



Instituto Politécnico Nacional

Centro de Investigación en Computación

Secretaría de Investigación y Posgrado

Controlador embebido sobre FPGA
para un vehículo aéreo no tripulado

T E S I S

QUE PARA OBTENER EL GRADO DE
MAESTRO EN CIENCIAS EN INGENIERÍA DE CÓMPUTO
CON OPCIÓN EN SISTEMAS DIGITALES

P R E S E N T A

ING. Edgar Martínez Navarro



DIRECTORES DE TESIS:

Dr. Marco Antonio Moreno Armendáriz

Dr. Sergio Salazar Cruz

MÉXICO, D.F.

2012



INSTITUTO POLITÉCNICO NACIONAL

SECRETARÍA DE INVESTIGACIÓN Y POSGRADO

ACTA DE REVISIÓN DE TESIS

En la Ciudad de México, D.F. siendo las 17:00 horas del día 28 del mes de mayo de 2012 se reunieron los miembros de la Comisión Revisora de la Tesis, designada por el Colegio de Profesores de Estudios de Posgrado e Investigación del:

Centro de Investigación en Computación

para examinar la tesis titulada:

“Controlador embebido sobre FPGA para un vehículo aéreo no tripulado”

Presentada por el alumno:

MARTÍNEZ

Apellido paterno

NAVARRO

Apellido materno

EDGAR

Nombre(s)

Con registro:

A	1	0	0	2	8	4
---	---	---	---	---	---	---

aspirante de: **MAESTRÍA EN CIENCIAS EN INGENIERÍA DE CÓMPUTO CON OPCIÓN EN SISTEMAS DIGITALES**

Después de intercambiar opiniones los miembros de la Comisión manifestaron **APROBAR LA TESIS**, en virtud de que satisface los requisitos señalados por las disposiciones reglamentarias vigentes.

LA COMISIÓN REVISORA

Directores de Tesis

Dr. Marco Antonio Moreno Armendáriz

Dr. Sergio Salazar Cruz

Dr. Luis Pastor Sánchez Fernández

Dr. Víctor Hugo Ponce Ponce

Dr. Marco Antonio Ramírez Salinas

PRESIDENTE DEL COLEGIO DE PROFESORES



Dr. Luis Alfonso Villa Vargas

DIRECCION



INSTITUTO POLITÉCNICO NACIONAL
SECRETARÍA DE INVESTIGACIÓN Y POSGRADO

CARTA CESIÓN DE DERECHOS

En la Ciudad de México D.F. el día 13 del mes Junio del año 2012, el (la) que suscribe Edgar Martínez Navarro alumno (a) del Programa de Maestría en Ciencias en Ingeniería de Cómputo con opción en Sistemas Digitales con número de registro A100284, adscrito a Centro de Investigación en Computación, manifiesta que es autor (a) intelectual del presente trabajo de Tesis bajo la dirección de Dr. Marco Antonio Moreno Arredáriz y Dr. Sergio Salazar Cruz y cede los derechos del trabajo intitulado Controlador embebido sobre FPGA para un vehículo aéreo no tripulado, al Instituto Politécnico Nacional para su difusión, con fines académicos y de investigación.

Los usuarios de la información no deben reproducir el contenido textual, gráficas o datos del trabajo sin el permiso expreso del autor y/o director del trabajo. Este puede ser obtenido escribiendo a la siguiente dirección sapehr@gmail.com. Si el permiso se otorga, el usuario deberá dar el agradecimiento correspondiente y citar la fuente del mismo.


Edgar Martínez Navarro

Nombre y firma

Resumen

Se presenta un sistema para determinar la orientación de un vehículo aéreo no tripulado (UAV por sus siglas en inglés) usando un FPGA (acrónimo en inglés de Field Programmable Gate Array).

Una solución UAV de bajo costo proveería de una alternativa atractiva a diversas aplicaciones civiles donde un helicóptero tradicional sería usado. Aplicaciones prácticas podrían incluir: supervisión del tráfico, operaciones de rescate y ayuda, desastres naturales, entre otras.

Un prototipo UAV de bajo costo será construido, el cual es una variación de helicóptero tradicional, ya que será impulsado por cuatro rotores montados en el fuselaje. Las características de diseño y selección de componentes serán expuestas extensivamente después en este trabajo. Se presenta una aproximación del modelo matemático a través de Euler-Lagrange, así como un controlador no lineal para estabilizar la orientación de la aeronave. La integración de todo los componentes del sistema (sensado, ejecución y control) se realiza por medio de la programación de lenguaje de descripción de hardware sobre FPGA, culminando así con la comprobación de la tesis durante el vuelo del prototipo.

Adicional al prototipo, se verificará el comportamiento del prototipo en un estudio de captura de movimiento con la intención de generar un estudio a partir de una referencia fija en un punto.

Abstract

A system for determining Unmanned Aerial Vehicle (UAV) attitude using a Field Programmable Gate Array (FPGA) and low cost gyroscopes is presented.

A simple low cost UAV solution would provide an attractive alternative to several civilian applications where a helicopter would traditionally be used. Practical applications could include: traffic surveillance, aiding search and rescue operations, natural disasters, among others.

A low cost prototype UAV is being built, it is a variation of a traditional helicopter, because it will be propelled using four rotors mounted to a fuselage. The design details and component selection will be discussed more extensively later in the thesis. This work includes the mathematical model that describes the quadrotor's dynamics using Euler-Lagrange movement equations, and also a proposed linear controller will be used for attitude stabilization of the UAV. The integration all the system components (sensors, execution and control) is done using hardware description language on FPGA.

Among the prototype, the flight behavior will be verified in a motion capture studio with the objective of having a complete description of the dynamics given by a fixed reference point.

