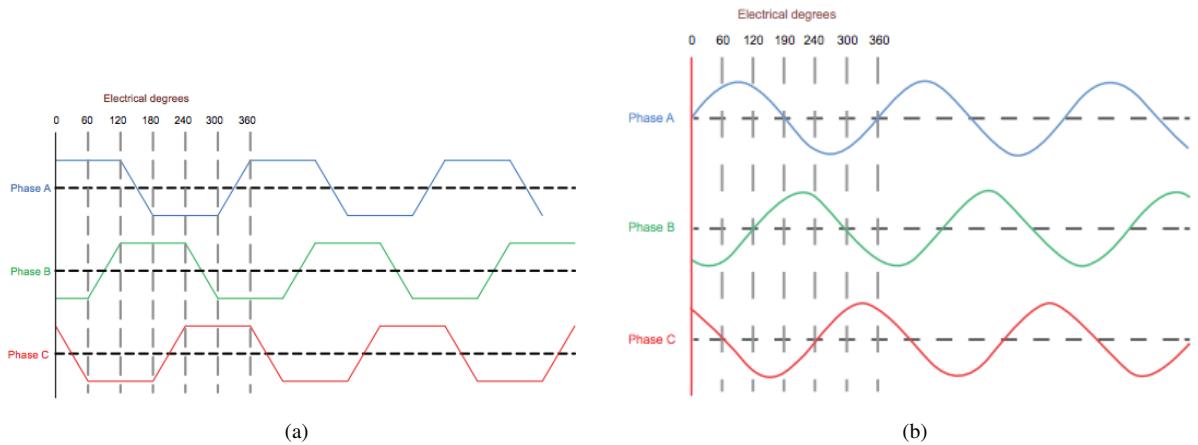


Figura VII.2.5.: Modos de conmutación de fases: a) Cuadrada, b) Senoidal

Un parámetro importante de comparación entre los dos esquemas es la potencia de las señales, ya que determina el torque máximo que pueden efectuar los motores. En el caso de la conmutación cuadrada, se energizan sólo dos fases por vez, lo que posibilita obtener tensiones positivas y negativas dependiendo de la dirección de la corriente. Por lo tanto, si hace circular corriente según la secuencia que muestra la Figura VII.2.6, se obtiene una señal como la de la Figura VII.2.5a, centrada en 0 y con amplitud a (ecuación VII.2.13). En el bobinado inactivo se induce una fuerza electromotriz por el movimiento del rotor, que se refleja en la recta que une los estados de tensión positiva ($+V$) y negativa ($-V$). Este fenómeno puede explotado para determinar la posición del rotor midiendo la tensión en la fase libre y así efectuar un control de posición. Sin embargo, en aplicaciones de movimiento impredecible, como las que se trata en este trabajo, este método no es de utilidad porque la forma de onda es mucho más irregular que como se presenta.

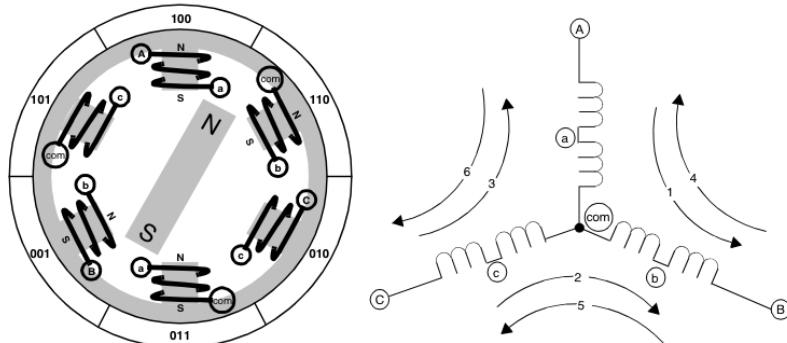


Figura VII.2.6.: Secuencia de conmutación para el movimiento del BLDC

$$RMS[\frac{a}{2} \cdot \sin(2\pi ft) + \frac{a}{2}] = \frac{3}{2\sqrt{2}} \cdot a \approx 1,06 \cdot a \quad (\text{VII.2.12})$$

$$RMS[a \cdot \text{sqr}(f)] = a \quad (\text{VII.2.13})$$

Por el lado de la conmutación senoidal, tenemos que todas las bobinas están siendo excitadas en todo momento, por lo que no puede haber inversión total de la corriente. Esto significa que la forma de onda de la señal de cada fase es un seno de amplitud $a/2$ con un *offset* de $a/2$, como se muestra en la ecuación VII.2.12. De aquí vemos que la potencia RMS de la señal senoidal es un 6% mayor que la de la cuadrada.

La forma del torque es otra característica importante a la hora de decidir el esquema a utilizar. En el caso de la onda cuadrada, los bobinados se energizan súbitamente debido a las discontinuidades, atraen al rotor y luego se apagan para permitir que el resto repita el proceso. Esto da como resultado un torque pulsante (Figura VII.2.7a) que genera movimientos alternantes y provoca vibraciones en los bobinados. En contraposición, la señal senoidal es una función continua con derivadas de orden n continuas, por lo que los cambios de velocidad y aceleración son suaves. Es por esta propiedad que el torque es casi constante, como se muestra en la Figura VII.2.7b.

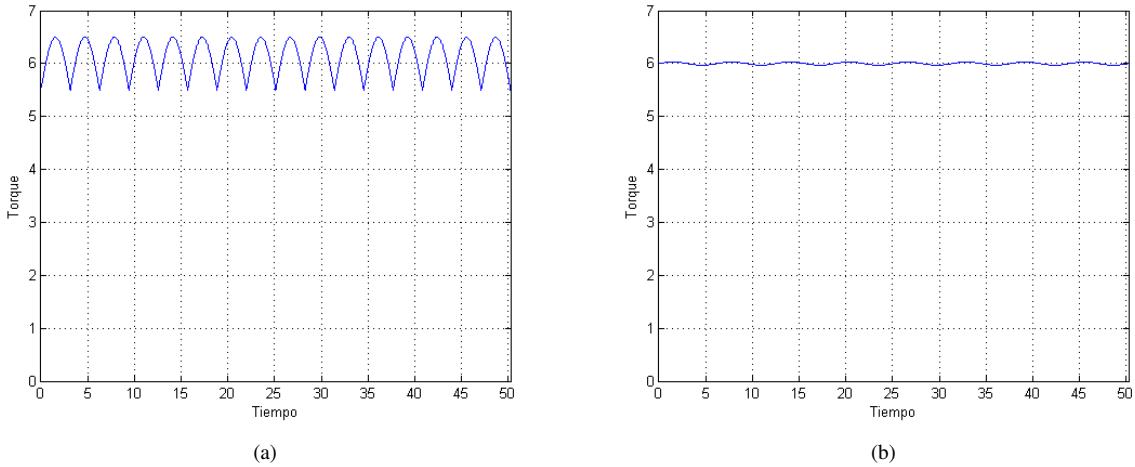


Figura VII.2.7.: *Torque resultante de conmutación: a) Cuadrada, b) Senoidal*

Habiendo cubierto los conceptos de la señal teórica, nos avocamos al análisis de la implementación en un microcontrolador. Ambos esquemas son realizables utilizando modulación PWM de alta frecuencia, generalmente $f \geq 16\text{kHz}$ para que no sea audible. Variando el *duty cycle* de la señal se puede controlar la amplitud de la señal debido a que los bobinados (modelados como un circuito R-L) actúan como pasabajos, extrayendo el valor de continua del PWM. Para la conmutación cuadrada, basta con almacenar 6 estados para caracterizar la secuencia, donde cada estado es compuesto de tres bits que determinan si la fase está encendida o apagada.

Por otra parte, la generación de un seno puede realizarse de dos maneras: calculando el valor de la función cada vez que se lo necesite, o buscando el valor más cercano en una tabla con valores precalculados. El primer enfoque es el más preciso pero también el más costoso en términos de tiempo y complejidad computacional. Por su parte, el método de la tabla es muy rápido, puesto que son simples accesos a memoria, pero puede generar errores si hay pocos puntos en el arreglo. Una razón muy fuerte para elegir almacenar una gran cantidad de puntos es para aumentar la precisión angular de los movimientos del motor (ec. VII.2.14, donde N es la cantidad de puntos del arreglo). Si comparamos este valor con la obtenida a partir del esquema de conmutación cuadrada (ec. VII.2.15), vemos que esta última es fija puesto que depende únicamente de la cantidad de pares de polos ($\#pp$) del motor.

$$Precisión_{seno} = \frac{360^\circ}{\#pp \cdot N} \quad (\text{VII.2.14})$$

$$Precisión_{cuadrada} = \frac{360^\circ}{\#pp} \quad (\text{VII.2.15})$$

	Potencia RMS	Torque	Memoria	Precisión
Senoidal	a	Pulsante	Baja	$\frac{360^\circ}{\#pp}$
Cuadrada	$1,06 \cdot a$	Continuo	Alta	$\frac{360^\circ}{\#pp \cdot N}$

Cuadro VII.1.: Resumen de las características de los esquemas de conmutación

La Tabla VII.1 resume las características de los dos esquemas y ayuda al análisis de la mejor solución. En este trabajo, se optó por utilizar la conmutación senoidal y lo que resta de esta subsección se dedicará a mostrar cómo se implementó este método.

Para comenzar, se define la estructura *BLDC* que representa al motor real con las propiedades que se detallan en la Figura VII.2.8. Las variables físicas son las que caracterizan al motor y se modifican durante la configuración inicial del sistema. Las variables de estado, en cambio, están en constante movimiento porque participan activamente del movimiento del motor y son invisibles al usuario. Lo mismo ocurre con las variables internas, aunque con un grado más de abstracción porque no representan cantidades físicas.

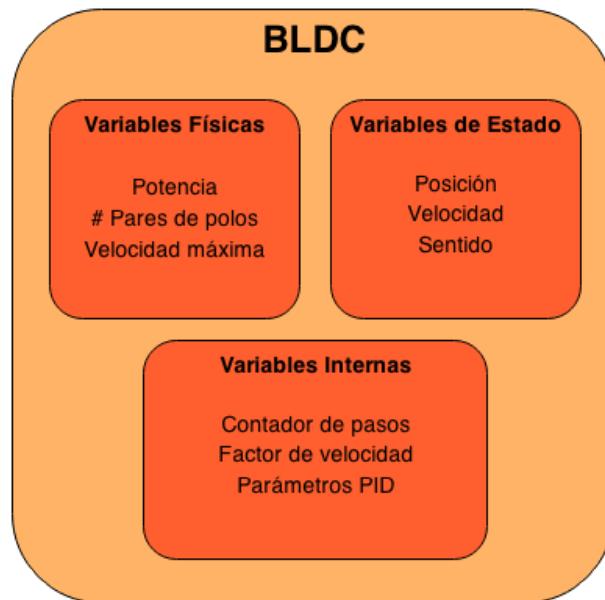


Figura VII.2.8.: Abstracción de software del motor BLDC

Las primeras dos variables físicas son cargadas por el usuario, mientras que la tercera se calcula con la fórmula VII.2.16. Como podemos ver, es la expresión VII.2.14 multiplicada por f , la frecuencia de refresco de la rutina que recorre el arreglo del seno. Esta cota para la velocidad es demasiado permisiva si se elige una frecuencia en los kHz, porque es de esperarse que el motor a lazo abierto comience a saltarse pasos al no poder seguir el ritmo de conmutación. Para obtener una restricción más sensata, se debe tener en cuenta la potencia del motor, que determinará cuánta fuerza pueden ejercer los bobinados sobre el rotor para atraerlo. Mediante pruebas que consistieron en llevar el motor a la máxima velocidad sin perder el sincronismo (aún cuando se aplicaba un torque frenante), se obtuvo la ecuación VII.2.17, donde P es la potencia del motor.

$$\omega_{MAX} = \frac{360^\circ}{\#pp \cdot N} \cdot f \quad (\text{VII.2.16})$$

$$\omega_{MAX_{MOD}} = \omega_{MAX} \cdot \frac{P}{0,476} \quad (\text{VII.2.17})$$

Para analizar el rol que cumple la potencia, primero debemos aclarar que los valores del arreglo del seno son de 8 bits y representan una senoidal de amplitud 255 con un *offset* de 127, de manera que todos los

valores con positivos. Ahora, se nos plantean dos posibles alternativas para escalar estos valores en función de la potencia deseada, que se ilustran en las Figuras VII.2.9a y VII.2.9b.

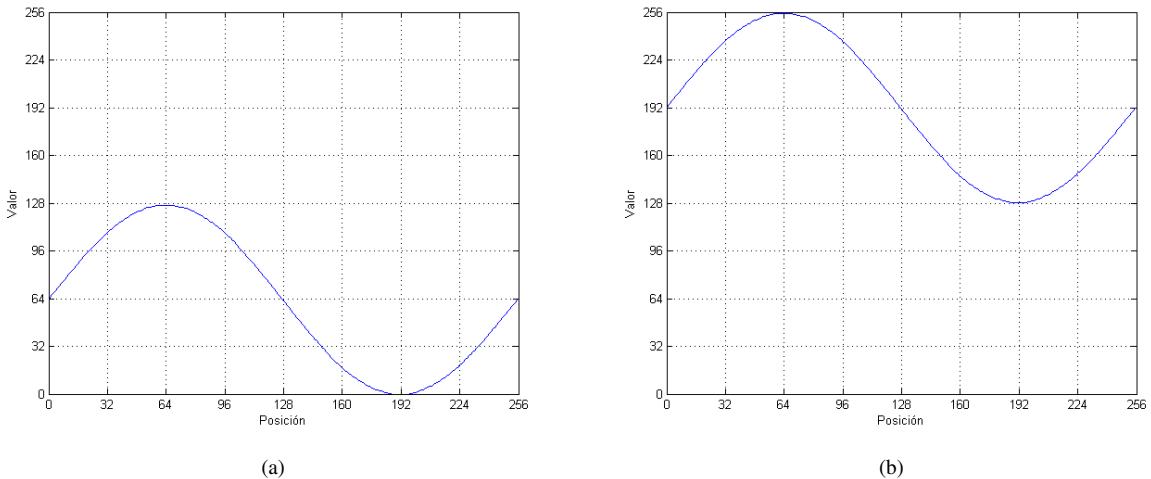


Figura VII.2.9.: Escalamiento del seno: a) Normal bajo, b) Normal alto

La ventaja del gráfico normal bajo es que consume menos corriente porque tiende a producir valores de tensión bajos, aunque también es su mayor debilidad. Como el torque generado es proporcional a la corriente, significa que, para el mismo valor de P (a excepción de 255), levantará menos peso que el esquema normal alto. Este último también tiene la ventaja de estar siempre energizado, por lo que tiene un torque de sostén que inmoviliza la cámara en todo momento (en el otro esquema, la tensión siempre pasa por la zona baja y se hace débil). Por esta razón, se elige el normal alto, que se implementa con la ecuación VII.2.18, donde p_n es el valor del arreglo en la posición n .

$$P_{eff} = p_n \cdot \frac{P}{255} + (255 - \frac{P}{2}) \quad (\text{VII.2.18})$$

Para desarrollar el control de comutación, primero se debe calcular la relación entre los grados eléctricos y los mecánicos, que surge de aplicar la ecuación VII.2.19 con los datos de las variables físicas.

$$r = \frac{360^\circ}{\#pp} \quad (\text{VII.2.19})$$

Con esta información podemos calcular cuántos grados mecánicos representa un incremento en el arreglo del seno (de ahora en más, se referirá al incremento como “paso”). De esta manera, para calcular el número de pasos que se debe avanzar para moverse una posición deseada, se aplica la ecuación VII.2.21. La velocidad de movimiento se regula con la frecuencia a la que se actualiza la posición del puntero al arreglo.

$$\theta_{paso} = \frac{r}{N} \quad (\text{VII.2.20})$$

$$n^{\circ} \text{ pasos} = \frac{\theta_{deseado}}{\theta_{paso}} \quad (\text{VII.2.21})$$

VII.2.1.2 Interfaz con la computadora

El intercambio de información entre la MCU y la PC se realiza a través de SCI utilizando un protocolo personalizado, que se basa en mandar paquetes de la forma que se presenta en la Figura VII.2.10. Los

encabezados (H1 y H2), separadores (S1 y S2) y terminadores (T1 y T2) se componen de 2 bytes para evitar que un byte de datos se confunda con uno reservado. Se definió de esta manera para permitir el intercambio de datos en bloque, para que el usuario haga varias modificaciones en la GUI de la PC y luego envíe las órdenes. Sin embargo, también soporta la transmisión de un solo dato, aunque con una eficiencia menor. Por ejemplo, si envía un entero de 16 bits con un byte de identificación, tendrá un *overhead* de 133 %, que baja al 50 % si manda 4 enteros con identificador.



Figura VII.2.10.: Esquema de un paquete de datos

En cualquier caso, el programa maneja la comunicación se lo hace a través de interrupciones de hardware (tanto en la MCU como en la PC), por lo que no produce complicaciones con el flujo del programa principal, sea cual fuere el volumen de datos. Para decodificar estos paquetes se diseñó una máquina de estados, cuyo diagrama se muestra en la Figura VII.2.11.

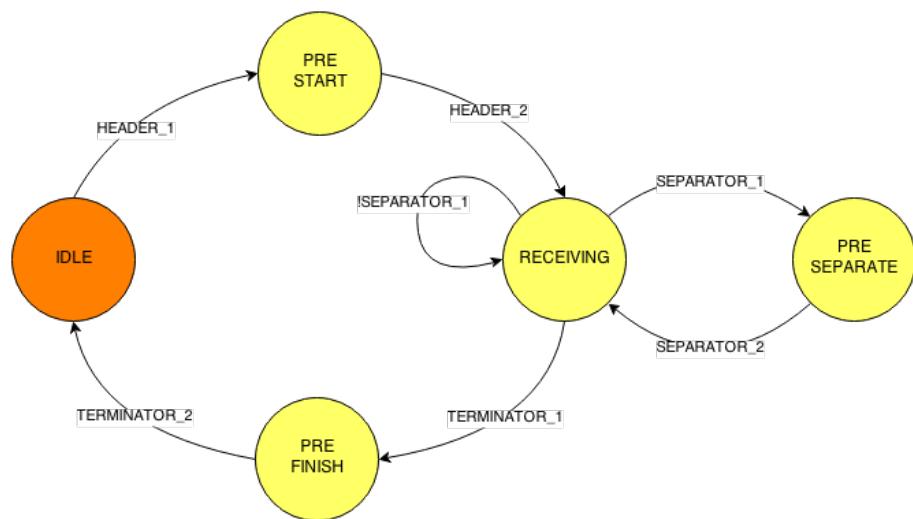


Figura VII.2.11.: Diagrama de estados del protocolo de comunicación PC-MCU

VII.2.1.3 Control remoto

El software del control remoto se divide en tres bloques: Medición, Procesamiento y Control, y Comunicaciones, como puede verse en la siguiente Figura(VII.2.12). A continuación se explicarán cada uno de estos bloques.

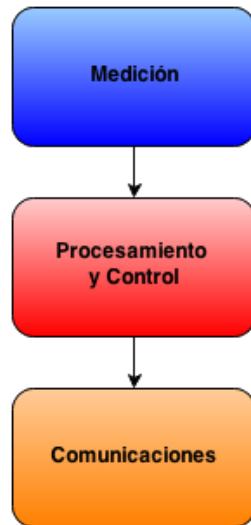


Figura VII.2.12.: Diagrama de bloques del control remoto.

Bloque de Mediciones

El bloque de mediciones se encarga de obtener los valores de los datos suministrados por las entradas analógicas y digitales. Este bloque se comunica con los dispositivos de entrada y obtiene los datos de las posiciones de los potenciómetros y el estado de los botones.

Bloque de Procesamiento y Control

El bloque de procesamiento y control se encarga de procesar los datos recibidos del bloque de Mediciones y de Comunicaciones, y en función de estos realizar distintas acciones. En una primera instancia, el control remoto se comporta como una maquina de estados, la cual se muestra en la Figura VII.2.13 presentada a continuación.

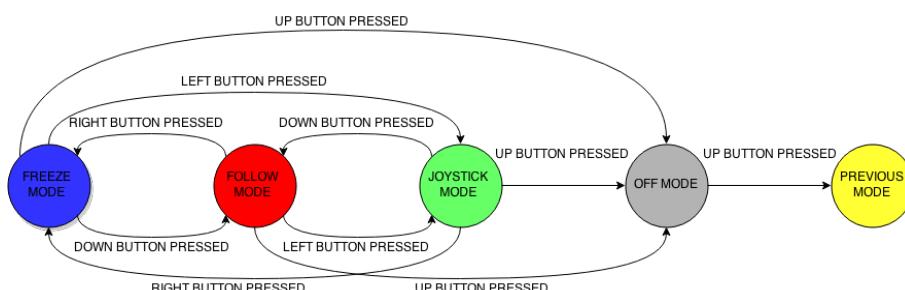


Figura VII.2.13.: Maquina de Estados del control remoto

Como se puede apreciar, la maquina cuenta con cuatro estados (Freeze Mode, Follow Mode, Joystick Mode y Off Mode), y cuatro acciones (Up Button Pressed, Down Button Pressed, Left Button Pressed y Right Button Pressed). Los estados activos, corresponden a los distintos modos de operación deseados, y para acceder a estos se debe presionar el botón izquierdo, inferior, o derecho respectivamente. Al presionar el botón superior, se entra en el estado Off. En este estado, se ignoran todos los eventos, excepto el correspondiente a presionar el botón superior. Si se presiona el botón superior en el estado Off, se vuelve al ultimo estado activo anterior a este. Finalmente, si se presiona el botón que activa el estado activo actual, el evento se ignora.

Cada evento de software, corresponde a una acción en hardware. En la Figura VII.2.14 presentada a continuación, se aprecia el algoritmo de la máquina de estado descripta anteriormente.

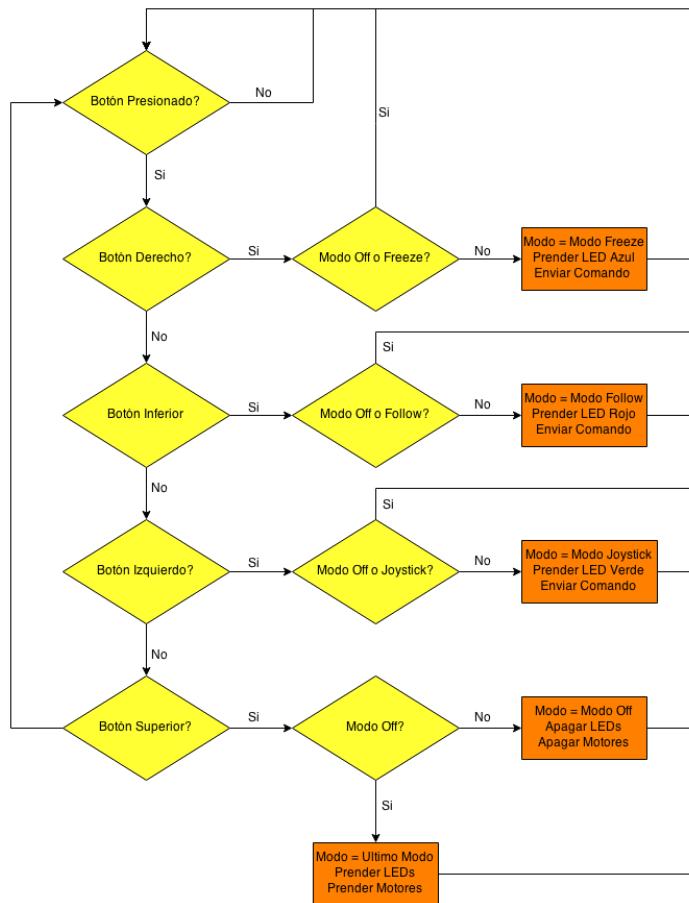


Figura VII.2.14.: Algoritmo del control remoto

Como se puede ver en la Figura, el el Algoritmo procesa los eventos de los botones. En caso de que se presione alguno de los botones correspondientes a los estados activos, y no nos encontramos en dicho modo, o en el modo Off, se debe actualizar el Modo actual, encender el LED correspondiente, y enviar a la placa de control el comando que indica en que modo se encuentra.

En caso de que se presione el botón superior, se pregunta si se esta en el modo Off, en caso negativo, se procede a actualizar el modo actual, apagar los LEDs, y a apagar los motores del *gimbal*. En caso afirmativo, se actualiza el Modo para que sea igual al ultimo estado activo, se procese a encender el LED correspondiente, y se encienden los motores del *gimbal*.

Finalmente, el main del control remoto se muestra en la Figura presentado a continuación.

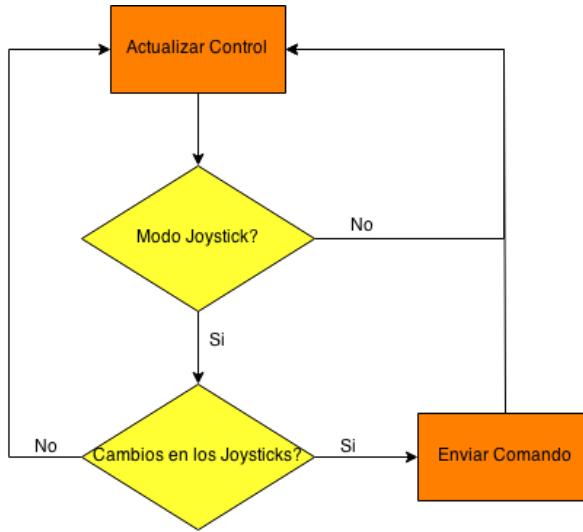


Figura VII.2.15.: Main del control remoto

Como se puede apreciar en la Figura, en el main se actualiza constantemente el estado de las distintas variable de control. En caso de que se encuentre en el modo Joystick, se analiza si este sufrió algún cambio, y en caso de que así sea, se envía este comando a la placa de control.

Bloque de Comunicaciones

El bloque de comunicaciones es el que permite la transmisión, y recepción de datos entre el control remoto, y la placa de control. Debido a que se desea compatibilidad entre los distintos módulos, y que el modulo Xbee funciona a través de UART/SCI, este bloque fue realizado de la misma forma en que el bloque que comunica la placa de control con la computadora. Dicho modulo fue desarrollado en la sección VII.2.1.2

VII.2.2 Análisis de complejidad

Para analizar las características del software generado se utiliza una de las métricas con mayor aceptación, la de complejidad ciclomática. Este proceso de evaluación se basa en la contabilización de estados posibles, caminos totales y procesos independientes de un programa. Para hacer la medición se necesitan plantear todos los diagramas de flujo de las rutina y aplicar la fórmula VII.2.22 o VII.2.23. La primera ecuación sirve para gráficos fuertemente conectados, es decir, los que son cílicos. Por su parte, la otra fórmula se aplica cuando se tienen nodos de salida que no vuelven al inicio.

$$M = E - N + P \quad (\text{VII.2.22})$$

$$M = E - N + 2P \quad (\text{VII.2.23})$$

donde

M es la complejidad

E es el número de caminos (flechas)

N es el numero de estados (nodos)

P es el número de nodos de salida

En este trabajo podemos diferenciar tres procesos que ocurren en paralelo: el control de la inclinación, la comunicación con la PC y en enlace con el control remoto. La complejidad del primero puede evaluarse

combinando los diagramas de flujo de las Figuras VII.2.2, VII.2.3 y VII.2.4. Así, tenemos que $E = 23$, $N = 20$ y $P = 1$, por lo que la complejidad resulta:

$$M_{inclinación} = 23 - 20 + 1 = 4 \quad (\text{VII.2.24})$$

Por otro lado, si analizamos el diagrama de la Figura VII.2.11, vemos que $E = 7$, $N = 5$ y $P = 1$, resultando en:

$$M_{PC\ link} = 7 - 5 + 1 = 3 \quad (\text{VII.2.25})$$

Finalmente, para determinar la complejidad del software del control remoto utilizamos los gráficos VII.2.13 y ???. Debido a que el proceso de refresco del *display* se ejecuta en paralelo al de recepción de datos, se analizan por separado. Para el primero, $E = 11$, $N = 8$ y $P = 1$, mientras que para el segundo $E = 12$, $N = 8$, $P = 1$:

$$M_{display} = 11 - 8 + 1 = 4 \quad (\text{VII.2.26})$$

$$M_{RC\ link} = 12 - 8 + 1 = 5 \quad (\text{VII.2.27})$$

Como podemos ver de los resultados de VII.2.24, VII.2.25 y VII.2.26, las complejidades no superan el valor de 10, que es un número que se considera aceptable según el creador de la métrica (McCabe) y asegura que son programas sin mucho riesgo. La clave está en la simpleza del código, que se logra planteando los grafos mínimos que efectúan las tareas que se desean.

VII.2.3 Descripción de subrutinas

VII.2.3.1 Unidad Controladora Principal

El software de la unidad central debe mantener 6 procesos activos, que comprenden algoritmos de manejo de datos y actualización del estado de los periféricos. La gran diferencia entre los dos radica en su complejidad: un común denominador en las rutinas que surgen eventos de hardware es la escasa cantidad de instrucciones, que tiene como propósito minimizar la fragmentación del hilo del programa. Al no poder atender una interrupción dentro de otra, un mayor tiempo de ejecución corresponde a una probabilidad de perder otros eventos más elevada.

En consecuencia, los programas de interpretación de información necesitan suplir la falta de procesamiento de las interrupciones, por lo que son más extensos y se ejecutan en el programa principal. Debido a la alta tasa de despacho de instrucciones, pueden ser interrumpidas sin causar pérdida de información.

Puerto serie

El microcontrolador posee dos *buffers* de 1 byte de entrada y salida, que disparan una interrupción siempre que reciben un nuevo dato. La subrutina que atiende este evento carga el dato en una cola de entrada, en caso de recepción, o lo deposita en una de salida en caso de ser una transmisión. El programa principal verifica constantemente el estado del buzón de entrada y al encontrar un nuevo dato, lo procesa con una máquina de estados de decodificación.

Decodificador

El programa de interpretación de datos del puerto serie implementa la máquina de estados de la Figura VII.2.11 utilizando una matriz de transición: las columnas representan el estado actual, las filas el evento

ocurrido y las celdas poseen información del estado siguiente y la acción a tomar.

Debido a la forma de los paquetes, los datos se guardan en colas de entrada y salida: en la transmisión, sólo se necesita “empujar” los datos y el programa se encarga de colocar el encabezado, el separador y el terminador. En la recepción, los mensajes se cargan en la cola una vez que se verificó que estén sanos y se notifica al programa principal.

PWM

El módulo de PWM cuenta con un contador de 32-bit que se recorre a una frecuencia establecida y dispara una interrupción que cambia el estado de la señal de salida cuando el valor registrado coincide con uno cargado previamente. Estas características hacen que también oficie de interrupción periódica que actualiza los procesos del controlador.

La frecuencia de la señal se eligió en 32kHz por estar encima del rango audible y por ser fácilmente obtenible con un cristal de cuarzo de 32.768kHz. El valor elevado hace que las rutinas de los distintos programas posean un contador para disminuir la frecuencia de ejecución.

Controlador de BLDC

VII.2.3.2 Control remoto

Se manera similar a la unidad central, el software de control remoto se puede separar en cinco bloques definidos los cuales se pueden separar en dos tipos, los controles de periféricos, y el algoritmo de control. Los controles de periféricos son: adquisición de datos analógicos (ADC) y digitales (Puerto A), comunicación inalámbrica (UART/SCI), y control de los LEDs(PORT C). El algoritmo de procesamiento es el que se encarga de interpretar el estado actual de los periféricos y ejecuta las acciones en función de eso.

Adquisición de datos digitales (PORT A)

Esta subrutina se encarga de tomar el valor físico de tensión actual de los joysticks y potenciómetro, y convertirlos en valores digitales. Debido a que nos interesa que los botones tengan un excelente tiempo de respuesta, esta subrutina se implementa utilizando una interrupción por flanco de alta prioridad. Esto nos permite actualizar los valores del estado de los botones cuanto se detectan los flancos ascendentes y descentente de los mismos.

Adquisición de datos analógicos (ADC)

Esta subrutina se encarga de tomar el valor físico de tensión actual de los joysticks y potenciómetro, y convertirlos en valores digitales.

Si cualquiera de los 3 potenciómetros (El joystick o el radial) se encuentran en su punto medio (es decir, no se están activando) el ADC debería leer 128 , y esto se debe traducir en un 0 lógico. El problema es que el ruido del sistema puede causar que la lectura del ADC tenga un desvío de algunos bits en su medición, en consecuencia, para eliminar el efecto del ruido en la medición se establecieron umbrales en los valores de conversión del ADC según:

$$\begin{aligned} 0 \leq \text{Val}_{\text{ADC}} < 59 &\longrightarrow \text{Punto Negativo} \\ 59 \leq \text{Val}_{\text{ADC}} < 197 &\longrightarrow \text{Punto Medio} \\ 59 \leq \text{Val}_{\text{ADC}} \leq 255 &\longrightarrow \text{Punto Positivo} \end{aligned}$$

Internamente el ADC se actualiza con una interrupción periódica, y de duración controlada.

Control de los LEDs (PORT C)

Esta es la subrutina mas sencilla. Consiste simplemente en la escritura de uno de los puertos (Puerto C) para el encendido o el apagado de los LEDs correspondientes. En consecuencia, esta subrutina no esta implementada con interrupciones de hardware, sino que eventos de software generan la llamada de la misma.

Comunicación Inalámbrica (UART/SCI)

Debido a que la comunicación se realiza utilizando el modulo Xbee anteriormente descrito, esta subrutina consiste sencillamente en la comunicación con este via UART/SCI. La subrutina funciona de la misma manera descrita en la parte VII.2.3.1 de la sección VII.2.3.1. Debido a que el control remoto funciona solo como transmisor, la subrutina solo requiere un buffer de salida.

Algoritmo de control

El algoritmo de control se encarga de interpretar los cambios ocurridos en los periféricos, y actuar en función de ello. Esta subrutina fue descrita en su totalidad en la sección VII.2.1.3

VII.2.4 Listados comentados de código

VII.2.4.1 Control de inclinación

```
#ifndef BLDC_H_
#define BLDC_H_

#include "PE_Types.h"
#include "general.h"

//***** Definitions *****/
// BLDC motors generally have 3 phases //
#define NUM_OF_PHASES      3

// Number of elements of the sine lookup tables
#define N_SIN      256

// Value in [deg/sec]
#define DEFAULT_SPEED      (PWM_FREQ/N_SIN)*360

// Decimal point shifted 4 times to the right //
#define ELECTRICAL_DEGREES_PER_STEP    ((360*10000)/N_SIN)

// Offset needed to produce sines shifted 120 deg from each other //
#define OFFSET      N_SIN/NUM_OF_PHASES
```

```

// Positive or negative direction //
#define BLDC_FWD      1
#define BLDC_REV     -1

// Status //
#define BLDC_FROZEN    1

// Phase indexes //
#define A      0
#define B      1
#define C      2

#define CC_FACTOR        32
#define MOTORUPDATE_FREQ 500           // In Hz. 500 is default //

#define LOOPUPDATE_FREQ  MOTORUPDATE_FREQ // Loop control sample rate //
// equals motor update rate //

#define DT_FLOAT          (1.0/LOOPUPDATE_FREQ*1.024) // loop controller sample //
// period dT           //

#define DT_INT_MS         (1000/MOTORUPDATE_FREQ) // dT, integer, (ms)       //
#define DT_INT_INV        (MOTORUPDATE_FREQ) // dT, integer, inverse, (Hz) //

// Gimbal mode //
typedef enum
{
    JOYSTICK      = 0,
    FOLLOW        = 1,
    FREEZE        = 2,
} Mode;

// PWM channel //
typedef struct
{
    uint8_t value;
    int8_t module;
    uint8_t channel;
} PWM;

typedef struct
{
    // PID constants //
    int16_t Kp;
    int16_t Ki;
    int16_t Kd;
}

```

```

// Set point //
int32_t setPoint;
int16_t setPointDir;
uint16_t setPointSens;
int16_t setPointMax;

// Difference between set-point and current angle //
int32_t error;

// Auxiliary variables //
int32_t errorSum;
int32_t lastError;

// Anti-windup guard //
int32_t windupGuard;

} PID;

typedef struct
{
    // PWM channels //
    PWM phase[3];                                // Array that stores the PWM ports
                                                    // assigned to each phase (MCU specific) //

    // Status & Position control //
    int16_t currentAngle;                         // Current angle of the motor
                                                    // (updated every 1/refreshRate) seconds //

    int16_t direction;                            // Direction of rotation
    int8_t invert;
    bool enabled;
    bool frozen;                                 // Locks the motor in position
                                                    // (useful when calibrating) //

    // Speed [deg/s] //
    uint16_t speed;                             // Current speed of the motor //

    uint16_t maxSpeed;                          // Maximum speed (in [deg/s]), determined
                                                // by the size of the sine wave array
                                                // and the RTI frequency //

    uint16_t speedFactor;

    // Power //
    uint8_t power;                             // Motor power in a scale [0,255] //