Agradecimientos

A Norma y Víctor, por haberme enseñado que con amor y dedicación, todo es alcanzable, siempre y cuando se lucha por conseguirlo. Su vida ha sido el ejemplo más grande en mi persona y siempre serán mi modelo a seguir.

A Axel, por siempre tener en tí a alguien en quien confiar y con quien platicar acerca de cualquier cosa. Por creer que el mundo debería de ser más justo y ser un ejemplo de que hay que romper paradigmas.

A Jessica con todo mi amor, por siempre apoyarme en todas mis decisiones, por aceptarme en mis ratos de locura y por ser mi gran pilar para construir una vida juntos.

A Elvira, Dolores, Manuel y Tomás, por siempre apapacharme y consentirme durante toda mi vida, por enseñarme que con amor todo es posible y por ayudarme a crecer y demostrar que cada día podemos ser mejores.

*A Claudia, Laura, Miguel, Juan
Carlos, Judith, Edgardo y Héctor, por
haberme enseñado lo que significa el
calor de hogar, lo que es una familia y
lo importante de ésta.*

*A Santiago, Juan Carlos, Miguel,
Fatima, Areli, Diego, Sebastián y
Daniela, para que esto sea aliciente
para ustedes.*

*A Duch, Víctor, Jaime, Jesús,
Christian, Iván, Ricardo, Steed, Diego,
Abraham, Alejandro, por darme una de
las mejores etapas de mi vida y
demostrarme que siempre podemos
sonreír ante cualquier situación y
aprender de cada experiencia.*

*A Sergio y Marco, por haberme guiado
a través de mi tesis de maestría y
concluirla exitosamente.*

*To Kathrin, Jochen, Luis and
Guillermo, for being part of my dream
and motivation to accomplish this goal.
Thank you for being my family in
Germany.*

Índice general

1	Introducción	11
1.1	Antecedentes	11
1.2	Planteamiento del problema	13
1.3	Justificación	13
1.4	Hipótesis	14
1.5	Objetivos	14
1.6	Alcances del trabajo	15
1.7	Contribuciones	15
1.8	Estancia de investigación y productos PIFI	15
1.9	Método de investigación y desarrollo utilizado	16
1.10	Organización del trabajo	16
2	Estado del arte	17
2.1	Prototipos	17
2.2	Modelos matemáticos	20
2.3	Control e implementación	22
2.4	Resumen	24
3	Prototipo	25
3.1	Diseño mecánico	26
3.2	Diseño eléctrico	27
3.3	Protocolos de comunicación	32
3.4	Resumen	35
4	Modelado matemático	37
4.1	Características del quadrotor	37
4.2	Modelado dinámico	39
4.3	Resumen	43
5	Estabilización	45
5.1	Linealización	45
5.2	Control de orientación	48
5.3	Simulación numérica	50
5.4	Resumen	53

6 Implementación en hardware	55
6.1 Consideraciones	55
6.2 Componentes de hardware	58
6.3 Recursos utilizados	69
6.4 Resultados experimentales	70
6.5 Resumen	75
7 Conclusiones y trabajo futuro	77
7.1 Conclusiones y aportaciones	77
7.2 Trabajo futuro	78
Apéndice	82
A Planos de dibujo	83
B Código fuente	85

Índice de figuras

1.1	Helicóptero con diseño tandem	11
1.2	Proyecto quadrotor del Dr. Bothezat	12
1.3	Quadrotor policía Qube	12
1.4	Giroscopio MEMS	13
2.1	Proyecto ETH OS4	18
2.2	Proyecto Compiegne	19
2.3	Proyecto ANU MARK II	20
2.4	Diagrama de bloques del diseño del controlador embebido en FPGA	24
3.1	Diseño de piezas en SolidWorks 2009	26
3.2	Prototipo de quadrotor basado en FPGA	26
3.3	Diagrama de bloques del sistema eléctrico del quadrotor	28
3.4	Central inercial Sparkfun SEN-10010	28
3.5	Filtro activo pasa bajos Sallen Key	29
3.6	Microcontrolador Atmel ATmega161	29
3.7	Circuito optoacoplado para la transmisión UART	30
3.8	Quadrotor con configuración X	30
3.9	Receptor FM ACT DSL4-Top	31
3.10	Patrón de franjas verticales	31
3.11	Circuito para la medición de velocidad de los rotores	31
3.12	Tarjeta de desarrollo CMCS002-2M	32
3.13	Controlador de velocidad y motor sin escobillas	32
3.14	Circuito eléctrico del protocolo UART	33
3.15	Protocolo de comunicación UART	33
3.16	Circuito eléctrico del protocolo I ² C	34
3.17	Protocolo de comunicación I ² C	35
4.1	Esquema básico de un quadrotor	38
4.2	Ángulos de orientación del quadrotor	38
4.3	Diagrama de cuerpo libre del quadrotor	40
5.1	Diagrama de bloques del esquema de control	49
5.2	Diagrama de bloques de la simulación en LabVIEW	50
5.3	Gráficas de simulación de los ángulos y velocidades angular	51

5.4	Gráficas de simulación de las acciones de control	52
6.1	Diagrama de bloques de la nueva estructura de control	56
6.2	Esquema de un motor de CD	57
6.3	Diagrama de bloques de los componentes de hardware generales	59
6.4	Diagrama de bloques de los subcomponentes de <i>imu</i>	59
6.5	Diagrama de bloques de los subcomponentes de <i>dsl</i>	60
6.6	Diagrama de bloques de los subcomponentes de <i>spdfb</i>	61
6.7	Gráficas de la relación de los valores de entrada-salida de los rotores	62
6.8	Diagrama de bloques simplificado del controlador de estabilización	63
6.9	Diagrama de bloques de los subcomponentes de <i>ctrl</i>	66
6.10	Diagrama de bloques de los subcomponentes de <i>esc</i>	68
6.11	Trama I ² C para actualizar la velocidad de los motores	68
6.12	Diagrama de bloques de los subcomponentes de <i>smalltest</i>	69
6.13	Gráficas experimentales de los ángulos y velocidades angulares	71
6.14	Gráficas de la evolución de las acciones de control en el prototipo	72
6.15	Gráficas experimentales de la velocidad angular medida de los motores	73
6.16	Gráficas experimentales de los valores I ² C de los motores	73
6.17	Visor tridimensional rviz de ROS	74
6.18	Hangar de vuelo en la Universidad de Freiburg, Alemania	74