```

```

    uint16_t offsetPower;           // Use a variable to avoid doing this operation //
                                // every time the sine waves are scaled          //

    // Number of poles      //
    // Must be divisible by 2) //
    uint8_t num0fPoles;

    // Phase cycle control //
    int16_t steps;                // Keeps track of position          //

    int16_t stepsLeft;            // Tracks the counter progress       //

    int16_t stepCounter;          // Counter to loop through the array //

    uint16_t speedCounter;        // Counter used for RTI frequency division //

    uint16_t degreesPerStep;      // Mechanical degrees per          //
                                // step in sine wave array          //

    // PID parameters //
    PID pidParams;

} BLDC;

//***** Initialization *****

// Initializes motors and PWM outputs //
void BLDC_Init(void);

// Enable and disable output //
void BLDC_EnableOutput(uint8_t motor);
void BLDC_DisableOutput(uint8_t motor);
void BLDC_ToggleOutput(uint8_t motor);

// Freeze the motor //
void BLDC_Freeze(uint8_t motor);
void BLDC_Unfreeze(uint8_t motor);
void BLDC_ToggleFreeze(uint8_t motor);

// Set status and position control to 0 //
void BLDC_ResetStatus(uint8_t motor);

// Reset all counters to 0 //
void BLDC_ResetCounters(uint8_t motor);

```

```

//***** Settings *****/


---


// Set the number of poles (very important!) //
void BLDC_SetNumberOfPoles(uint8_t motor, uint8_t numOfPoles);

// Set the motor power (number must be [0,255]) //
void BLDC_SetPower(uint8_t motor, uint8_t power);

// Gimbal mode //
void BLDC_SetMode(int16_t md);
Mode BLDC_GetMode(void);

// Set PID values //
void BLDC_SetPID(uint8_t motor, uint16_t Kp, uint16_t Ki, uint16_t Kd);
void BLDC_SetErrorSum(uint8_t motor, uint16_t value);
void BLDC_SetLastError(uint8_t motor, uint16_t value);
void BLDC_SetKp(uint8_t motor, uint16_t Kp);
void BLDC_SetKi(uint8_t motor, uint16_t Ki);
void BLDC_SetKd(uint8_t motor, uint16_t Kd);
void BLDC_SetWindupGuard(uint8_t motor, int32_t windupGuard);

// Sets the position that the motor should maintain //
void BLDC_SetSetPoint(uint8_t motor, int16_t setPoint);
void BLDC_SetSetPointSensitivity(uint8_t motor, uint16_t sensitivity);
void BLDC_UpdateSetPoint(uint8_t motor);
void BLDC_UpdateSetPointDirection(uint8_t motor, int16_t direction);
void BLDC_ResetSetPoint(void);

// Invert direction //
void BLDC_InvertDirection(uint8_t motor);

// Flag that runs at 500Hz and is used as periodic interrupt //
void BRUGUI_SetMotorUpdate(bool state);
bool BRUGUI_GetMotorUpdate(void);

// Moves the motors at predefined angle and with certain power //
void BRUGUI_MoveMotor(uint8_t motorNumber, int MotorPos, uint16_t maxPWM);

// PID computation //
int32_t BRUGUI_ComputePID(uint8_t motor, int32_t in);

// Calculates the required motor action to corrent tilt //
void BLDC_CalculateCorrection(void);

// Updates the PWM values //
void BRUGUI_ISR(void);
#endif

```

VII.2.4.2 Interfaz con la computadora

```
#ifndef COMM_H
#define COMM_H

#include "general.h"
#include "queue.h"

//***** Definitions *****/
// Motor IDs //
#define KP 0
#define KI 1
#define KD 2
#define POWER 3
#define POLES 4

#define INVERT 7
#define ENABLE 8
#define FROZEN 9
#define SET_POINT 10
#define MODE 14
#define PID_MODE 13

#define CONTROL_MODE 14

// Sensor IDs //
#define CALIBRATE_ACCEL 11
#define CALIBRATE_GYRO 12
#define GYRO_TRUST 5

// RC IDs //
#define RC_SENS 6

// Ports //
#define PC 0
#define RC 2

// State machine specs //
#define COMM_NUM_STATES 5
#define COMM_NUM_EVENTS 7

// Special characters //
#define HEADER1 '$'
#define HEADER2 '%'
#define SEPARATOR1 '+'
#define SEPARATOR2 '_'
```

```

#define TERMINATOR1      '@'
#define TERMINATOR2      '&'

//***** Decoder State Machine *****

// States //
typedef enum
{
    IDLE      = 0,
    PRE_START = 1,
    RECEIVING = 2,
    PRE_SEPARATE = 3,
    PRE_FINISH = 4
} CommState;

// Actions //
typedef enum
{
    DO_NOTHING      = 0,
    PUSH_CHAR       = 1,
    PUSH_TERMINATOR1 = 2,
    PUSH_SEPARATOR1 = 3,
    PUSH_TERMINATOR1_AND_CHAR = 4,
    PUSH_SEPARATOR1_AND_CHAR = 5,
    DUMP_BUFFER     = 6,
    SET_DATA_READY = 7
} CommAction;

// Events //
typedef enum
{
    HEADER_1      = 0,
    HEADER_2      = 1,
    CHAR          = 2,
    SEPARATOR_1   = 3,
    SEPARATOR_2   = 4,
    TERMINATOR_1  = 5,
    TERMINATOR_2  = 6
} CommEvent;

// Transition matrix cell //
typedef struct
{
    CommState nextState;
    CommAction actionToDo;
} CommElement;

```

```

// Communications structure //
typedef struct
{
    Queue inBox;
    Queue outBox;
    CommState currentState;
    uint8_t commandReady;
} Comm;

//***** Functions *****/
// Initializes the input and output queues and sets the initial state
void commInit(void);

// Evaluates the next state and action to perform depending on current state and event
CommAction commStateEval(uint8_t commType, CommEvent event);

// Queue access functions
void pushCommand(uint8_t commType, int8_t* buffer, uint8_t len);
uint8_t popCommand(uint8_t commType, int8_t* buffer, uint8_t len);

// Writes the commands in the output queue to the serial port
void sendCommand(uint8_t commType);

// Parses the new command with the state machine
void parseCommand(uint8_t commType, int8_t character);

// Separates the received message into 4 byte commands
void processCommands(uint8_t commType);

// Decodes the 4 byte commands
void decodeCommand(int8_t* command);

// Notifies when a new message is received
bool isCommandReady(uint8_t commType);

// Clears the messsge ready flag
void resetCommandReady(uint8_t commType);

#endif

```

VII.2.4.3 Control remoto

```

#ifndef RC_H_
#define RC_H_

// Peripheral libraries //

```

```

#include "PE_Types.h"
#include "general.h"
#include "buttons.h"
#include "joystick.h"
#include "LED.h"
#include "comm.h"

//***** Definitions *****/
#define RC_NUM_STATES      4
#define RC_NUM_EVENTS       5

// Buttons ID's //
#define UP_BUTTON           0
#define DOWN_BUTTON         1
#define LEFT_BUTTON          2
#define RIGHT_BUTTON         3
/
/ Controller structure //
typedef struct
{
    Buttons buttons;
    Joystick joystick;
    LED leds;
    uint8_t mode;
} Controller;

//Operation mode ID's
typedef enum
{
    JOYSTICK      = 0,
    FOLLOW        = 1,
    FREEZE        = 2,
    OFF           = 3,
} RCState;

// FSM status //
typedef enum
{
    DO_NOTHING = 0,
    UPDATE     = 1,
} RCAction;

// FSM events //
typedef enum
{
    UP_BUTTON_PRESSED     = 0,

```

```

DOWN_BUTTON_PRESSED      = 1,
LEFT_BUTTON_PRESSED     = 2,
RIGHT_BUTTON_PRESSED    = 3,
NO_BUTTON_PRESSED       = 4,
} RCEvent;

// FSM transition matrix element //
typedef struct
{
    RCState nextState;
    RCAction actionToDo;
} RCElement;

//***** Functions *****
// Initialization //
void RC_Init(Controller *pToController);

//Update functions//
// These functions update the software structure of the controller//
// with the actual physical values acquiered from the peripherals //

void RC_UpdateButtons(Controller *pToController);
void RC_UpdateLED(Controller *pToController);
void RC_UpdateJoystick(Controller *pToController);
void RC_Update(Controller *pToController);

//Controller Finite State Machine functions //
//These functions analize the current states, the events //
//and generate a new state and a transition action      //

RCAction RC_StateEval(RCEvent event);
void RC_ProcessCommand(Controller *pToController);

//Communication functions //
//This functions send their respective    //
// commands to the main controller board //

void RC_SendPowerON(void);
void RC_SendPowerOFF(void);
void RC_SendModeUpdate(Controller *pToController);
#endif

```

VII.2.5 Plan de prueba de módulos y de depuración de software

VII.2.5.1 Control remoto

Pruebas del bloque de mediciones

Se envían las mediciones a la consola de *debuggeo* en vez de al bloque de control, de manera que permite verificar si los valores medidos son correctos.

Pruebas del bloque de procesamiento y control

Este ensayo se realiza corriendo el programa paso por paso, enviándole un grupo de valores de prueba que representen a las mediciones y los datos recibidos, para ver si son interpretados, traducidos, y transmitidos correctamente por el bloque.

Pruebas del bloque de comunicaciones

Para verificar el funcionamiento del enlace, se envían cadenas de datos de longitud arbitraria entre el control remoto y el controlador principal, verificando que se detecte la confirmación de que ha recibido un dato válido. También se simulará un corte de la comunicación, esperando que el sistema la finalice por *timeout*.

VIII Construcción del prototipo

VIII.1 Definición de los módulos

El prototipo consta de 5 partes principales: la unidad controladora principal, el sensor de inclinación, el control remoto, la estructura mecánica y la aplicación de PC.

VIII.1.1 Unidad Controladora Principal

Este dispositivo se encarga de registrar la posición de la cámara y corregirla a través de los 4 motores del *gimbal*, además de comunicarse con el control remoto inalámbrico y la computadora. Debido a su complejidad, se dividió el diseño en cuatro partes: microcontrolador, interfaz USB-UART, controladores de motores, y fuente de alimentación.

VIII.1.2 Sensor de inclinación

Este módulo debe implementarse en un PCB separado para ubicarlo en la bandeja de la cámara y, por lo tanto, debe ser lo más pequeño posible para no estorbar. Las cuatro señales de I2C se transmiten por un cable a la unidad central.

VIII.1.3 Receptor RF

El PCB del receptor sólo necesita albergar el módulo XBee y poseer cuatro señales de salida: alimentación, referencia, transmisión y recepción. Asimismo, se incluyen dos LEDs de señalización, para corroborar que esté correctamente alimentado y para indicar la intensidad de la señal.

VIII.1.4 Control Remoto

El PCB del control remoto se diseñó para que cupiera dentro de la carcasa de un control remoto de Playstation 2, el cual fue modificado para adecuarse a las necesidades. Tales modificaciones se hicieron para reducir la cantidad de aperturas para botones que presentaba, y para que haya espacio para colocar una batería y el módulo XBee.

Consta de 5 partes: de alimentación, de control, de dispositivos de entrada analógicos, de dispositivos de entrada digitales, y emisor RF.

Se diseñaron 3 PCBs para cumplir estas funciones.

VIII.1.5 Aplicación de PC

La aplicación que corre en la PC es una ejecutable de Java (.jar) que utiliza la librería gráfica SWT de Eclipse y se comunica con la controladora del *gimbal* por puerto serie. Para ejecutar el programa es

necesario contar con la última versión de Java (a la fecha de publicación, JRE 8) y el *driver* de virtualización de puerto serie (VCP CP210x).

En la Figura VIII.1.1 se presenta la ventana principal, que está dividida en 7 grupos: los 3 paneles principales controlan los parámetros de los motores, los 3 laterales permiten configurar el sensor de inclinación, la sensibilidad del control y la conexión serie, y el panel inferior muestra información relevante, controla la visibilidad del graficador y sincroniza los parámetros con la placa controladora. Los valores del PID se pueden modificar con los deslizadores o mediante las cajas de texto, mientras que los polos se pueden ingresar manualmente o con incrementos unitarios con las flechas incorporadas. A diferencia de los botones, que tienen efecto inmediato, los parámetros se descargan al *gimbal* cuando se oprime el ícono de descarga (flecha verde).

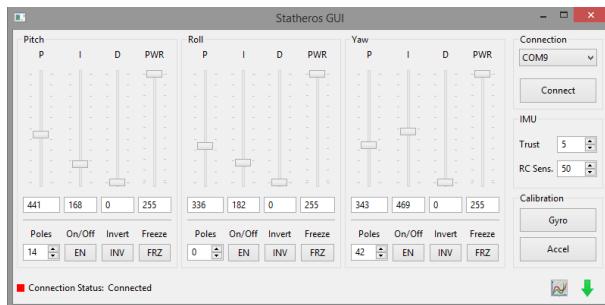


Figura VIII.1.1.: Interfaz gráfica de la aplicación de PC.

La ventana del graficador, que se muestra en la Figura VIII.1.2, permite mostrar información en tiempo real del ángulo de inclinación en los 3 ejes, al igual que las mediciones crudas de los sensores, útiles para la calibración. La interfaz también brinda facilidades para cambiar los límites de la ordenada del gráfico (manualmente o automático) y establecer la ventana de tiempo.

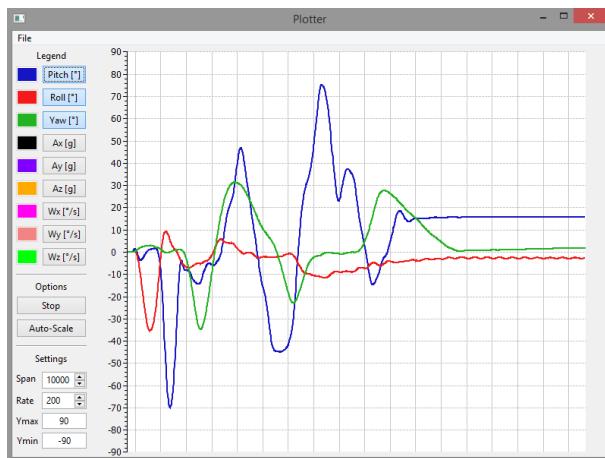


Figura VIII.1.2.: Ventana del graficador de datos

VIII.2 Diseño de los circuitos impresos

VIII.2.1 Microcontrolador

Este bloque requiere de muy pocos componentes para funcionar; como se ve en la Figura VIII.2.1, sólo necesita un cristal de 8MHz y capacitores de desacople. Lo restante se compone de resistores de *pull-up* para la comunicación I2C, un divisor resistivo para acondicionar la tensión de la batería para el ADC, un

LED de señalización y un *pull-up* con desacople para el *reset*. Por su puesto, también posee las conexiones pertinentes a cada uno de los tres módulos restantes.

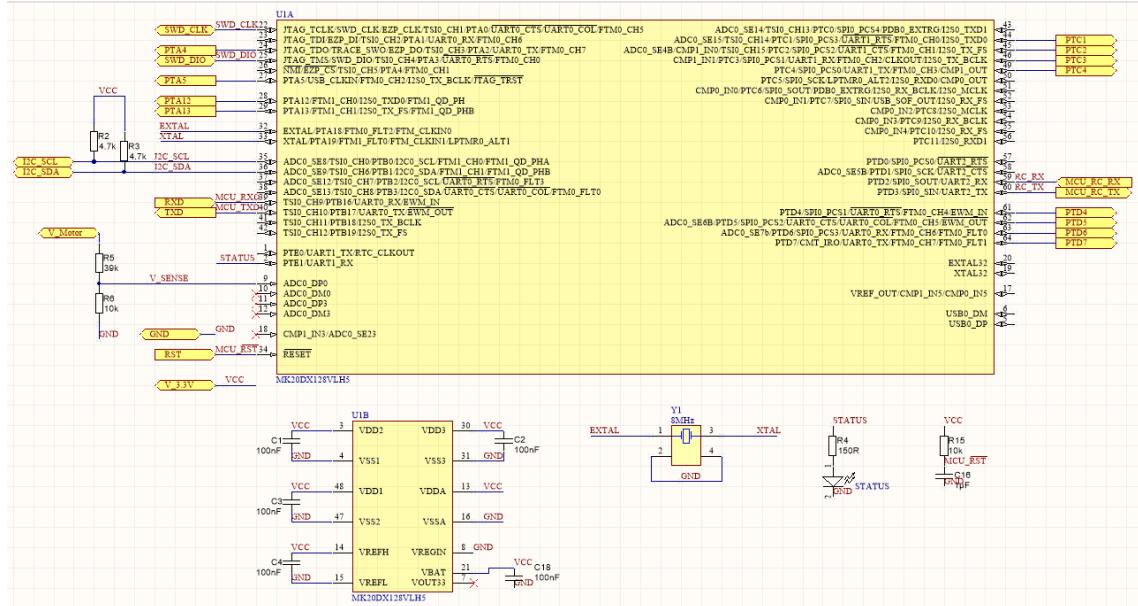


Figura VIII.2.1.: Esquemático del microcontrolador.

VIII.2.2 Interfaz USB-UART

Al igual que un microcontrolador, estos circuitos integrados sólo requieren capacitores de *pull-up* y capacitores de desacople para funcionar, como se muestra en la Figura VIII.2.2. En cuanto al conector, se optó por un Micro USB Tipo B por su popularidad en nuevos dispositivos móviles.

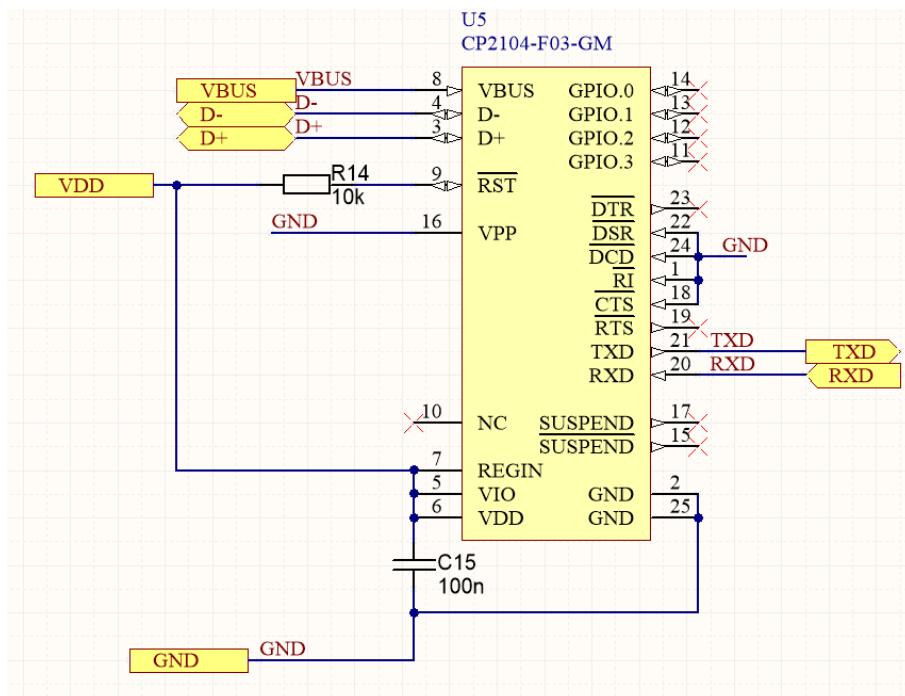


Figura VIII.2.2.: Esquemático del conversor USB-UART.