Índice de tablas

2.1	Comparativa de quadrotores comerciales	20
2.2	Principales proyectos internacionales relacionados con helicópteros multirotor	22
5.1	Ganancias para la simulación numérica del controlador PD	51
6.1	Recursos del FPGA utilizados por el diseño	69
6.2	Ganancias experimentales para controlador embedido en FPGA	70

Capítulo 1

Introducción

1.1 Antecedentes

Un helicóptero puede ser definido como cualquier máquina que es capaz de volar usando alas rotativas, por ejemplo rotores que lo proveen de elevación, propulsión y fuerzas de control. El rotor produce una fuerza de elevación igual al peso de helicóptero y debido a la generación de dicha fuerza es capaz de levantarse verticalmente de suelo y permanecer suspendido [18].

Entonces, un helicóptero se podría redefinir como una aeronave que hace uso de rotores y es capaz de permanecer suspendido en vuelo. Cuando éste inclina la orientación de sus rotores, se generan fuerzas y momentos que controlan el helicóptero en vuelo, originando desplazamientos en posición y orientación a deseo del piloto.

Dentro de este principio fundamental se crearon diversos conceptos que divergen del convencional, él cual posee un rotor principal y un rotor en la cola. Uno de estos modelos se materializó en el llamado helicóptero coaxial que incorpora un rotor adicional en su estructura central, mientras que otros diseños, como el tipo tandem que redistribuye ambos rotores horizontalmente, uno al frente y el otro atrás de la aeronave. La figura 1.1 muestra una fotografía de un helicóptero tandem.

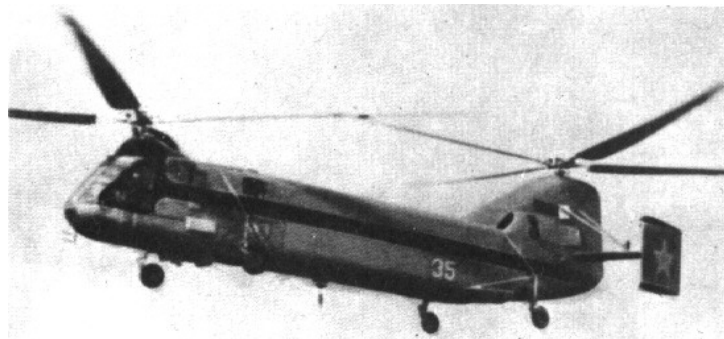


Figura 1.1: Helicóptero con diseño tandem

Una perspectiva de diseño más fue ideada en 1921 por el Dr. George de Bothezat y el Dr. Ivan Jerome en conjunción con la US Army Air Corps para desarrollar una máquina con capacidad de volar verticalmente. La estructura tenía la forma de una cruz, un peso de 1678 Kg y una longitud en cada brazo de nueve metros; dicho brazo era capaz de sostener un rotor con seis aspas que en conjunto hacían un diámetro de 8.1 metros. Cada rotor tenía un control individual de pitch con la finalidad de producir una diferencia en la fuerza de empuje en el vehículo para su traslación.

Su primer vuelo se realizó en Ohio en el año 1922 y se realizaron aproximadamente 100 despegues más subsecuentes, uno de ellos con tres pasajeros. La idea teórica al inicio del proyecto establece que la aeronave debía de permanecer suspendida a una altura de 100 metros, pero realmente sólo logró cinco metros de altura. Esto demostró que las ideas de Bothezat eran teóricamente posibles, pero en la práctica no tuvo la suficiente potencia, la mecánica era demasiado compleja y la carga de trabajo para el piloto era excesiva durante la sustentación por lo cual nunca se intentó el desplazamiento lateral [19]. La figura 1.2 muestra la imagen del primer quadrotor construido por el Dr. Bothezat.



Figura 1.2: Proyecto quadrotor del Dr. Bothezat

En 1924, Étienne Cehmichen construyó un prototipo similar al de Bothezat con el que estableció un récord de distancia alcanzando el primer kilómetro volado lateralmente.

Actualmente este concepto ha sido retomado por la industria militar para desarrollar aeronaves a escala, las cuales han recibido la designación de vehículo aéreo no tripulado y sus principales características consisten en la reusabilidad y la capacidad de mantener un nivel de vuelo controlado y sostenido. Aunado a esto, en las últimas décadas cada vez más instituciones académicas y empresas comerciales han hecho grandes investigaciones y aportaciones a los vehículos aéreos no tripulados (por sus siglas en inglés UAV) principalmente por dos razones.

La primera consiste en la demanda de estas aeronaves en tareas de protección civil o aplicaciones militares, donde variantes de diseño entre aviones y helicópteros han sido fusionadas para incrementar la autonomía del vehículo, garantizar la estabilización a distintas velocidades de vuelo y el almacenamiento o transmisión segura de la información. La figura 1.3 muestra al Qube, el cual es un quadrotor prototipo diseñado para la policía de Estados Unidos para ser ensamblado en menos 5 minutos, es energizado por medio de baterías durante 40 minutos, es controlado por una tableta y está equipado con cámaras de color y térmicas.



Figura 1.3: Quadrotor policía Qube

La segunda se deriva a la consolidación de nuevas disciplinas relacionadas principalmente con el diseño y fabricación de sistemas microelectromecánicos (por sus siglas en inglés MEMS), cuyo principal objetivo es la miniaturización de sensores, actuadores, entre otros, dotándolos de mayor precisión, menor tiempo de respuesta, bajo consumo de energía en conjunción a la reducción de sus propiedades físicas como dimensiones y peso. La figura 1.4 muestra un giroscopio hecho en esta tecnología.

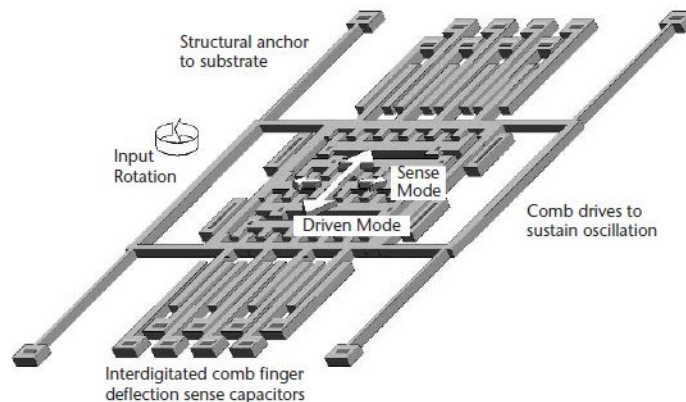


Figura 1.4: Giroscopio MEMS

Debido a la gran demanda existente en el mercado y a la producción de nuevos sensores y actuadores que potencializan las capacidades de vuelo, cada vez más instituciones hacen más aportaciones a la investigación de esta área y exponen nuevos prototipos, modelos, leyes de control, así como nuevas aplicaciones.

1.2 Planteamiento del problema

Según el INEGI, los fenómenos naturales se convierten en calamidades cuando causan muertes, lesiones o daños en los bienes y sistemas ecológicos. Los efectos de las calamidades naturales sobre los asentamientos humanos tienen particular importancia, pues la probabilidad de tales efectos parece intensificarse como consecuencia del mayor desarrollo de asentamientos en zonas potencialmente peligrosas [14].

Cifras adicionales de este organismo señalan que en el año 2010, las inundaciones fueron los desastres naturales más devastadores. Entre mayo y octubre se pronosticaron 14 fenómenos en Norte y Centroamérica, con más actividad para la cuenca del Atlántico como consecuencia del fenómeno de El Niño. En ese mismo año, se registraron más de 750 fenómenos meteorológicos extremos en el planeta, lo que marcó un nuevo récord en magnitud, frecuencia y alcance.

Aunado a los problemas derivados por los fenómenos naturales, existen factores sociales que causan situaciones de riesgo para la población civil, que van desde zonas con altos niveles de delincuencia hasta mítines en cárceles o en cualquier otro espacio cerrado.

En México existe la necesidad de crear tecnología capaz de realizar tareas de búsqueda o labores de reconocimiento en zonas de difícil acceso, las cuales no requieren de una tripulación que sobrevuele una aeronave. Inclusive, esta tecnología se podría extrapolar en situaciones en las cuales las vidas de un aeronave tripulada se encuentren en riesgo como es el caso de los desastres naturales o sociales.

1.3 Justificación

La misión del Centro Nacional de Prevención de Desastres (CENAPRED) consiste en prevenir, alertar y fomentar la cultura de autoprotección para reducir el riesgo de la población ante fenómenos naturales y antropogénicos que amenacen sus vidas, bienes y entorno a través de la investigación, monitoreo, capacitación y difusión [8]. Las autoridades mexicanas disponen de organismos como el CENAPRED para disminuir las afectaciones cuando los desastres ocurren, pero desgraciadamente existen situaciones que aún con la mayor prevención, las emergencias se convierten en calamidades.

Para combatir dichas emergencias, se propone diseñar un controlador embebido para un vehículo aéreo autónomo capaz de sobrevolar zonas de difícil acceso o que representen peligro para una aeronave tripulada. Dicho controlador recibirá como señal de entrada la posición deseada y será capaz de llevar a la aeronave a dicho lugar. El vehículo a controlar será del tipo quadrotor y se eligió esta aeronave debido a que cuenta con características de los helicópteros como lo son el vuelo estacionario, el despegue y aterrizaje vertical, pero posee una propiedad única que es la estabilidad durante el vuelo, originada por la simplicidad de su diseño mecánico.

1.4 Hipótesis

Es posible construir un vehículo aéreo no tripulado de tipo quadrotor con propiedades autónomas, su funcionamiento estará basado en la implementación de un controlador embebido sobre un FPGA, el cual ofrece una gran potencialidad para la incorporación de sensores, actuadores y leyes de control, así como telemetría, ya que las características de éste sobrepasa cualquier otro dispositivo usado como unidad de procesamiento. Con ello se asegura una respuesta en tiempo necesaria para estabilizar y compensar la dinámica rápida del quadrotor, además de lidiar con las perturbaciones externas al sistema.

Para sustentar esta hipótesis y comprobarla se plantea un modelo matemático basado en las ecuaciones de Euler-Lagrange y la simulación mediante el software de LabVIEW, empleándose para realizar la verificación del controlador sin la necesidad de hacerlo directamente con el prototipo físico.

1.5 Objetivos

Con base en lo expuesto en el planteamiento del problema, la justificación y la hipótesis, se propone en este trabajo de investigación los siguientes objetivos:

Objetivo general

Diseñar un controlador embebido en un FPGA para a un vehículo aéreo no tripulado.