VIII.2.3 Controladores de Motor

La elección de utilizar un circuito integrado especialmente diseñado para manejar motores BLDC hizo que la cantidad de elementos externos sea reducida. Para calcular los valores y tipo de componentes se utilizaron las recomendaciones de la hoja de datos, con la excepción que se utilizaron resistores de *pull-down* en las entradas de *enable*, para asegurar que, cuando el microcontrolador tenga sus salidas de GPIO en alta impedancia, los motores queden inhabilitados. Asimismo, se decidió unir las 3 señales para detener íntegramente el motor si así se desea, en vez de inhabilitar fases individuales.

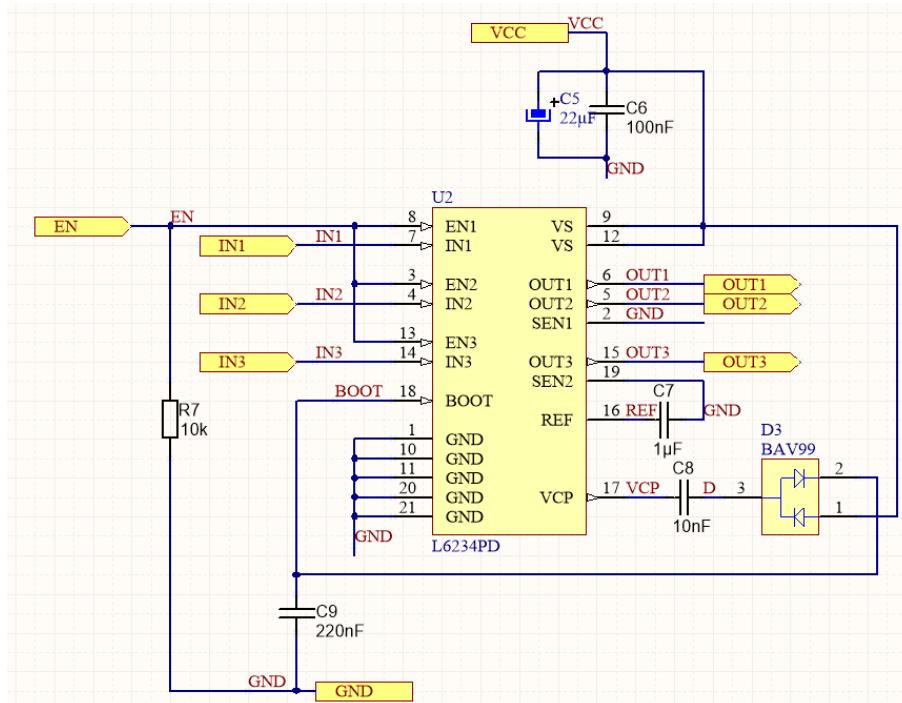


Figura VIII.2.3.: Esquemático del controlador de motores BLDC.

VIII.2.4 Fuente de alimentación

La fuente tiene 2 propósitos: proveer una tensión estable para los controladores de los motores y convertir la tensión del puerto USB y de la batería a 3,3V para alimentar el microcontrolador y sus periféricos. Para lograrlo, se emplea un convertidor DC-DC de tipo *buck* para reducir los 12V que provee la batería a 5V, que se complementa con un regulador lineal de baja caída que entrega los 3,3V.

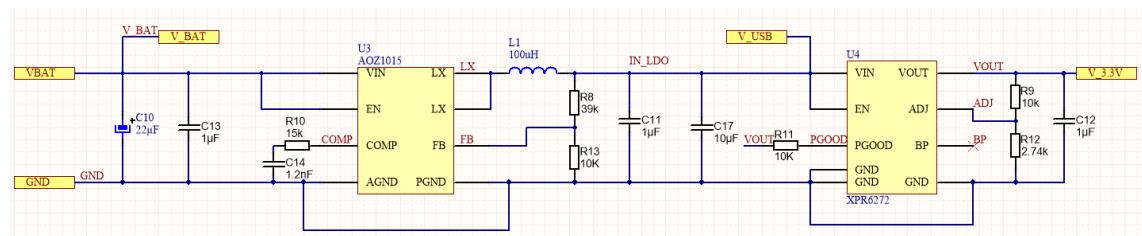


Figura VIII.2.4.: Esquemático de la fuente de alimentación.

VIII.2.5 Diseño de la placa principal

El esquema de la placa principal se muestra en la Figura VIII.2.5, donde se pueden distinguir claramente los bloques analizados anteriormente, en verde. Este diagrama muestra claramente todas las conexiones

entre el microcontrolador y los periféricos, a saber, el conversor USB-UART, los tres controladores de los motores, la placa del sensor, la placa del receptor RF y la interfaz de programación SWD.

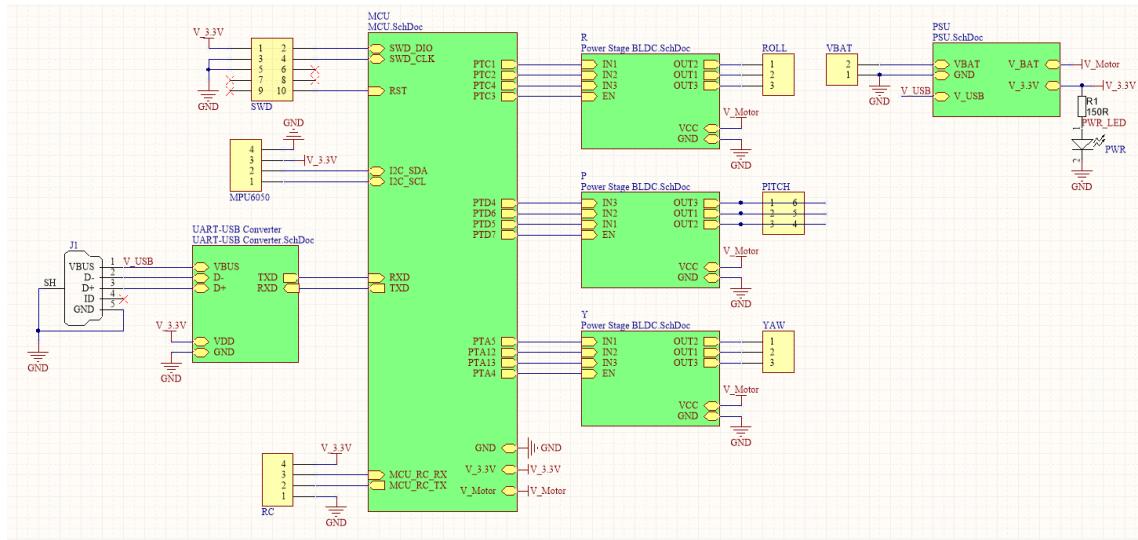


Figura VIII.2.5.: Esquemático de la placa principal.

A partir de los esquemáticos elaborados se realizó la placa de la Figura 2. Comenzando el suministro de tensión, vemos que las pistas de alimentación de 12V y de referencia están ubicadas en los bordes de la placa, en capas opuestas. Esta disposición permitió que las pistas sean anchas para soportar las altas corrientes y simplificó el diagramado de los módulos. Cabe destacar que no se cerró ningún lazo, a fin de evitar la inducción de corrientes parásitas en las posibles espiras.

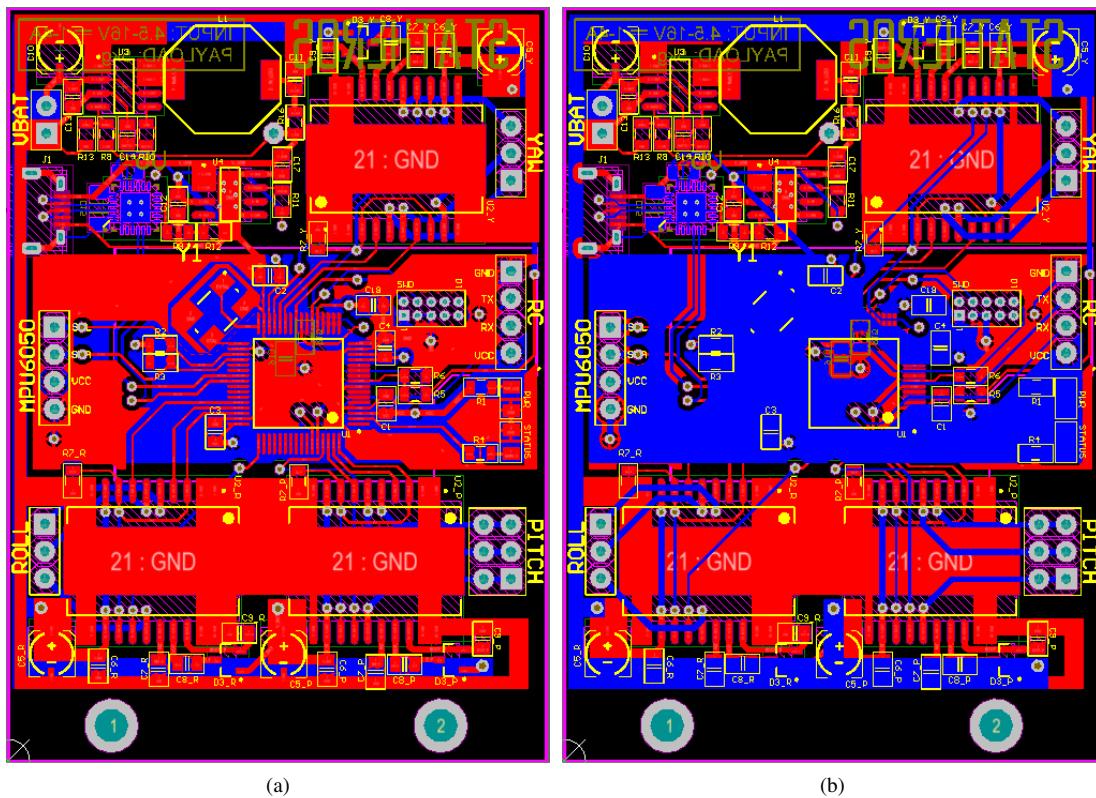


Figura VIII.2.6.: Diseño del PCB de la placa principal: a) capa superior, b) capa inferior.

Otra de las consideraciones tomadas en el diseño es la separación entre la parte lógica y de potencia. Los drivers de los motores se ubican en 3 esquinas del PCB, mientras que la última está ocupada por el regulador buck y el LDO, con los conectores para USB y la batería. El microcontrolador ocupa el centro y cuenta con un plano de masa en la capa superior y un plano de alimentación en la inferior, proveyendo desacople eléctrico. La referencia lógica se conecta con la de potencia mediante una sola pista, para evitar que cualquier sobrepico espurio contamine la unidad de procesamiento.

Desde el punto de vista de accesibilidad, se procuró colocar todos los conectores en los bordes laterales de la placa, excepto el de programación SWD, debido que sólo tiene que ser accesible durante el desarrollo. La elección de contar con un receptor/transmisor externo otorga libertad de utilizar cualquier módulo con interfaz serie y reduce el tamaño de la placa considerablemente.

En cuanto a las dimensiones del PCB, se redujo el tamaño lo más posible utilizando sólo dos capas, ya que los diseños de cuatro son más complejos y, por tanto, costosos. Los agujeros inferiores coinciden con los del soporte trasero del *gimbal* para brindar rigidez mecánica.

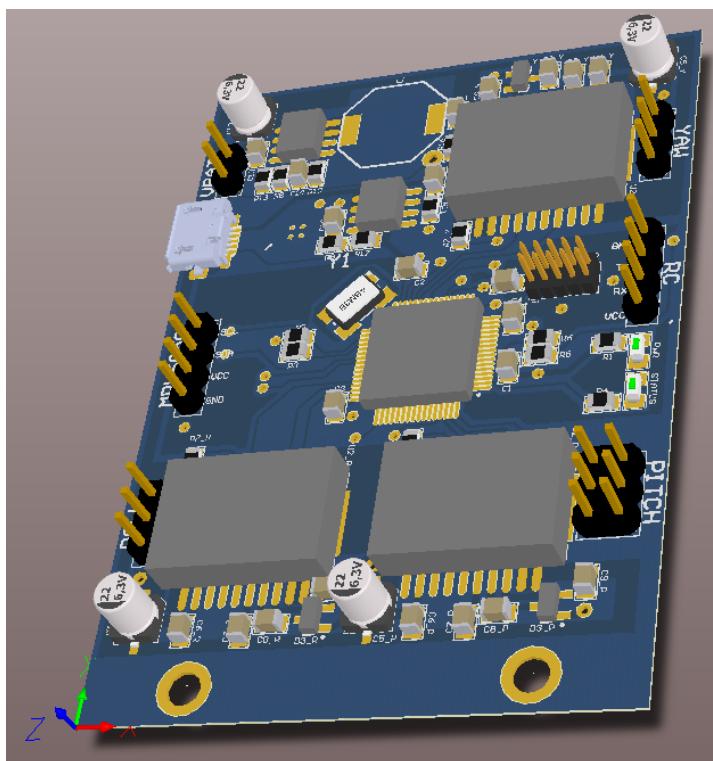


Figura VIII.2.7.: Vista 3D de la placa principal.

VIII.2.6 Diseño de las placas del control remoto

Microcontrolador

Como en la placa principal, este bloque requiere de muy pocos componentes para funcionar; sólo se necesita un cristal de 8MHz, cinco capacitores de desacople, y un *pull-up* con desacople para el *reset*. También posee las conexiones necesarias para poder comunicarse con cada uno de los tres módulos restantes.

Esto puede verse en la Figura VIII.2.8

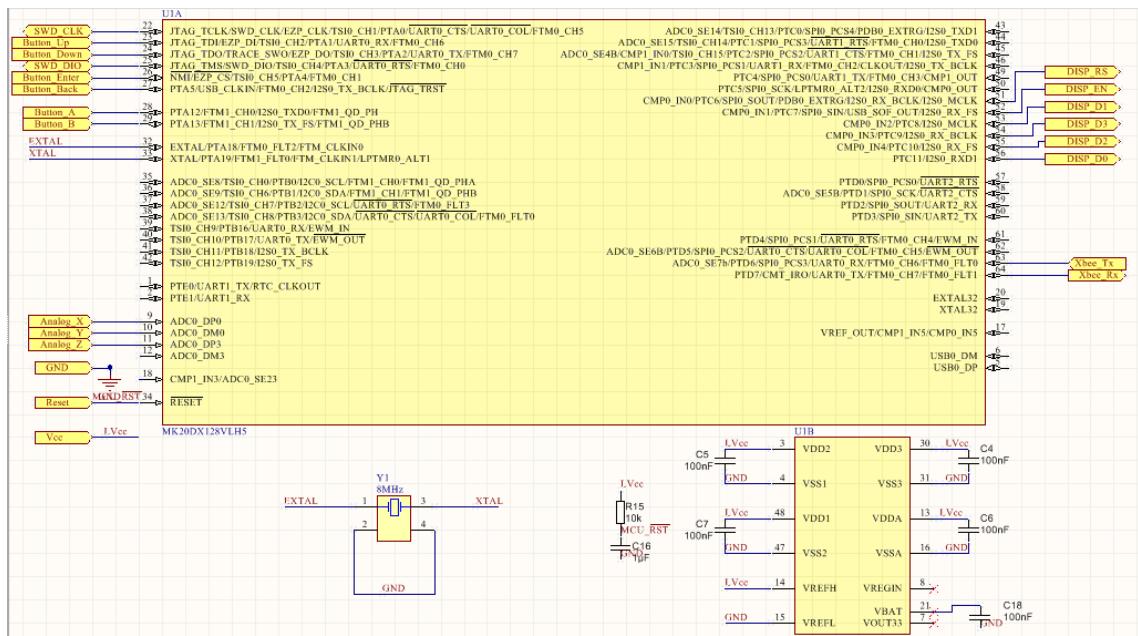


Figura VIII.2.8.: Esquemático del microcontrolador del control remoto.

Fuente de alimentación

La fuente tiene como propósito convertir la tensión de la batería a 3,3V para alimentar el microcontrolador y sus periféricos. Para lograrlo, se emplea un regulador lineal de baja caída que entrega los 3,3V.

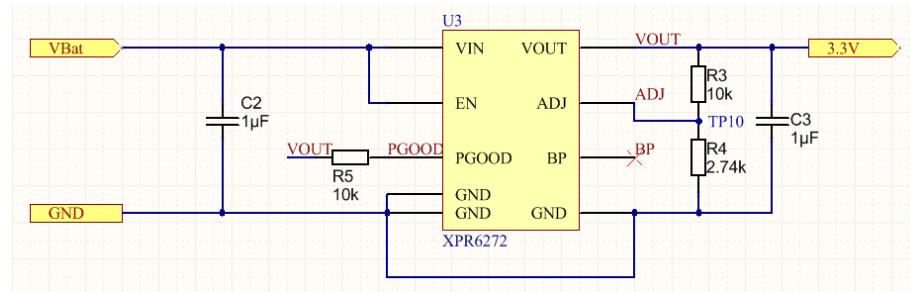


Figura VIII.2.9.: Esquemático de la fuente de alimentación del control remoto.

Dispositivos de entrada digitales

Los dispositivos digitales de entrada fueron diseñados en una sección anterior y su diseño se mantuvo sin cambios para el PCB. Se decidieron utilizar pulsadores del estilo membrana, los cuales consisten en dos contactos grabados en el PCB y una membrana conductora plástica que se coloca encima.

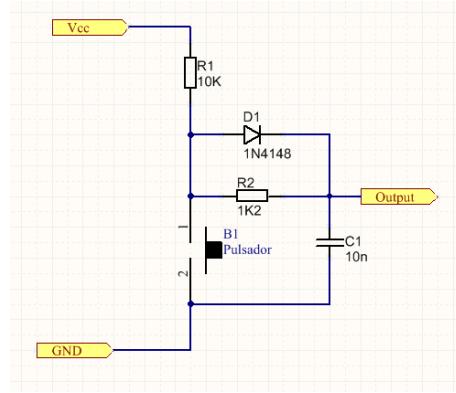


Figura VIII.2.10.: Esquemático de un botón del control remoto.

Dispositivos de entrada analógicos

Para esta sección se decidió usar la placa de joysticks analógicos que se encontraba presente dentro del control remoto, reemplazando uno de los joysticks por un potenciómetro radial.