Objetivos específicos

1. Diseñar y construir un prototipo de un quadrotor, el cual se utilizará como modelo para la experimentación y verificación de la hipótesis.
2. Utilizar como base el modelo matemático del quadrotor propuesto en el libro *Unmanned Aerial Vehicles Embedded Control* [19] para proponer un controlador capaz de estabilizar al vehículo con la finalidad de realizar vuelo suspendido.
3. Validar mediante simulaciones el desempeño del controlador diseñado.
4. Embeber sobre FPGA el controlador usando un lenguaje de descripción de hardware con el fin de garantizar la autonomía del sistema.

1.6 Alcances del trabajo

El alcance de este trabajo es diseñar y construir un quadrotor, el cual será estabilizado por la implementación de un controlador embebido en un FPGA. Es importante hacer notar que el trabajo concluirá con la fase experimental del prototipo evaluando su autonomía en vuelo estacionario.

Es decir, este trabajo se limita a establecer un controlador de orientación el cual permite estabilizar al quadrotor y le da la posibilidad a un piloto competente disponer de éste para

realizar movimientos laterales a través de un transmisor de radio control. Este proyecto no incluye un controlador traslacional, por lo que no se puede establecer al helicóptero de un plan de vuelo basado en coordenadas o marcas y que éste autónomamente lo recorra.

1.7 Contribuciones

Este trabajo contribuye con el diseño de un controlador lineal completo para un vehículo aéreo no tripulado tipo quadrotor, en donde todos los componentes sean diseñados y optimizados para reducir el tiempo de respuesta y minimizar la cantidad de recursos utilizados. Aunado a esto, todas las interfaces con los periféricos como sensores o actuadores o telemetría serán gestionados por medio del FPGA evitando así el uso de componentes externos como microcontroladores o cualquier otro dispositivo. Garantizando así una solución paralelizada pero concentrada en un solo dispositivo, permitiendo así que si existe algún cambio en algún periférico o se requiere uno nuevo, las modificaciones son mínimas.

1.8 Estancia de investigación y productos PIFI

El autor de este texto realizó una estancia de investigación de 4 meses en el Laboratorio de Sistemas Inteligentes Autónomos de la Universidad de Freiburg, Alemania, bajo la tutela del Dr. Slawomir Grzonka y cuyo objetivo consistió en aplicar los principios teóricos en el vuelo de una plataforma así como sintonizar un controlador usando la información obtenida del estudio de captura de movimiento.

Dentro de la actividades concluidas se encuentra la descripción del modelo del quadrotor basado en la integración de los datos internos de la aeronave y del sistema de captura de movimiento y el vuelo autónomo a través de sistema de captura de movimiento, para vuelo suspendido y seguimiento de trayectoria.

En complemento a la estancia de investigación, se redactó un artículo publicado en revista indexada. Los detalles de este artículo se muestra a continuación.

- The trajectory tracking problem for an unmanned four-rotor system: Flatness-based approach, C. Aguilar-Ibáñez, H. Sira-Ramírez, M. S. Suárez-Castañón, E. Martínez-Navarro and M. A. Moreno-Armendariz, *International Journal of Control* (0020-7179), 85(1), 69-77, 2012.

1.9 Método de investigación y desarrollo utilizado

Para poder construir el prototipo del quadrotor se realizó una revisión del estado del arte, en la cual se encontraron distintas alternativas para el diseño y los materiales, de los cuales se seleccionaron los más adecuados para la construcción del prototipo. Después se procedió al desarrollo del modelo matemático que engloba todos los parámetros experimentales del prototipo para garantizar su semejanza.

Por otra parte se diseñará un controlador lineal y se usará el software LabVIEW para su verificación durante la fase de pruebas, y se evaluará su respuesta en base a los parámetros

de diseño expuestos en la literatura.

Por último, este controlador será implementado en lenguaje de descripción de hardware, completando de esta manera el prototipo, evaluando su desempeño en vuelo suspendido.

1.10 Organización del trabajo

Este trabajo está dividido en 7 capítulos, los cuales se detallan a continuación.

El capítulo 2, que lleva por nombre estado del arte, hace un compendio y a partir de éste un análisis de las principales ideas que fueron usadas para la realización de este trabajo. El capítulo 3, prototipo, muestra el proceso de diseño, fabricación y construcción del prototipo empleado para la experimentación. El capítulo 4, modelado matemático, desglosa el modelo matemático no lineal a partir de la ecuaciones de Euler-Lagrange del quadrotor. El capítulo 5, estabilización, se propone y diseña un controlador para la orientación del vehículo, este controlador es verificado a través de una simulación numérica computacional. El capítulo 6, implementación en hardware, detalla cada uno de los pasos empleados para la construcción y síntesis del proyecto en lenguaje de descripción de hardware así como los resultados experimentales obtenidos del prototipo. Por último, el capítulo 7, engloba las conclusiones y trabajo futuro correspondientes a este trabajo.

Capítulo 2

Estado del arte

Un quadrotor es una aeronave que es propulsada por cuatro rotores y su movimiento se genera variando la velocidad relativa de cada uno de estos, con el fin de modificar la fuerza de empuje y el torque, generando así un cambio en la posición y orientación de la aeronave.

Históricamente, existen dos generaciones principales del diseño de los quadrotores. La primera generación, fue en la década de los 20, se diseñó para el transporte de pasajeros (máximo tres), tardó más de dos años para que el prototipo se pudiera poner en marcha y su elevación máxima fue de cinco metros. A pesar de esto, las aeronaves prototipo tenían un desempeño muy pobre (hipotéticamente se deberían elevar hasta 100 metros) y requerían de un gran trabajo y conocimiento del piloto para maniobrarlas, debido a que no eran muy estables [19].