Diseño de las placas del control remoto

El esquema de la placa principal se muestra en la Figura VIII.2.11, donde se pueden distinguir claramente los bloques analizados anteriormente, en verde. Este diagrama muestra claramente todas las conexiones entre el microcontrolador y los periféricos, la placa del emisor RF y la interfaz de programación SWD.

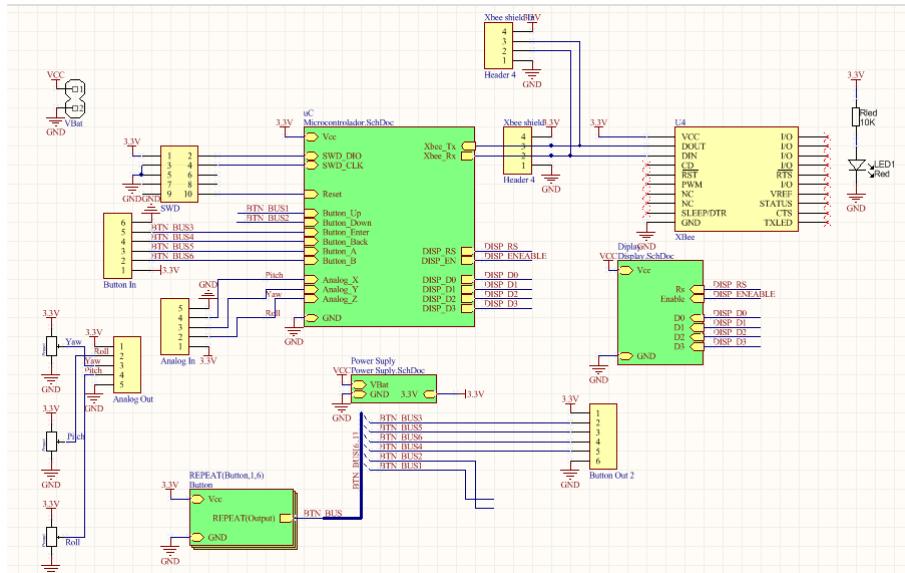


Figura VIII.2.11.: Esquemático de la placa principal.

A partir de los esquemáticos elaborados se realizó la placa de la Figura VIII.2.12. Esta placa posee la fuente de alimentación, el microcontrolador y dos de los botones. Una de las cosas a destacar es la elección de colocar la alimentación y la referencia en los bordes de la placa, en capas opuestas, con el cuidado de no cerrar un lazo con ninguno de los dos. La razón de esta decisión es para disminuir la cantidad de ruido electromagnético que pueda captar. Asimismo, se decidió que el plano de masa de arriba sea tierra, y el plano de masa de abajo sea alimentación, estos se comportan como un capacitor gigante, que permite mejorar el desacople.

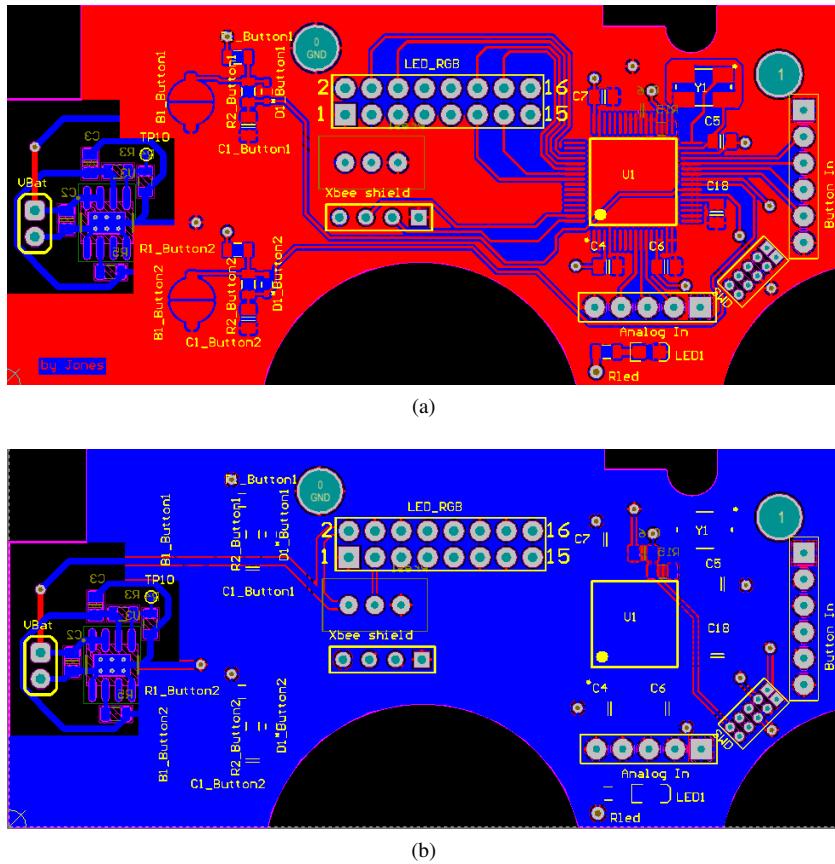


Figura VIII.2.12.: Diseño del PCB de la placa principal del control remoto: a) capa superior, b) capa inferior.

Se realizaron dos placas mas, una que contenía los cuatro botones restantes, y otra donde montar el modulo Xbee. Estas placas no requirieron consideraciones de diseño especiales, y pueden verse en las figuras VIII.2.13 y VIII.2.14 respectivamente.

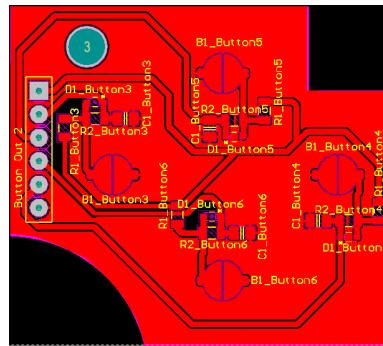


Figura VIII.2.13.: Diseño del PCB de la placa de dispositivos de entrada digitales del control remoto.

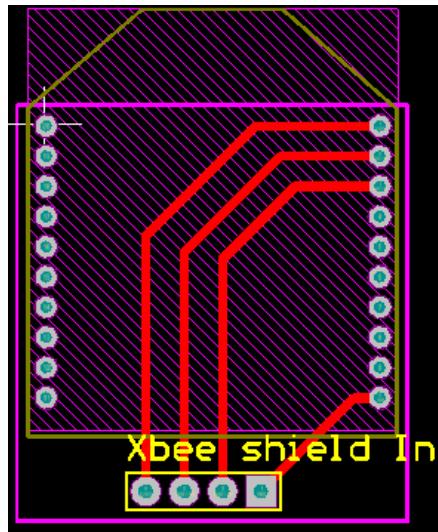


Figura VIII.2.14.: Diseño del PCB de la placa estilo Shield del modulo XBee del Control Remoto.

VIII.3 Diseño mecánico

Debido a que el *gimbal* es una estructura prefabricada, las decisiones de diseño mecánico se centran en la disposición de la placa de control y la batería de litio. La ubicación de este último componente es crítica debido a su peso, puesto que puede dificultar e incluso imposibilitar la calibración si se coloca en un lugar equivocado. Durante el período de diseño se elaboró un gabinete que estaba asegurado en la parte trasera del motor de *roll*, aunque este experimento fracasó. El peso combinado de la batería y el gabinete provocó que no fuera suficiente el desplazamiento del tubo vertical hacia adelante. En consecuencia, no se podía realizar la calibración mecánica del eje *yaw*.

A la luz de estos resultados se decidió colocar la batería fuera del lugar de acción de los motores, de manera que sólo se deba balancear el peso de la cámara. El único lugar que cumple este requisito es en el manubrio y, por conveniencia, se ubicó debajo del agarre central. Esto es así porque la forma de ranura del perfil de fibra de carbono tiene la medida justa para deslizar la batería, de manera que queda asegurada en posición.

En cuanto a la placa, los agujeros en los extremos tienen como propósito sujetarla a la estructura, que resulta sencillo porque tienen la misma separación que en las placas de fibra de carbono del *gimbal*. Para evitar el contacto directo entre los componentes de la capa inferior de la placa con la fibra de carbono, se utilizaron arandelas de goma de 3mm de espesor.

La flexibilidad en la colocación dio lugar a dos posibles opciones: en la parte trasera del motor de *roll* y en la parte inferior del motor de *yaw*. Ambas alternativas permiten que los cables de *roll*, de *yaw* y del sensor no se tensen en ningún momento, al estar en secciones que se mueven solidariamente con los motores. Sin embargo, la ubicación debajo del motor de *yaw* tiene la ventaja adicional de no tensar el cable de alimentación, puesto que se encuentra en el punto pivotante del agarre central.

VIII.3.1 Diseño mecánico del control remoto

Como se mencionó anteriormente, se modificó la carcasa de un control de Playstation 2 para usarla como control remoto.

VIII.4 Detalles de construcción y precauciones especiales de montaje

VIII.4.1 Control remoto

Las modificaciones que se hicieron al control de Playstation 2 fueron:

- La remoción de estructuras de soporte interno de los botones del lado izquierdo para poder colocar un módulo Xbee.
- El agregado de una caja trasera donde colocar la batería.
- El agregado de agujeros para pasar los cables de la batería.
- El pintado de la carcasa con aerosol negro mate.

Las placas de circuitos se diseñaron para encajar perfectamente en la carcasa del control remoto, de modo que queden sujetas por la presión de una serie de estructuras internas. El módulo Xbee está asegurado con un tornillo en uno de los antiguos soportes de uno vibradores del control remoto.

En las Figuras VIII.4.1, VIII.4.2, VIII.4.3, VIII.4.4 y VIII.4.5 puede verse la carcasa completada y con las placas de circuitos ya montados.



Figura VIII.4.1.: Perspectiva isométrica del control remoto modificado.



Figura VIII.4.2.: Vista de frente del control remoto modificado.



Figura VIII.4.3.: Vista trasera del control remoto modificado.

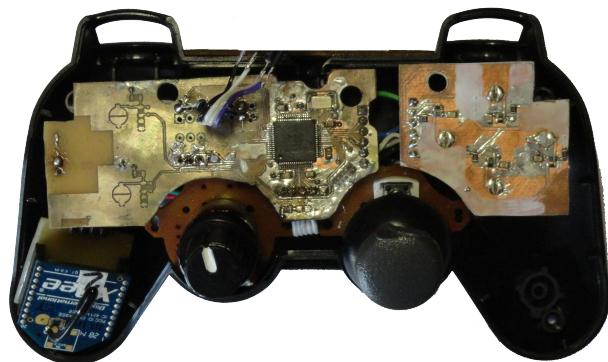


Figura VIII.4.4.: Vista frontal de las placas de circuitos dentro del control remoto modificado.

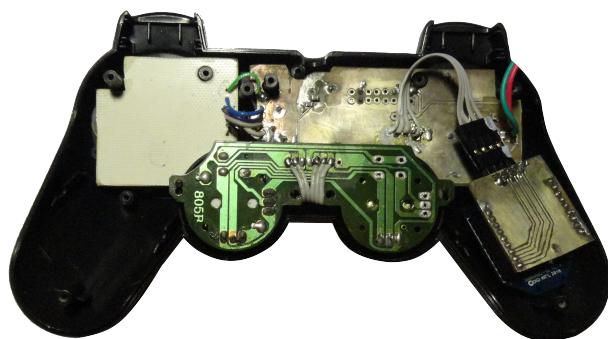


Figura VIII.4.5.: Vista trasera de las placas de circuitos dentro del control remoto modificado.

IX Validación del prototipo

Con la finalidad de verificar el correcto funcionamiento del prototipo, se diseñaron y llevaron a cabo una serie de mediciones y pruebas, que relevarían tanto el hardware como el software. A continuación se detallan los distintos protocolos empleados en cada caso.

IX.1 Validación de hardware

IX.1.1 Plan y protocolos especiales de medición

Para esta categoría se consideraron los siguientes módulos:

- Batería.
- Regulador de tensión *Buck*.
- Regulador de tensión lineal.
- *Drivers* de los motores.
- Motores.

Cabe destacar que ni el módulo Xbee ni el IMU se incluyeron en esta categoría debido a que su funcionamiento está más sujeto a cuestiones de software que de hardware; adicionalmente, la verificación del hardware es autónoma en cada uno de los módulos.

A continuación, se detallan los procedimientos elaborados para la verificación de cada uno de los módulos mencionados anteriormente. Es necesario aclarar que los procedimientos deben llevarse a cabo en el orden establecido, ya que cada módulo depende del correcto funcionamiento del anterior.

1. **Batería:** En primer lugar se desconecta la batería y se mide la tensión entre bornes, que debe estar entre los 11,5V y 12,5V. Luego, se conecta una carga conocida a la batería, compuesta por tres resistencias de alambre bobinado en serie; midiendo la tensión entre los extremos de la segunda resistencia, y verificando que esta sea diferente de cero, se comprueba que la batería está entregando corriente. Es necesario para la realización de esta verificación que la batería permanezca desconectada de la placa.
2. **Regulador de tensión *Buck*:** Si al encender el equipo el primer led de *status* se enciende, entonces queda claro que la fuente está entregando corriente; finalmente, se debe utilizar un voltímetro para verificar que la tensión entregada esté en el rango de 4,5V a 5,5V.
3. **Regulador de tensión lineal:** Si al encender el equipo el segundo led de *status* se enciende, entonces queda claro que la fuente está entregando corriente; finalmente, se debe utilizar un voltímetro para verificar que la tensión entregada esté en el rango de 3,0V a 3,6V.

4. **Drivers de los motores:** Para la verificación de este módulo se diseña un software de prueba que genere una señal PWM con un patrón conocido, y luego se observa, utilizando un osciloscopio, que el patrón se reproduzca a la salida de los *drivers*. Es necesario, para la realización de esta verificación, que los motores permanezcan desconectados de la placa.
5. **Motores:** Para verificar que los motores estén funcionando en forma correcta se diseña un software de prueba que genere tres señales PWM de *duty cycle* fijo calculadas para que el motor se coloque en una posición fija. Si esto ocurre, se debe forzar un movimiento en el rotor del motor, el cual debería fijarse a 120° de la primera posición; luego, se debe repetir el movimiento forzado y observar una tercera posición fija, a 120° de la segunda y 240° de la primera; finalmente, al forzar un nuevo movimiento, el rotor debería volver a la posición inicial.

En cuanto al control remoto, se consideraron los siguientes módulos:

- Batería.
- Regulador de tensión lineal.
- Dispositivos de entrada analógicos.
- Dispositivos de entrada digitales.

Cabe destacar que el módulo del LED de estado RGB no se incluyó en esta categoría debido a que su funcionamiento está más sujeto a cuestiones de software que de hardware; adicionalmente, la verificación del hardware es autónoma en cada uno de los módulos.

A continuación, se detallan los procedimientos elaborados para la verificación de cada uno de los módulos mencionados anteriormente. Es necesario aclarar que, en este caso, solo primeros dos procedimientos deben llevarse a cabo en el orden establecido, ya que el correcto funcionamiento del regulador lineal depende del correcto funcionamiento de la batería. En el caso de los otros dos módulos, debido a que los dispositivos de entrada analógicos y digitales se encuentran en placas separadas de la placa de control, estas pueden ser probadas independientemente.

1. **Batería:** En primer lugar se desconecta la batería y se mide la tensión entre bornes, que debe estar entre los 3,4V y 4,1V. Luego, se conecta una carga conocida a la batería, compuesta por tres resistencias de alambre bobinado en serie; midiendo la tensión entre los extremos de la segunda resistencia, y verificando que esta sea diferente de cero, se comprueba que la batería está entregando corriente. Es necesario para la realización de esta verificación que la batería permanezca desconectada de la placa.
2. **Regulador de tensión lineal:** Se debe utilizar un voltímetro para verificar que la tensión entregada esté en el rango de 3,0V a 3,6V.
3. **Dispositivos de entrada analógicos:** Se deben conectar las entradas de tensión y tierra a una fuente de tensión que entregue 3,3V, y utilizar un voltímetro para verificar que la tensión de salida de los potenciómetros varie entre de 0V a 3,3V de forma lineal en relación al movimiento de estos.
4. **Dispositivos de entrada digitales:** Se deben conectar las entradas de tensión y tierra a una fuente de tensión que entregue 3,3V, y utilizar un osciloscopio para verificar que entreguen 3,3V en estado inactivo y 0V en estado activo con un sobrepico menor a 1,1V en los cambios de estado.

IX.1.2 Mediciones

Batería

Luego de completar un ciclo de carga, se conectó la batería a una carga compuesta por tres resistencias de $3,3\Omega$, 10W. En estas condiciones, se utilizó un voltímetro para medir la tensión entre los bornes de la batería y entre los terminales de una de las resistencias; los resultados de estas mediciones fueron 11,83V y 3,75V respectivamente.

Regulador de tensión *Buck*

Luego de conectar la batería a la placa y oprimir el botón de encendido, se observó el encendido del primer led de *status*. Posteriormente, se empleó un voltímetro para medir la tensión entregada por el regulador; el resultado de esta medición fue 4,91V.

Regulador de tensión lineal

Luego de conectar la batería a la placa y oprimir el botón de encendido, se observó el encendido del segundo led de *status*. Posteriormente, se empleó un voltímetro para medir la tensión entregada por el regulador; el resultado de esta medición fue 3,22V.

Driver de los motores

Luego de desconectar los motores, se procedió a correr el programa, el cual generó una señal PWM que variaba su *duty cycle* desde 50% a 0% en forma periódica. Utilizando un osciloscopio se observó a la salida del *driver* la imagen correspondiente a la Figura IX.1.1 presentada a continuación.

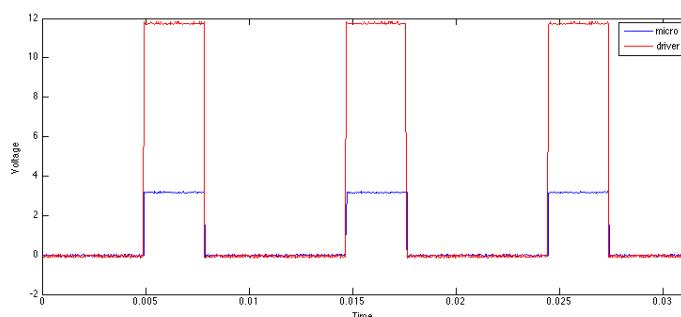


Figura IX.1.1.: Medición de prueba a la salida de los drivers

La señal PWM observada resultó idéntica a la generada por el programa de prueba, excepto que montada sobre 11,82V.

Motores

Luego de conectar los motores, se procedió a correr el programa, el cual generó un movimiento del motor hasta una posición en la cual se detuvo. Se forzó el movimiento del rotor, el cual se estableció a, aproximadamente, 120° de la posición inicial. Se forzó el rotor una segunda vez, observando nuevamente un desplazamiento de, aproximadamente, 120°. Un último movimiento forzado devolvió el rotor a la posición inicial.

Batería del control remoto

Luego de completar un ciclo de carga, se conectó la batería a una carga compuesta por tres resistencias de $3,3\Omega$, 10W. En estas condiciones, se utilizó un voltímetro para medir la tensión entre los bornes de la batería y entre los terminales de una de las resistencias; los resultados de estas mediciones fueron 4,02V y 1,34V respectivamente.

Regulador de tensión lineal del control remoto

Se empleó un voltímetro para medir la tensión entregada por el regulador; el resultado de esta medición fue 3,26V

Dispositivos de entrada analógicos

Se empleó un voltímetro para verificar que en los tres potenciómetros la tensión varíe linealmente con el movimiento de los mismos, los resultados pueden verse en la Figura IX.1.2.

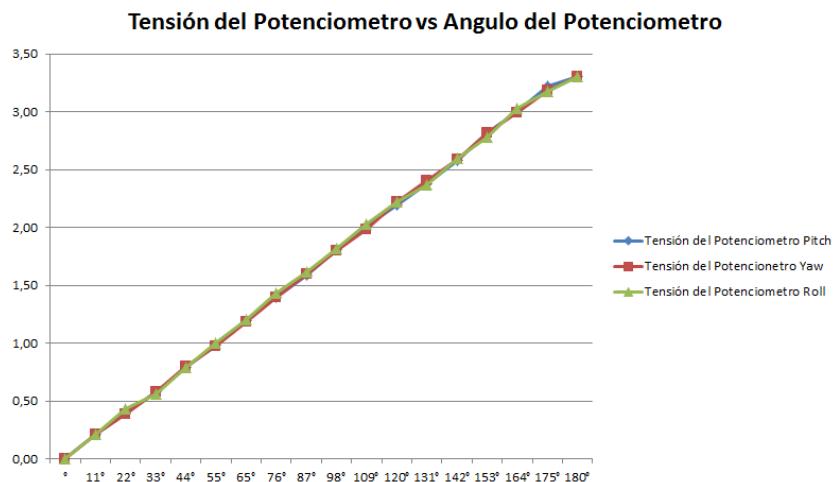


Figura IX.1.2.: Medición de prueba a la salida de los potenciómetros

Dispositivos de entrada digitales

Se midió la tensión en la salida del circuito antirebotes del pulsador mediante el uso del osciloscopio, los resultados pueden ser vistos en la Figura IX.1.3.

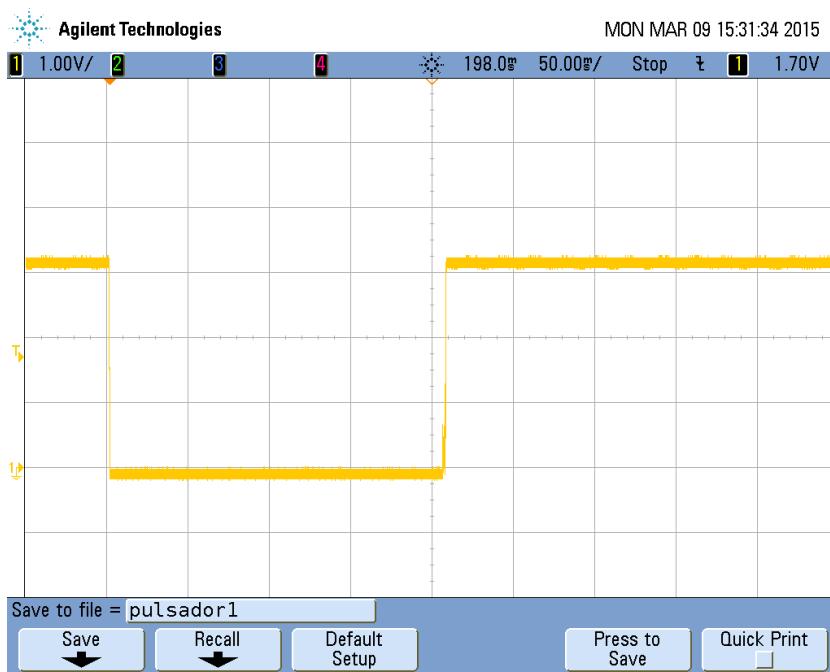


Figura IX.1.3.: Medición de prueba a la salida de los pulsadores

Puede verse que el sistema no tiene rebotes, ni sobrepicos apresiables.

IX.1.3 Evaluación

Batería

Los resultados indican que la primera medición se encuentra dentro del rango de aceptación, y que la segunda es diferente de cero, por lo que se concluye que la batería opera según las especificaciones .

Regulador de tensión Buck

Debido a que la tensión entregada por el regulador (4.91V) es compatible con la tensión provista por el USB ($5V \pm 0.25V$)

Regulador de tensión lineal

La medición de tensión (3.26V) se encuentra dentro del rango del 5% de aceptación estándar (3.465V - 3.135V) se determina que la fuente de alimentación funciona correctamente.

Driver de los motores

Dada la perfecta transmisión de la señal de control para cada uno de los *drivers*, se verifica el correcto funcionamiento de estos.

Motores

Dado que se observaron tres posiciones fijas en una vuelta de cada rotor para una misma señal de control, se verifica el correcto funcionamiento de los motores.

Batería del control remoto

Dado que la primera medición se encuentra dentro del rango de aceptación, y que la segunda es diferente de cero, se verifica el correcto funcionamiento de la batería del control remoto.

Regulador de tensión lineal del control remoto

Dado que la medición se encuentra dentro de su rango de aceptación, se verifica el correcto funcionamiento del regulador de tensión lineal del control remoto.

Dispositivos de entrada analógicos

Dado que la medición se encuentra dentro de su rango de aceptación, se verifica el correcto funcionamiento de los dispositivos de entrada analógicos.

Dispositivos de entrada digitales

Dado que la medición se encuentra dentro de su rango de aceptación, se verifica el correcto funcionamiento de los dispositivos de entrada digitales.

IX.1.4 Resultados

La realización de los procesos de prueba, y la posterior evaluación de los resultados, permite determinar que los módulos de hardware del prototipo están funcionando correctamente.

IX.2 Validación de software

IX.2.1 Plan y protocolos especiales de medición

Para esta categoría se consideraron los siguientes módulos:

- Control de motores.
- IMU (procesado digital).
- GUI.
- Control de balanceo.
- Xbee (comunicación inalámbrica).

Es necesario explicitar la diferencia entre control de motores y de balanceo; el primero de los dos es un conjunto de subrutinas que permiten colocar el rotor del motor en una posición angular específica y realizando el movimiento a una velocidad determinada, mientras que el segundo es el responsable de determinar estas posiciones y velocidades en tiempo real con el objetivo de mantener balanceada la cámara. De esto surge que el control de balanceo puede ser evaluado únicamente después de haber validado el control de posición y el IMU.

A continuación, se detallan los procedimientos elaborados para la verificación de cada uno de los módulos mencionados anteriormente. A diferencia del caso del hardware, no es necesario llevar a cabo los procedimientos en el orden establecido, ya que existe solo una única dependencia que ha sido aclarada en el párrafo anterior.

1. **Control de motores:** Con los motores conectados, se utiliza un software de prueba que hace girar al motor ininterrumpidamente; variando los parámetros de este programa se puede evaluar que el control de velocidad esté funcionando correctamente. Luego, con los motores conectados aún, se utiliza un segundo software que posiciona el rotor del motor según un *slider*; variando este parámetro, se puede evaluar que el control de posición esté funcionando correctamente.
2. **IMU:** Para evaluar el sensor inercial se emplea un graficador de uso libre adquirido en internet; este permitirá evaluar en forma visual que las mediciones realizadas por el sensor y transmitidas por el protocolo de comunicación estén siendo procesadas en forma correcta. Este procedimiento puede ser de índole visual debido a que el procesado digital de la señal apunta a la remoción de ruido y *drift*, factores que pueden ser evaluados a través de un procesador con gran facilidad.
3. **GUI:** Un proceso de *debugging* en tiempo real permite verificar que la comunicación entre la GUI y el microprocesador esté funcionando en forma correcta.
4. **Control de balanceo:** Con la placa y la batería montados en el *gimbal* y los motores conectados, se realiza una calibración física y luego una calibración digital, según lo establecido en los manuales de procedimiento. Esto debería resultar en un control suave y preciso de la posición de la cámara. Si esto no fuera posible, y teniendo en cuenta que ya se han realizado validaciones sobre el control de posición y el IMU, deben realizarse modificaciones sobre la implementación del control de balanceo.
5. **Xbee:** Para verificar que el módulo Xbee esté transmitiendo y recibiendo en forma correcta se utiliza el XCTU, un software gratuito de Digi que permite enviar y recibir comandos por una terminal. Cabe destacar que para realizar esta prueba se debe contar con dos conversores USB/UART. La validación se considera exitosa si se puede establecer una comunicación bidireccional.

IX.2.2 Mediciones

Control de motores

Ambos programas de prueba generaron los resultados esperados luego de su ejecución.

IMU

Las lecturas observadas en el graficador presentaron un nivel de ruido lo suficientemente pequeño (menor a 3/250deg para el giroscopio y 0.02g para el acelerómetro) como para interpretar las mediciones con una precisión de hasta 0.1°. Asimismo, no se apreció *drift* alguno.

GUI

Las imágenes presentadas en el graficador resultaron visualmente similares a las del software descargado; a su vez, la ejecución de comandos en la GUI produjo los cambios correspondientes en el programa interno del microprocesador, visibles gracias al led de estado.

Control de balanceo

Con el *gimbal* cargado y calibrado físicamente, se procedió a modificar los parámetros del control PID de posición. Utilizando la GUI como ayuda visual para interpretar los efectos de estos cambios, se logró una calibración digital del *gimbal* satisfactoria, así como un efectivo balanceo de la cámara.

Xbee

Ambos programas de prueba generaron los resultados esperados luego de su ejecución.

IX.2.3 Evaluación

Control de motores

Dado que el comportamiento de los motores se corresponde con lo esperado, se verifica el correcto funcionamiento del control de motores.

IMU

Dado que las señales visualizadas tienen las características requeridas, se verifica el correcto funcionamiento del módulo IMU.

GUI

Dado que el graficador desarrollado se corresponde con el descargado, y que los comandos de la GUI fueron interpretados correctamente por el procesador, se verifica el correcto funcionamiento del módulo GUI.

Control de balanceo

Dado que es posible balancear una cámara dentro de los parámetros establecidos, se verifica el correcto funcionamiento del control de balanceo.

Xbee

Dado el éxito de la comunicación inalámbrica, se verifica el correcto funcionamiento del módulo Xbee.

IX.2.4 Resultados

La realización de los procesos de prueba, y la posterior evaluación de los resultados, permiten determinar que los módulos de software del prototipo están funcionando correctamente.

X Estudios de confiabilidad de hardware y de software

X.1 Hardware

Dado que los módulos internamente no poseen redundancias, ya que no fue considerado necesario, se consideran en forma interna como un sistema completamente en serie. Por lo tanto una falla en cualquiera de sus componentes producirá un fallo del módulo.

A continuación se presenta un estudio de confiabilidad de cada uno de los módulos según los componentes que lo conforman. Los λ se consideran en *fallas/10⁶ horas*.

Módulo microprocesador

Componentes	Cantidad	λ
R: 150R	1	0.09774512
R: 4k7	2	0.02186404
R: 10k	2	0.02186404
R: 39k	1	0.02186404
C: 100n	5	0.104729664
C: 1u	1	0.1334619
ABM3B	1	0.13212792
LED: R	1	0.005888
MK20DX128VLH5	1	1.56387613

Cuadro X.1.: *Estudio de confiabilidad del módulo microprocesador*

Como se indicó previamente, los elementos se consideran conectados en serie, por lo que el λ correspondiente al módulo se calcula como la sumatoria de los de cada componente; de esta forma, se obtiene un valor para λ_M de 2.56606759, lo cual implica un MTBF de aproximadamente 389701.35 horas.

Módulo USB

Componentes	Cantidad	λ
R: 10k	1	0.02186404
C: 100n	5	0.104729664
CP2104-F03-GM	1	1.797307025

Cuadro X.2.: *Estudio de confiabilidad del módulo USB*

Como se indicó previamente, los elementos se consideran conectados en serie, por lo que el λ correspondiente al módulo se calcula como la sumatoria de los de cada componente; de esta forma, se obtiene un valor para λ_U de 2.342819385, lo cual implica un MTBF de aproximadamente 426836.15 horas.

Módulo driver

Componentes	Cantidad	λ
R: 10k	2	0.02186404
C:220n	2	0.112824322
C: 100n	1	0.104729664
C: 10n	1	0.085424941
C:22u	1	0.428946633
C: 1u	1	0.1334619
BAV99	1	0.0114048
L6234PD	1	0.171331302

Cuadro X.3.: Estudio de confiabilidad del driver

Como se indicó previamente, los elementos se consideran conectados en serie, por lo que el λ correspondiente al módulo se calcula como la sumatoria de los de cada componente; de esta forma, se obtiene un valor para λ_D de 1.204675964, lo cual implica un MTBF de aproximadamente 830098.7 horas.

Módulo fuente

Componentes	Cantidad	λ
R: 2k74	1	0.02186404
R: 10k	3	0.02186404
R: 15k	1	0.02186404
R: 39k	1	0.02186404
C: 1.2n	1	0.07058477
C: 1u	3	0.1334619
C:10u	1	0.159068685
C: 22u	1	0.428946633
L: 100u	1	0.000594
AOZ1013	1	0.071908471
XPR6272	1	0.13212792

Cuadro X.4.: Estudio de confiabilidad de la fuente de alimentación

Como se indicó previamente, los elementos se consideran conectados en serie, por lo que el λ correspondiente al módulo se calcula como la sumatoria de los de cada componente; de esta forma, se obtiene un valor para λ_F de 1.394800419, lo cual implica un MTBF de aproximadamente 716948.45 horas.

Módulo XBee

Componentes	Cantidad	λ
Xbee Transceiver	1	1.797307025

Cuadro X.5.: Estudio de confiabilidad del módulo XBee

Como se indicó previamente, los elementos se consideran conectados en serie, por lo que el λ correspondiente al módulo se calcula como la sumatoria de los de cada componente; de esta forma, se obtiene un valor para λ_X de 1.797307025, lo cual implica un MTBF de aproximadamente 5563879.66 horas.

Placa completa

Para el estudio de la placa completa se considera nuevamente que todos los módulos se encuentran conectados en serie debido a que, si bien existen tres módulos driver independientes entre sí, la falla de cualquiera de ellos implica el mal funcionamiento de la placa. Así, se calcula el λ_{total} según:

$$\lambda_{total} = \lambda_F + \lambda_U + \lambda_M + 3 \cdot \lambda_D + \lambda_X,$$

lo que da un resultado de 11.715022311, que implica un MTBF de aproximadamente 85360.5 horas.

X.2 Software

El estudio de confiabilidad para software se realizó con una modalidad de predicción; esta se basa en datos históricos de proyectos similares buscando estimar una tasa de fallos que aplique al software del proyecto propio.

Para esto se utilizó el modelo de Musa, cuya ecuación fundamental es la X.2.1 presentada a continuación.

$$\lambda = k \cdot p \cdot \omega_0 \quad (\text{X.2.1})$$

$$p = \frac{r \cdot ER}{SLOC} \quad (\text{X.2.2})$$

En esta fórmula, k es una constante de valor $4,2 \cdot 10^{-7} \cdot 360^{-1}$, p estima el número de ejecuciones por unidad de tiempo, y ω_0 estima las fallas en el programa. Para determinar p se requiere conocer la totalidad de líneas de código en el proyecto, que se estimó estaría al rededor de las 3000, así como la velocidad de la ejecución de instrucciones r ; ER es una constante de valor 2.5 para códigos realizados en C.

Así, los valores adoptados para cada variable pueden observarse en el siguiente cuadro:

Constante	Valor
k	$4,2 \cdot 10^{-7} \cdot 360^{-1}$
ω_0	0,01 fallas/SLOC
r	14M/s
ER	2.5
$SLOC$	3000
p	11666.67
λ	$13,61 \cdot 10^{-8}$

Cuadro X.6.: Constantes del estudio de confiabilidad de software

De esta forma, el MTBF para el software es de aproximadamente 2040 horas.

XI Conclusiones

XI.1 Conclusiones excelencias y objetivos alcanzados

A lo largo de este proyecto se logró asentar y expandir los conocimientos adquiridos durante la carrera, fundamentalmente en los temas de diseño de sistemas mecánicos, de PCBs, de sistemas de control digitales, y en programación de software, tanto en C como en Java.

Fue posible llevar a cabo las etapas de diseño, construcción y validación de un equipo de estabilización de tres ejes con control remoto y de su software de calibración. El prototipo también cumplió con los requerimientos y especificaciones surgidos de la entrevista con los potenciales clientes, y del estudio del mercado actual.

También se consiguió que el proyecto fuera económicamente viable, tal como se puede ver en el análisis económico realizado.

XI.2 Fallos. Recomendaciones para diseños futuros

Se observó ruido eléctrico en las señales de I2C del sensor, producto de la commutación de alta frecuencia del PWM que controla los BLDC. Para resolver este inconveniente se proponen las siguientes mejoras:

- Mejorar el desacople del canal de I2C respecto a la masa del circuito utilizando capacitores de desacople.
- Mejorar la aislación de la señal de I2C utilizando cable coaxil, que presenta mejor inmunidad al ruido que el plano.
- Disminuir la inducción de ruido eléctrico de los motores en el canal de comunicación bobinando los cables de PWM a núcleos de ferrites.

XII Anexos

XII.1 Contrato modelo de locación de equipo

NUESTRA EMPRESA, en adelante el LOCADOR, y, el LOCATARIO, cuya individualización surge al dorso, convienen en celebrar el siguiente contrato de locación de equipos de filmación, de acuerdo con las normas establecidas por el artículo 1499 del Código Civil y que se regirá por las siguientes cláusulas y condiciones:

PRIMERO:

el LOCATARIO declara recibir en locación los objetos que se detallan al dorso de este contrato y que forman parte del mismo, de plena conformidad y en perfecto estado de funcionamiento, por haberlos probado en el acto de su recepción obligándose a su restitución en el plazo establecido entre las partes y en las mismas condiciones de funcionamiento.

SEGUNDO:

el plazo de la locación como así también su destino es establecido entre las partes y su precio es el estipulado en las tarifas vigentes que el LOCATARIO declara conocer.

TERCERO:

los bienes locados, objetos de este contrato, deberán ser devueltos antes de las ocho (8) horas del día siguiente del vencimiento del plazo, en el domicilio del LOCADOR, sea cual fuere el día y horario de finalización.