La segunda generación, la cual inicio al principio de los 90, se diseñó con la intención de fabricar UAV y se utilizan sistemas electrónicos para pilotear y garantizar la estabilidad de la aeronave, aunado a un tamaño reducido, brindando con esto una alta maniobrabilidad para los modelos existentes.

Existen múltiples aportaciones de investigadores y universidades a este rubro a lo largo del mundo, proponiendo e integrando diferentes métodos para el diseño, construcción, control e implementación de helicópteros multirotor. A continuación se detallan aquellas aportaciones que sirvieron de guías para la realización de este proyecto de tesis en los diferentes rubros.

2.1 Prototipos

Cualquier helicóptero multirotor se construye a partir de una idea general, la cual consiste en designar la cantidad de rotores que el prototipo tendrá (la cantidad mínima es de 3), para cada rotor se designa un brazo que unirá al rotor con el centro del helicóptero. Teniendo esta idea en mente, por la distribución de masa principalmente derivada de la baterías, el centro de gravedad teóricamente debería de coincidir con el centro geométrico de la aeronave (suponiendo que la estructura es simétrica), otorgando así amortiguamiento físico al sistema.

Es de suma importancia aproximar el peso final de la aeronave para así seleccionar los motores y las hélices correctamente, buscando así la mejor relación del torque del rotor y consumo de energía, minimizando la descarga de las baterías.

Oficialmente, no existe una guía que describa el diseño, los materiales o los componentes con los cuales construir un helicóptero multirotor, pero a través de los resultados de los trabajos publicados e información disponible en Internet, los errores durante la fase de diseño y selección de materiales se reduce notablemente.

Al ser esto una realidad los centros de investigación y universidades alrededor del mundo fueron pioneros y desarrollaron diversos prototipos. A continuación se enlistan algunos de estos proyectos que motivaron al resto de las instituciones científicas alrededor del mundo.

Swiss Federal Institute of Technology

El proyecto del Swiss Federal Institute of Technology consistió en el diseño mecánico, modelado dinámico, sensado y control de un robot autónomo de despegue y aterrizaje vertical (VTOL, por sus siglas en inglés) cuyo nombre fue OS4. Es importante resaltar que tres de los seis grados de libertad que posee la aeronave fueron bloqueados.

Desde el puerto RS232 de un computadora personal, las señales de control eran enviadas al OS4. Se incorporaron módulos de conversión protocolos de RS232 a I²C, este último es el utilizado por los controladores de velocidad para motores sin escobillas. Estos módulos contienen un controlador PID en un microcontrolador PIC16F876 y son capaces de operados en lazo abierto o cerrado. Utiliza una central inercial MT9-B8, la cual estima a través de un filtro de Kalman la orientación tridimensional y brinda los datos calibrados de la aceleración y velocidad a una tasa de transferencia de 115,200 bps.

Los brazos son hechos de fibra de carbón tubular a través del vehículo y su masa es de 240 gramos. Cada motor tiene un peso de 29 gramos incluyendo los *encoders* magnéticos, más un tren de engranes de 6 gramos y una hélice que también pesa 6 gramos. La figura 2.1 muestra la fotografía del robot OS4.

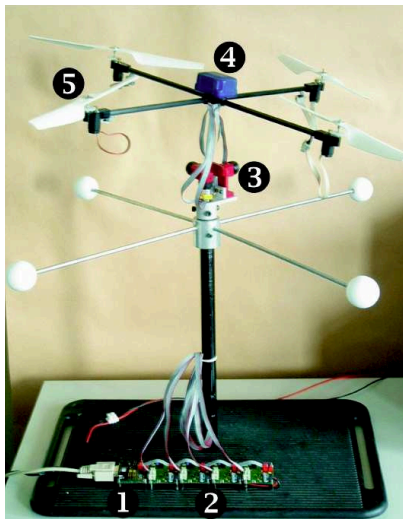


Figura 2.1: Proyecto ETH OS4

University of Technology in Compiègne

El quadrotor usado en este proyecto fue manufacturado especialmente por la compañía Draganfly. Las señales de control fueron enviadas a través de un radiotransmisor Futaba Skysport 4.

El receptor de radio y la computadora personal fueron conectadas usando una tarjeta de adquisición de datos Advantech PCL-818HG y PCL-726. La central inercial Polhemus fue conectada por medio del puerto RS232 a la computadora personal. Este sensor tiene ciertas características que lo hacen muy sensible al ruido electromagnético, por lo que fue instalado lo más lejos posible del circuito de los rotores y de sus controladores de velocidad. El quadrotor posee 3 giroscopios que ayudan a la estabilización de la aeronave. La figura 2.2 hace referencia al proyecto de la University of Technology in Compiègne.



Figura 2.2: Proyecto Compiègne

Australian National University

El proyecto de la Australian National University es llamado Mark II, el cual es un quadrotor de 2 kilogramos y una longitud de 70 cm.

El diseño de la electrónica y la central inercial fueron hechos especialmente por el CSIRO ICT Centre. La tarjeta de control contiene dos microprocesadores HC-12 con entradas y salidas de propósito general. La central inercial de nombre Eimu, es una central inercial completa de 6 ejes con magnetómetros. Esta es usada en el modo vertical para obtener los ángulos del marco de referencia inercial. El diseño de este quadrotor considero un espacio para la central inercial aproximadamente en el centro de gravedad de la aeronave.

En conjunción con el uso de baterías, este quadrotor fue de los pioneros en realizar el cálculo del control a bordo de la propia aeronave sin la necesidad de recurrir a una computadora personal, siendo así el primer autocontrolado a bordo. La figura 2.3 muestra la fotografía del proyecto Mark II.

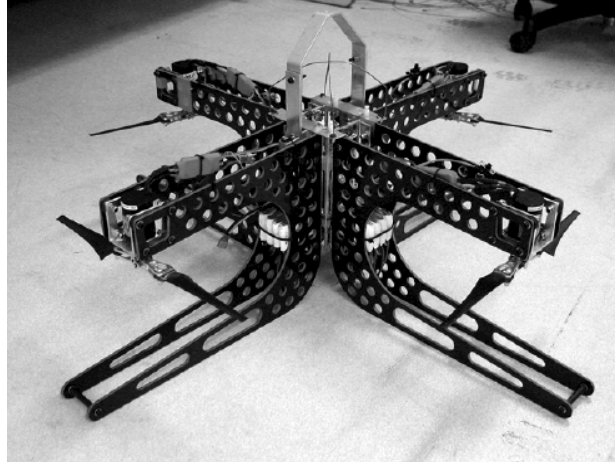


Figura 2.3: Proyecto ANU MARK II

Prototipos comerciales

Actualmente existen compañías que llevaron todos estos adelantes tecnológicos y convirtieron a los quadrotores en plataformas comerciales multidisciplinarias, es decir, los consumidores van desde personas que disfrutan pilotear modelos a escala de radio control, o aquellos que gusten la fotografía o vídeo aéreo hasta instituciones de investigación que prefieren comprar un prototipo ya construido y probar variantes a nivel de software. La tabla 2.1 muestra una breve comparativa entre las empresas que proveen al mercado con prototipos ya construidos.

Marca	Licencia del código	Características
Draganfly	Completamente cerrado	Comercial
AscTec	Completamente cerrado	Comercial
MikroKopter	No es libre, porciones del código disponible	C, microcontrolador AVR
NG-UAVP	Código libre y abierto	C y C++, CPU ARM7

Tabla 2.1: Comparativa de quadrotores comerciales

Los precios de estos helicópteros multirrotor oscilan de los \$1,500 a los \$10,000 dólares al momento de realizar esta tesis y no existen representantes de estas marcas en México.

2.2 Modelos matemáticos

En los últimos años, se han propuesto varios modelos referentes al quadrotor. La mayor parte de ellos obtienen el modelo tomando el vehículo como un cuerpo rígido, algunos otros agregan algunos términos aerodinámicos como son el cálculo de la matriz de inercia, los momentos giroscópicos y de coriolis [6].

También es importante hacer notar que existen diversos métodos para obtener las ecuaciones

de movimiento de un quadrotor, la aproximación mayormente encontrada en la literatura es a partir de las ecuaciones de movimiento de Newton, cuya ventaja principal radica en que es posible conocer o analizar las fuerzas generalizadas que se requieren aplicar al vehículo para lograr un movimiento deseado de las coordenadas generalizadas, o bien, a través de las ecuaciones de Euler-Lagrange cuyo enfoque permite describir de una manera más amplia la dinámica de las coordenadas generalizadas.

Ecuaciones de Newton

Modelando el quadrotor como un sistema rígido, podemos denotar a \mathcal{I} como el marco de referencia inercial y a \mathcal{A} como el marco de referencia sobre el cuerpo del vehículo. Utilizando la parametrización de los ángulos de Euler, la orientación del vehículo en el espacio se obtiene por una rotación \mathcal{R} desde \mathcal{A} hasta \mathcal{I} donde \mathcal{R} es una matriz de rotación.

Entonces a partir de las ecuaciones de Newton-Euler:

$$\begin{aligned}\dot{\xi} &= v & (2.1) \\ m\dot{v} &= f \\ \dot{R} &= R\hat{w} \\ \mathbf{I}\dot{w} &= -w \times \mathbf{I}w + \tau\end{aligned}$$

donde $\xi = (x, y, z)^T$ denota la posición del centro de masa del cuerpo en el eje de referencia \mathcal{I} , $v \in \mathcal{I}$ representa la velocidad lineal, $w \in \mathcal{A}$ denota la velocidad angular, m es la masa del vehículo, $\mathbf{I} \in \mathbb{R}^{3 \times 3}$ es la matriz de inercia (con respecto a \mathcal{A} , \hat{w} es la matriz antisimétrica del vector w , $f \in \mathcal{I}$ representa el vector principal de fuerzas no conservativas aplicadas al vehículo incluyendo la fuerza principal u y los términos de fricción asociados con los motores, $\tau \in \mathcal{A}$ representa los momentos que actúan en el vehículo.

El vector principal de fuerzas no conservativas f , se obtiene como:

$$f = R_{E_z}u - mgE_z \quad (2.2)$$

donde g es la fuerza de gravedad que actúa sobre Ez .

Entonces los momentos de inercia que proveen al vehículo de movimiento, se definen como:

$$\tau_A = \begin{bmatrix} \sum_{i=1}^4 \tau_{M_i} \\ l(f_1 + f_2 - f_3 - f_4) \\ l(f_2 + f_3 - f_1 - f_4) \end{bmatrix} = \begin{bmatrix} \tau_\psi \\ \tau_\theta \\ \tau_\phi \end{bmatrix} \quad (2.3)$$

donde τ_{M_i} es el torque de i ésimo rotor, l es el largo entre los motores y el centro de gravedad, f_i es la fuerza de empuje generada por el i ésimo motor. El conjunto de $\tau_\psi, \tau_\theta, \tau_\phi$ denotan los momentos de inercia que generan movimiento sobre los ángulos de yaw, pitch y roll respectivamente.

Para la simplificación del modelo, se considera que la inercia de los rotores es muy pequeña y por lo tanto despreciable, por ende no existen efectos giroscópicos.