CUARTO:

en caso de incumplimiento a lo establecido en el artículo precedente, el LOCATARIO pagará al LOCADOR por cada día de demora, incluído los feriados, el precio del alquiler pactado más un recargo del cincuenta por ciento (50 %) de dicha suma, en concepto de cláusula penal, y que se operará de pleno derecho y por el mero vencimiento del plazo pactado, sin necesidad de interpellación alguna. Si por razones de producción o climáticas, el LOCATARIO solicita la cancelación del arriendo con una antelación inferior a las veinticuatro horas de la fecha estipulada de locación, deberá igualmente abonar al locador el cincuenta por ciento (50 %) calculado sobre la tarifa diaria. Si la cancelación se comunica luego de haber retirado los bienes locados, el LOCATARIO deberá igualmente abonar al LOCADOR el cien por ciento (100 %) de la tarifa diaria.

QUINTO:

el LOCATARIO ha probado y reconoce el perfecto funcionamiento de los bienes locados, y en consecuencia, el LOCADOR no asume responsabilidad alguna sobre el resultado de los trabajos que se efectúen con dichos elementos.

SEXTO:

el LOCATARIO se obliga y a su costa a custodiar y mantener en perfectas condiciones los elementos objetos de la locación. En caso de rotura o deterioro de éstos durante el lapso del contrato y hasta su entrega definitiva según lo establecido en el artículo cuarto de este contrato, el LOCATARIO se obliga y a su costa al reemplazo y/o reparación de las piezas y/o mecanismos y/o partes afectadas que impidan el uso normal de los equipos en cuestión por otras legítimas de las marcas que correspondan y las mencionadas reparaciones se efectuarán por los términos que el LOCADOR indique en la emergencia.

SÉPTIMO:

para el supuesto caso del artículo precedente, jugarán las condiciones previstas en el artículo cuarto respecto al pago de la locación por cada día de demora y sus recargos correspondientes, hasta el día en que los elementos estén perfectamente reparados y entregados de conformidad al LOCADOR. Esto sin perjuicio del pago de las reparaciones a costas del LOCATARIO.

OCTAVO:

el LOCATARIO se hace responsable por los daños y perjuicios que pudieran sufrir los bienes dados en locación, ya sea por caso fortuito, fuerza mayor, culpa, hechos terceros, etc.

NOVENO:

el LOCATARIO para el supuesto caso de lo previsto en el artículo precedente, o que se produzca la pérdida, destrucción o roturas de los bienes durante el lapso de la duración del contrato que impidan en el futuro el uso normal de los equipos locados, se obliga a reponer cualquiera o todos esos elementos que hubieran sufrido el daño o pérdida, o a abonar al LOCADOR el valor de los mismos a elección de éste, al precio establecido por los comercios de plaza o sus importadores, con las pertinentes cargas impositivas y aduaneras y fletes que permita al LOCADOR proceder al reemplazo de dichos elementos. Queda establecido para el supuesto caso en que por aplicación de leyes vigentes en el futuro se impida el reemplazo de estos elementos por estar vedada su importación, el LOCATARIO se obliga al pago de los bienes según el valor dado a los mismos con un recargo del cien por ciento (100 %) en concepto de indemnización por los daños y perjuicios que le ocasiona al LOCADOR la pérdida de dichos elementos. Para el caso que los equipos hayan sido discontinuados, la sustitución se realizará por el modelo siguiente. Una vez devuelto el equipo, el LOCADOR comunicará al LOCATARIO dentro de las 24 horas siguientes todos los deterioros o desperfectos que constate en su funcionamiento, y las partes acuerdan que el LOCADOR queda facultado para disponer la reparación o sustitución de aquellos ítems que se encuentren dañados.

DÉCIMO:

el LOCATARIO se obliga a no trasladar los bienes locados a una distancia mayor de 100 Km. de la Capital Federal, a no ser que el LOCADOR preste su conformidad para ese traslado. Para el supuesto caso de incumplimiento de lo establecido precedentemente, el contrato quedará rescindido de pleno derecho sin necesidad de interpellación judicial o extrajudicial. Queda facultado el LOCADOR para retirar los elementos

locados, por sí o por interpósta persona en el lugar que se encuentren y en el momento que se estime conveniente. En caso que el LOCATARIO, o los eventuales tenedores de los elementos opusieran resistencia a la entrega de los mismos, el LOCADOR podrá solicitar sin más trámite el secuestro judicial de los mismos, siendo por cuenta del LOCATARIO los gastos y los costos que esta acción implique. Queda expresamente establecido que el LOCADOR no asume responsabilidad alguna por los perjuicios que pudiera significar al LOCATARIO el retiro y/o secuestro judicial de los elementos, que podrá solicitarse en este último caso sin necesidad de presentar caución o garantía alguna.

DÉCIMO PRIMERO:

el LOCADOR en todo momento tendrá derecho a inspeccionar y verificar el estado de los bienes locados cualquiera sea el lugar donde se encuentren, y para el supuesto caso en que se comprobare que el uso y destino, como así también la guarda y custodia de dichos bienes hace peligrosa su integridad y buen funcionamiento futuro, el LOCADOR queda plenamente facultado para rescindir el contrato, quedando obligado el LOCATARIO a la restitución de los bienes dentro del plazo y lugar que el LOCADOR indique, sin perjuicio de las facultades conferidas en los artículos precedentes y de lo establecido en el artículo cuarto del presente contrato.

DÉCIMO SEGUNDO:

queda absolutamente prohibido al LOCATARIO subalquilar y/o prestar y/o dar en comodato o de cualquier manera desprenderse de la tenencia de los objetos dados en locación. Estos además deberán ser operados por personal técnico idóneo a criterio del LOCADOR, y en caso contrario se procederá de acuerdo a lo establecido en los artículos décimo y decimoprimero.

DÉCIMO TERCERO:

para el supuesto caso que se den las condiciones previstas en los artículos cuarto, sexto, octavo, noveno, décimo y décimo primero (4°, 6°, 8°, 9°, 10° y 11°), que de cualquier manera se obstaculizara el derecho de inspección, o que se den las condiciones previstas por las cuales se faculta al LOCADOR a rescindir el presente contrato, esta rescisión se opera de pleno derecho y sin necesidad de interpellación judicial o extrajudicial alguna, obligándose el LOCATARIO a la entrega inmediata de los objetos dados en locación y sin obligación por parte del LOCADOR de restitución de suma alguna para el supuesto caso en que los pagos se hubieran efectuado por adelantado, quedando a su vez obligado el LOCATARIO a cumplimentar el pago del arrendamiento de los bienes locados hasta el vencimiento del plazo previsto, teniendo la presente condición carácter de cláusula penal. Por otra parte la falta de entrega de los bienes dentro de los términos establecidos en el presente contrato, determinará la obligación del LOCATARIO al pago del alquiler pactado, más una suma igual, en carácter de cláusula penal.

DÉCIMO CUARTO:

el material se entrega en arriendo sin cobertura de seguro alguna y es obligación del LOCATARIO contratar pólizas de seguro previniendo cualquiera de los riesgos que asume por el presente contrato (pérdida total o parcial del equipo, hurto, desperfectos o roturas, confiscación, etc.). La póliza deberá realizarse tanto para el traslado como para la utilización de los bienes locados, y deberán cederse los derechos de la misma al LOCADOR, a quien se entregará una copia con antelación a que los bienes locados sean retirados de las instalaciones del mismo. Para el caso que la cobertura no sea total, el LOCATARIO deberá en primer instancia poner en conocimiento de esta determinación al LOCADOR y con posterioridad responder frente

a éste por la reparación o reposición de los bienes y/o gastos insuficientemente cubiertos. El LOCADOR proporcionará al LOCATARIO toda la información necesaria a los efectos de realizar estas gestiones.

DÉCIMO QUINTO:

el precio de la locación se pagará, para el supuesto caso que exista cuenta corriente, a los treinta (30) días, sin perjuicio de los plazos y recargos por mora, precedentemente previstos.

DÉCIMO SEXTO:

para el supuesto caso que el LOCATARIO no de cumplimiento al pago de las sumas pactadas por el precio de la locación, como así también las cláusulas penales y recargos establecidos, el LOCADOR queda facultado a cobrar las sumas adeudadas por medio de la vía ejecutiva y de acuerdo a las normas establecidas por los arts. 523 inc. 2 y 525 del Código de procedimientos de la Capital Federal. En caso de ejecución se podrá tratar embargo sobre las películas y/o trabajos y/o negativos efectuados con los elementos locados, como así también en las taquillas de las salas de exhibición y/o cualquier otro bien que sea de la propiedad del LOCATARIO.

DÉCIMO SEPTIMO:

las partes se someten a los tribunales ordinarios de la Capital Federal, renunciando a cualquier otro fuero o jurisdicción en prueba de conformidad se firman dos (2) ejemplares de un mismo tenor y a un solo efecto, en lugar y fecha indicada al frente del presente contrato.

XII.2 Manual de Mantenimiento

Este manual tiene la función de brindar instrucciones explícitas y claras para la calibración y mantenimiento del STATHEROS.

Los procedimientos de calibración se dividen en dos partes: la **calibración física**, es decir el posicionamiento de la cámara en el centro de gravedad del sistema, y la **calibración digital**, la cual consiste en la asignación de variables P, I, y D para el algoritmo de control de la GUI. La correcta ejecución de los pasos descriptos en este documento es imperativa para lograr el funcionamiento óptimo del STATHEROS.

El procedimiento de mantenimiento desarrollado en este manual, debe seguirse a la hora de recambiar un motor que ha dejado de funcionar.

XII.2.1 STATHEROS

El STATHEROS es un producto de estabilización de cámaras de filmación en tres ejes de calidad profesional. Basado en una estructura tipo *gimbal* y operando motores BLDC, se logra estabilizar tomas en movimiento con una precisión de hasta 1°.

Especificaciones

- Peso: 2kg.
- Dimensiones:
 - Alto: 80cm.
 - Ancho: 60cm.
 - Largo: 40cm.

- Carga máxima: *5kg.*
- Duración de la batería: *3hs.*
- Temperatura de operación: de -10°C a 40°C.
- Alcance de la comunicación inalámbrica: *50m.*

XII.2.2 Recomendaciones de Seguridad

Para garantizar un uso seguro del STATHEROS se deben tener en cuenta las siguientes consideraciones:

- No exceder las condiciones de operación establecidas en las especificaciones.
- Al movilizar el STATHEROS, incluso si este no estuviera funcionando, procure sujetarlo siempre de los manubrios.
- Minimizar el número de cables colgando; siempre que se pueda (es decir, que no haya riesgo de desconexión), precintar o encintar los cables a la estructura.
- Asegurar la conexión de los cables de alimentación de placa de control y motores, a fin de prevenir problemas eléctricos.
- De trabajar en un ambiente que involucre agua o partículas de polvo en exceso, asegurar el correcto sellado de la carcasa protectora para la placa de control.
- Asegurar, previo a la utilización, la libre rotación de los motores; aceitar los rodamientos con WD-40 en caso de que el movimiento se vea dificultado.
- Al ajustar las los tornillos en el armado de la estructura, o en el procedimiento de calibración física, es recomendable no ajustar demasiado los mismos, sólo lo suficiente como para que las piezas no “bailen”.

XII.2.3 Calibración Física

Los siguientes procedimientos deben realizarse para calibrar al STATHEROS desde un punto de vista físico; el objetivo de esta parte del manual es lograr que los centros de masa de cada uno de los sub-sistemas físicos de la estructura se encuentren sobre el eje del motor que los acciona. Como resultado de esta calibración, cada motor tendrá que entregar un torque mínimo para poder mover a su sub-sistema.

Materiales Necesarios

- STATHEROS.
- Cámara.
- Estructura de calibración.
- Juego de llaves Allen.

Preparación del STATHEROS

Los primeros pasos consisten en la correcta preparación del STATHEROS, de forma tal que este quede en condiciones de ser calibrado.

1. Posicione el STATHEROS sobre la estructura de calibración; esta debe ser tal que permita al STATHEROS moverse libremente en todas direcciones al menos 30°. Asegurarse de que el manubrio quede lo más horizontal posible; si los agarres del manubrio están colocados, retírelos para así maximizar el rango de movimiento del equipo.
2. Monte la cámara sobre la bandeja del STATHEROS y atornílelo usando uno de los dos tornillos provistos; no es extraño, ni debería preocuparle, que el STATHEROS se incline una vez colocada la carga.
3. Equipe la cámara con todos los accesorios que se vayan a utilizar en la filmación y remueva la tapa protectora de la lente; la cámara debería quedar ya lista para filmar. Haga lo mismo con el STATHEROS (batería, placa, cables, etc).
4. Posíñese en frente del STATHEROS, es decir, en forma tal que la cámara esté mirando directamente hacia usted. Todas las referencias a posición, orientaciones o rotaciones son hechas en alusión a este punto de referencia.

Su equipo está ahora en condiciones de ser calibrado físicamente.

Pitch

El primer eje a calibrar es el de *pitch*.

Posicionamiento vertical

1. Incline la bandeja unos 90°, de forma que la cámara quede mirando directamente hacia arriba, o hacia abajo.
2. Suelte la bandeja y observe hacia donde se inclina la cámara.
3. Si esta queda mirando hacia atrás (es decir que lo mira a usted), entonces la bandeja está muy abajo, si, por otro lado, esta queda mirando hacia adelante (o sea, de cabeza), entonces la bandeja está muy arriba (as figuras XII.2.1a y XII.2.1b presentadas a continuación proveen una ayuda visual para este paso). Si en vez de esto la cámara queda mirando hacia abajo o arriba, vaya al Posicionamiento horizontal.



(a) Bandeja baja



(b) Bandeja alta

Figura XII.2.1.: Referencia para el posicionamiento vertical de la bandeja

4. Usando las tuercas mariposa a los costados de la bandeja, suba o baje la bandeja según sea necesario. Puede utilizar la graduación en las vías para asegurarse de tener el mismo nivel en ambos lados de la bandeja. Vuelva a ajustar la bandeja.
5. Repita los pasos 1 a 4 hasta que la cámara no se incline siempre de la misma manera.

Posicionamiento horizontal

1. Incline la bandeja unos 45° hacia arriba, de forma que la cámara quede mirando hacia arriba.
2. Suelte la bandeja y observe hacia donde se inclina la cámara.
3. Si esta queda mirando hacia arriba, entonces la bandeja está *tail-heavy*, si, por otro lado, esta queda mirando hacia abajo, entonces la bandeja está *nose-heavy* (las figuras XII.2.2a y XII.2.2b presentadas a continuación proveen una ayuda visual para este paso). Si en vez de esto la cámara queda mirando hacia adelante o atrás, vuelva al Posicionamiento vertical.



(a) Nose-heavy: requiere desplazamiento hacia atrás



(b) Tail-heavy: requiere desplazamiento hacia adelante

Figura XII.2.2.: Referencia para el posicionamiento horizontal de la bandeja

4. Si la bandeja está *nose-heavy*, entonces use el ajuste *quick-release plate* para deslizar la cámara hacia atrás; si, en cambio, la bandeja está *tail-heavy*, entonces deslice la cámara hacia adelante. Una herramienta de ajuste fino se provee con STATHEROS para realizar los últimos ajustes de esta calibración. Al encontrar una posición que se deseé mantener, vuelva a ajustar la cámara a la bandeja.

5. Repita los pasos 1 a 4 hasta que la cámara permanezca inmóvil una vez inclinada, o se desplace un ángulo insignificante de la posición en la cual se la ubica. Si encuentra que ninguna posición de la *quick-release plate* logra un buen equilibrio en la bandeja, entonces retire la cámara y móntela nuevamente en una ubicación que resulte más favorable; luego, comience nuevamente con el paso 1 de esta lista.

El eje de *pitch* ha sido exitosamente calibrado.

Roll

Sigue ahora el eje *roll*. Para este eje se pueden realizar ajustes finos o bruscos; se recomienda, en general, comenzar por estos primeros, y sólo realizar ajustes bruscos en casos donde sea absolutamente necesario.

Ajustes finos

1. Asegúrese de que la pieza de aluminio sujetada con tres tornillos al motor de *roll* esté centrada, tal como se puede observar en la Figura XII.2.3 presentada a continuación. De no ser así, utilice una llave Allen para aflojar los tornillos, luego deslice la pieza a la posición correcta y finalmente asegúrela en la posición correcta.



Figura XII.2.3.: Pieza centrada

2. Observe la inclinación de la cámara en el eje horizontal; si esta no se encuentra perfectamente horizontal, utilice una llave Allen para aflojar los tornillos que sujetan la *quick-release plate* a la bandeja de *roll*.
3. Si la cámara se inclina en forma horaria, desplace *quick-release plate* y cámara hacia la izquierda; si, por otro lado, la cámara se inclina en forma anti-horaria, deslice *quick-release plate* y cámara hacia la derecha. Puede observar las figuras XII.2.4a y XII.2.4b presentadas a continuación como referencia para ejecutar este paso en forma correcta. Si el sistema se desliza muy libremente, ajuste un poco los tornillos, pero no al punto de dejar inmóvil al sistema.



Figura XII.2.4.: Referencia para ajustes finos en roll

4. Repita el paso 3 hasta que la cámara permanezca estable en cualquier inclinación a la que se la someta; luego de haber calibrado exitosamente la cámara ajuste los tornillos debajo de la *quick-release plate*. Si encuentra que ninguna posición de la *quick-release plate* logra un buen equilibrio en la bandeja, entonces ajuste los tornillos debajo de la *quick-release plate* y proceda a la sección Ajustes bruscos.

Ajustes gruesos

1. Utilice una llave Allen para aflojar los tres tornillos que sujetan la pieza de aluminio. Intente no removerlos completamente de la estructura.
2. Si la cámara se inclina en forma horaria, desplace la pieza de aluminio hacia la izquierda; si la cámara se inclina en forma anti-horaria, deslice la pieza hacia la derecha. Si el sistema se desliza muy libremente, ajuste un poco los tornillos, pero no al punto de dejar inmóvil al sistema.
3. Repita el paso 3 hasta que la cámara permanezca estable en cualquier inclinación a la que se la someta; luego de haber calibrado exitosamente la cámara ajuste los tornillos.

El eje *roll* ha sido exitosamente calibrado.

Yaw

Para finalizar la calibración física, se debe calibrar el eje *yaw*.

Alineación en profundidad

1. Alinee la cámara de forma tal que esta quede mirando hacia un costado tal y como se muestra en la Figura XII.2.5 presentada a continuación.

Figura XII.2.5.: *Cámara alineada*

2. Incline el STATHEROS hacia arriba unos 15° utilizando el manubrio central y observe la inclinación de la estructura.
3. Si la estructura se inclina de forma tal que la cámara queda mirando hacia adelante, entonces el eje de yaw está muy adelante; si la estructura se inclina de forma tal que la cámara queda mirando hacia atrás, entonces el eje de yaw está muy atrás. Puede observar las figuras XII.2.6a y XII.2.6b presentadas a continuación como referencia para ejecutar este paso en forma correcta. Si la cámara permanece mirando hacia el costado, entonces no hay necesidad de corregir la posición del eje.
4. Para corregir la posición del eje, utilice una llave Allen para aflojar los tornillos que sujetan al eje de yaw al manubrio, y desplácelo hacia adelante o hacia atrás según corresponda.

(a) *Requiere desplazar hacia atrás*(b) *Requiere desplazar hacia adelante*Figura XII.2.6.: *Referencia para alineación en profundidad de yaw*

5. Repita los pasos 1 a 4 hasta que la cámara permanezca mirando hacia un costado luego de inclinarse la estructura, pero no ajuste aún los tornillos del eje.

Alineación rotacional

1. Alinee la cámara de forma tal que esta quede mirando hacia atrás (o sea, hacia usted).
2. Incline el STATHEROS hacia arriba unos 15° usando el manubrio central y observe la inclinación de la estructura.
3. Si la estructura se inclina de forma tal que la cámara queda mirando hacia la derecha, entonces el eje de yaw está inclinado en forma anti-horaria; si, por otro lado, la estructura se inclina de forma tal que la cámara queda mirando hacia la izquierda, entonces el eje de yaw está inclinado en forma horaria. Si la cámara permanece mirando hacia atrás no hay necesidad de corregir la inclinación del eje.
4. Para corregir la inclinación del eje, utilice una llave Allen para aflojar los tornillos que sujetan al eje de yaw al manubrio (en caso de que estos estén ajustados), e incline el eje en forma horaria o anti-horaria según corresponda compensar. Puede observar las figuras XII.2.7a y XII.2.7b presentadas a continuación como referencia para ejecutar este paso en forma correcta. Para la mayoría de los casos, el eje deberá permanecer alineado con la vertical.

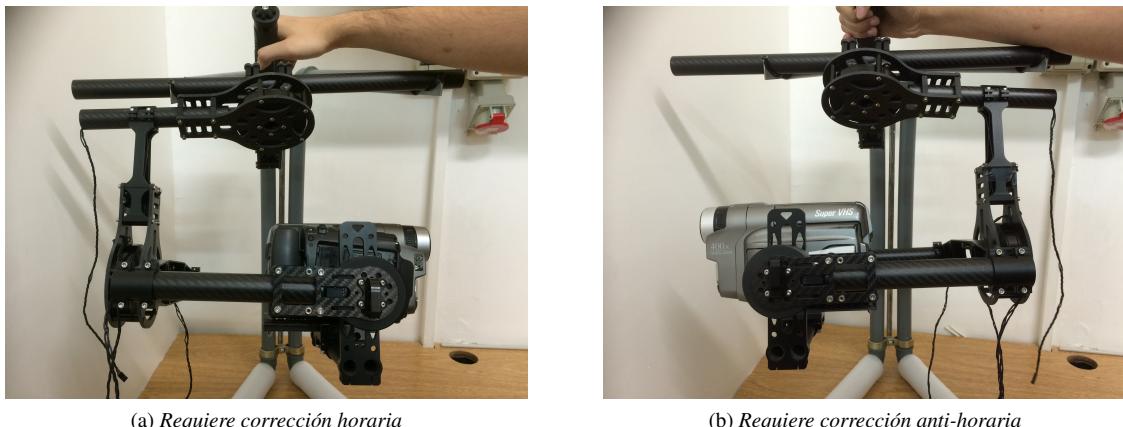


Figura XII.2.7.: Referencia para alineación rotacional de yaw

5. Repita los pasos 1 a 4 hasta que la cámara permanezca mirando hacia atrás luego de inclinarse la estructura, y termine el proceso de calibración ajustando los tornillos del eje.

El eje yaw ha sido exitosamente calibrado; la calibración física ha concluido.

XII.2.4 Calibración Digital

Los siguientes procedimientos calibran al STATHEROS desde un punto de vista digital; el objetivo de esta parte del manual es obtener valores adecuados para las variables de control P , I y D , de forma tal que el algoritmo de control funcione lo mejor posible.

Materiales Necesarios

- STATHEROS calibrado físicamente.
- Cámara.
- Estructura de calibración (o una segunda persona).
- Batería.
- Placa de control con IMU.

- Cable de alimentación.
- Cable de conectividad.
- Computador con software Equip.

Preparación del STATHEROS

1. Posicione el STATHEROS sobre la estructura de calibración, o, en caso de contar con una segunda persona para realizar la calibración, pídale a esa persona que sostenga al STATHEROS.
2. Coloque la batería.
3. Coloque el IMU en la bandeja de *pitch* o en la cámara.
4. Coloque la placa de control dentro de la carcaza protectora.
5. Conecte la placa de control a la computadora usando el cable de conectividad.
6. Inicie el software Equip en su computador y habilite la conexión con la placa de control.
7. Apague todos los motores y coloque todas las variables de control en cero.
8. Calibre el giroscopio.
9. Calibre el acelerómetro.
10. Conecte la placa de control a la batería con el cable de alimentación.

Su equipo está listo para ser calibrado digitalmente.

Calibración de un Eje

Debido a que cada eje rotacional del STATHEROS se comporta en forma independiente a los otros dos, la calibración de cada uno es independiente en todo aspecto excepto en uno: la potencia asignada a cada motor. A continuación se muestran pasos generales para calibrar un eje genérico, haciendo una salvedad en el paso donde se determina la potencia a entregar al motor. Se recomienda calibrar los ejes en el siguiente orden: *pitch*, *roll*, *yaw*.

1. Conecte el motor correspondiente al eje a calibrar a la placa de control.
2. Encienda los motores correspondientes al eje a calibrar, y apague el resto.
3. Asigne una potencia adecuada al eje; para esto considere que *yaw* debe tener la mayor potencia de los tres ejes, a *roll* le correspondería un 75 % de la potencia de *yaw*, y a *pitch* un 60 % de la potencia de *roll* aproximadamente.
4. Comience incrementando *P* con *I* y *D* en cero; busque el valor de *P_u* que hace oscilar al sistema.
5. Utilizando el graficador integrado en el software Equip, obtenga el valor del período de oscilación *T_p*.
6. Setee el valor de *P* en $0,6 \cdot P_u$.
7. Setee el valor de *I* en $2 \cdot P_u / T_p$.
8. Setee el valor de *D* en $P_u \cdot T_p / 8$.

La calibración digital ha concluido.

XII.2.5 Recambio de un Motor de Pitch

Esta parte del manual se avoca a describir los pasos a seguir al momento de recambiar un motor de *pitch* que ha dejado de funcionar.

Materiales Necesarios

- STATHEROS.
- Juego de llaves Allen.
- Motor funcional.

Preparación del STATHEROS

1. Asegúrese de que el motor esté desconectado de la batería y la placa de control.
2. Remueva la cámara de la bandeja de *pitch* para facilitar los procedimientos.
3. Suba la bandeja de *pitch* hasta el punto más alto posible para facilitar los procedimientos.

El equipo está listo para realizar el recambio.

Remoción del Motor

El primer paso es, por supuesto, remover el motor.

1. Posicione el STATHEROS de forma tal que el motor quede en la posición que se muestra en la Figura XII.2.8 presentada a continuación.

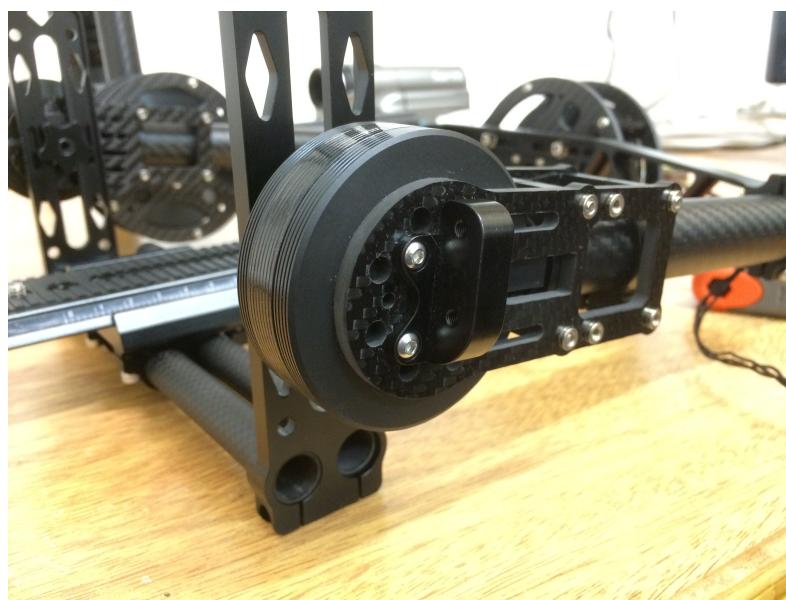


Figura XII.2.8.: Referencia para la posición deseada del motor

2. Remueva las tuercas que sujetan la bandeja a los motores.
3. Utilice una llave Allen para aflojar los cuatro tornillos que sujetan el sujetador del motor al STATHEROS, como se muestra en la Figura XII.2.9 presentada a continuación.

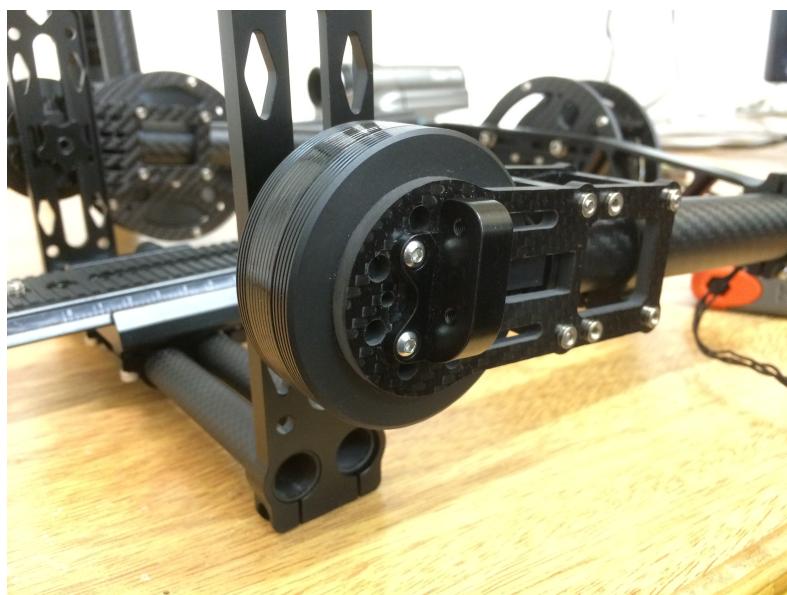


Figura XII.2.9.: Referencia para la remoción del sujetador

4. Remueva motor y sujetador de la estructura. Para desacoplarlo de la balanza se recomienda girar el sujetador hasta que el eje roscado salga de la vía.
5. Utilice una llave Allen para remover los cuatro tornillos que sujetan el estator del motor al sujetador, como se muestra en la Figura XII.2.10 presentada a continuación. Remueva la pieza completamente.



Figura XII.2.10.: Referencia para la remoción del tope del estator

6. Utilice una llave Allen para remover los tres tornillos que sujetan el rotor del motor al eje roscado, como se muestra en la Figura XII.2.11 presentada a continuación. Remueva la pieza del eje roscado, la arandela y el motor. Antes de continuar, asegúrese de tener todas las piezas separadas, como se muestra en la Figura XII.2.12 también presentada a continuación.



Figura XII.2.11.: Referencia para la remoción del eje roscado



Figura XII.2.12.: Referencia de piezas resultantes

7. Es momento de rearmar el STATHEROS; comience colocando nuevamente la pieza del eje roscado, e inserte la arandela como se muestra en la Figura XII.2.13 presentada a continuación.

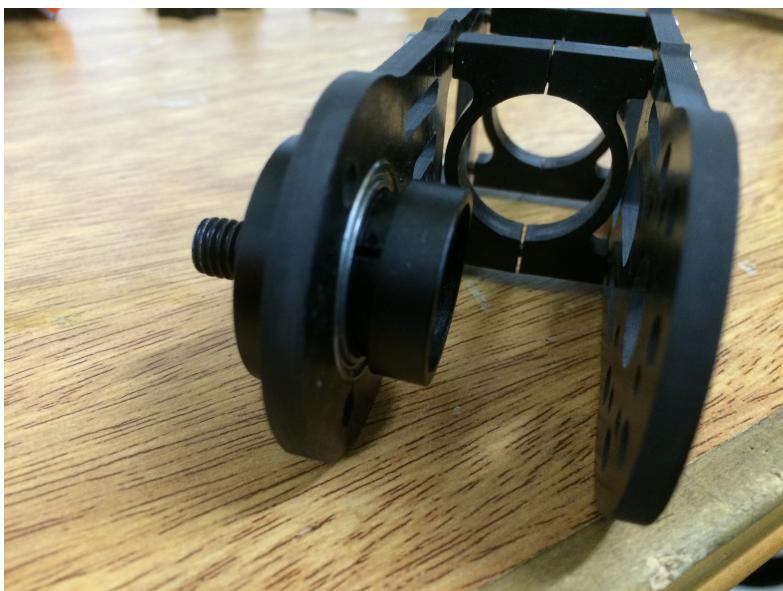


Figura XII.2.13.: Referencia para la colocación de la arandela

8. Conecte el cable al nuevo motor y deslícelo a su lugar; es altamente recomendable pasar el extremo libre del cable conector a través de los orificios del sujetador.
9. Utilice una llave Allen para ajustar los tres tornillos que sujetan el rotor del motor al eje roscado.
10. Coloque nuevamente la pieza que sujeta el estator del motor al sujetador, como se muestra en la Figura XII.2.14 presentada a continuación. Utilice una llave Allen para ajustar los cuatro tornillos que sujetan el estator del motor al sujetador. Antes de continuar asegúrese de que su motor y sujetador se ven como los de la Figura.

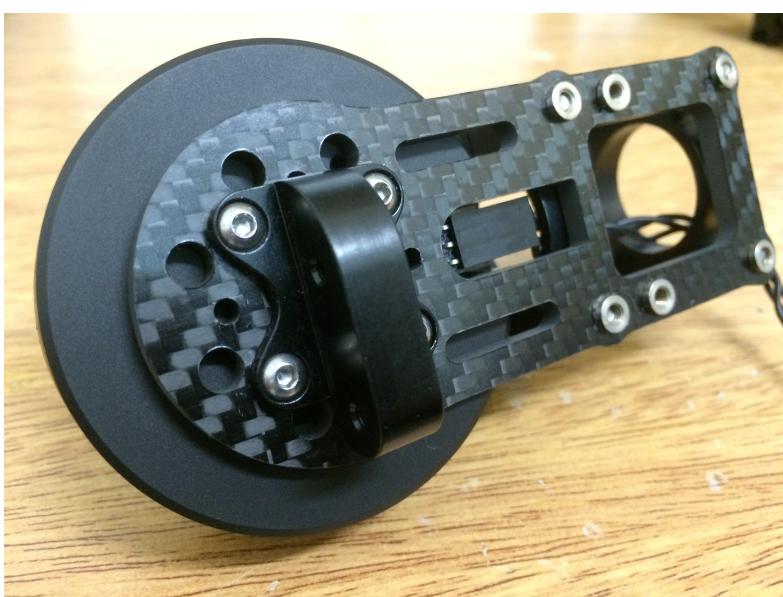


Figura XII.2.14.: Referencia para la colocación del tope del estator

11. Pase el extremo libre del cable conector a través del caño de fibra de carbono donde se coloca el sujetador. Una vez hecho esto, inserte el sujetador; como puede observarse en la Figura XII.2.15

presentada a continuación, el caño debe llegar al segundo orificio del sujetador, pero no avanzar mucho más de ese límite.

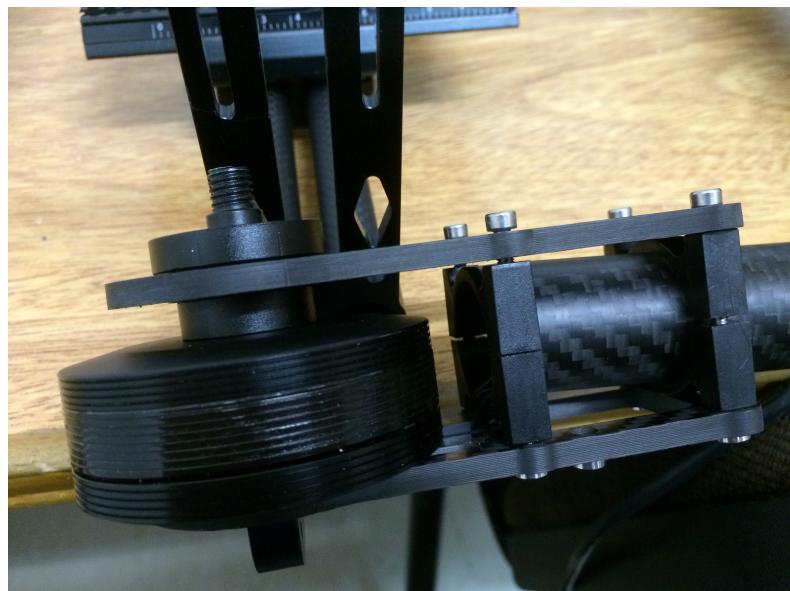


Figura XII.2.15.: Referencia para la colocación del sujetador

12. Acople el eje roscado del sujetador a la vía de la bandeja.
13. Utilice una llave Allen para ajustar los cuatro tornillos que sujetan el sujetador del motor al STAT-HEROS, como se muestra en la Figura XII.2.9 presentada anteriormente.
14. Asegure la bandeja ajustando la tuerca removida anteriormente.

El recambio del motor ha finalizado con éxito.

XIII Bibliografía

La bibliografía empleada para el proyecto se detalla en la siguiente lista.

- Datasheet: BAV99.
- Datasheet: L6234.
- Datasheet: AOZ1013AI.
- Datasheet: XRP6272.
- Datasheet: SRU1048.
- Datasheet: CP2104.
- Datasheet: PS-MPU-6000A.
- Datasheet: RM-MPU-6000A.
- FRDM-K20D50MUM User's Manual.
- K20 Sub-Family Reference Manual.
- Paper: Position and Speed Control of Brushless DC Motors Using Sensorless Techniques and Application Trends.
- Paper: A PID Feedback Control without a Subsidiary Speed Loop.
- Paper: AN - Brushless DC Motor Control Made Easy.
- Paper: Error Reduction Techniques for a MEMS-Accelerometer Based Input Device.
- Paper: Kalman Filter Implementation on an Accelerometer.
- Paper: Mass Properties Factors in Achieving Stable Imagery From a Gimbal.