Finalmente el modelo matemático se puede expresar como:

$$\begin{aligned}\dot{\xi} &= v & (2.4) \\ m\dot{v} &= R_{E_z}u - mgE_z \\ \dot{R} &= R\hat{w} \\ \mathbf{I}\dot{w} &= -w \times \mathbf{I}w + \tau_A\end{aligned}$$

Ecuaciones de Euler-Lagrange

Este trabajo hace uso de la aproximación de Euler-Lagrange, el modelo matemático detallado a través de este método se encuentra en el capítulo 4 con título modelo matemático.

2.3 Control e implementación

Las técnicas de control usadas en los helicópteros multirotor son muy variadas desde técnicas control clásico, hasta control inteligente como inteligencia artificial o visión. La tabla 2.2 muestra la relación de los principales proyectos expuestos por sus correspondientes universidades y la técnica de control empleada.

Proyecto	Técnica de control
STARMAC, Starford University (2005)	Aprendizaje reforzado
OS4, EPFL (2007)	Backstepping
Pensylvania State University (2007)	PI
Helio-copter, Brigham University (2008)	Visión artificial
HMX-4, Pennsylvania State Univesity (2002)	Retroalimentación lineal
Quad-Rotor Flying Robot, Universiti Teknologi Malaysia (2006)	PID

Tabla 2.2: Principales proyectos internacionales relacionados con helicópteros multirotor

Actualmente el control y su implementación se consideran dos tópicos que van fusionados, ya que es imposible pensar en una técnica de control no lineal o algún algoritmo de visión artificial, la cual haga un uso extensivo de matrices u otros elementos matemáticos, los cuales consumen muchos recursos o procesamiento dentro de un dispositivo; y que por dadas razones dicha técnica o algoritmo no puedan ser implementados en el dispositivo utilizado para realizar el computo.

Cabe resaltar que cada vez ésta complejidad aumenta, ya que más investigadores alrededor del mundo desarrollan nuevas técnicas de control aplicadas a la estabilización y posicionamiento de quadrotores [28], trazado y seguimiento de trayectorias [4], visión artificial [2], entre otras áreas como la generación de mapas, etc. Estos nuevos algoritmos requieren unidades de procesamiento más poderosas para lidiar con datos de múltiples sensores, cálculos

complejos, ejecutar multitareas o inclusive paralelismo para calcular todos los resultados previstos en tiempo para estos sistemas de muy rápida dinámica.

La mayoría de los prototipos existentes están basados en unidades de procesamiento como: microcontroladores [23], microprocesadores [24] o procesadores digitales de señales (DSP, por sus siglas en inglés) [17], los cuales a pesar de su alta frecuencia de oscilación, principalmente los últimos dos mencionados, están atados a ejecutar el programa de forma serial.

Actualmente la tecnología de procesadores con múltiples núcleos está generando revuelo entre la comunidad científica con las nuevas tarjetas de procesamiento cuyo procesador es un Texas Instruments OMAP o un Intel Atom, y un reloj que fácilmente supera 1 GHz. El problema principal de esta tecnología de múltiples núcleos radica en que la tarea de programación es más difícil en términos de sincronización de tareas y el acceso simultáneo a recursos compartidos como a la memoria o a los periféricos.

Por otro lado, existen otras soluciones orientadas a la descripción de hardware en las cuales el diseñador puede virtualmente crear cualquier clase de circuito o circuitos que pueden desempeñar tareas de forma concurrente en el mismo dispositivo. Uno de los más importantes dispositivos orientado a la descripción de hardware es el FPGA (acrónimo en inglés de Field Programmable Gate Array) el cual tiene las siguientes ventajas:

- Consumo recursos en base al diseño propuesto
- Múltiples circuitos concurrentes en el mismo hardware
- Flexibilidad en tamaño de buses de datos o control
- Flexibilidad de tipo de datos
- Alto desempeño en relación procesamiento-tiempo

Diferentes propuestas han sido presentadas alrededor de este dispositivo para realizar el control de un quadrotor, una de ellas consiste en la creación de un procesador suave embebido en un FPGA emulando a un Intel xscale chip-PXA225 y ejecutando un sistema operativo embebido de Linux. El FPGA utilizado para esta trabajo fue un Xilinx XCS30XL-4VQ100 y gran parte del esfuerzo de este trabajo reside en el diseño de la aplicación de un controlador del autopilotaje [12].

Una propuesta más aproximada a esta tesis es la implementación del sistema de control implementado para un vehículo aéreo no tripulado tipo quadrotor, en el que se usa un FPGA Xilinx Spartan-3 dentro de una tarjeta producida por Digilent. Se toma ventaja de los periféricos de la tarjeta, ya que está provee físicamente de un puerto serial usado para comunicar al FPGA con la central inercial Microstrain 3DM-GX1, así como switches y displays que facilitan acceder a la opciones y estado del sistema. La tarjeta incluye un oscilador a 50 MHz y por optimización de recursos los componentes diseñados fueron minimizados al máximo para expandir la configuración. Este trabajo consta de una descripción somera de los componentes descritos en hardware. La figura 2.4 muestra a diagrama de bloques el diseño propuesto para el FPGA Xilinx [31].

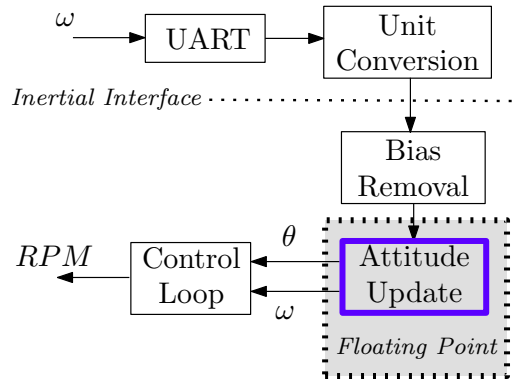


Figura 2.4: Diagrama de bloques del diseño del controlador embebido en FPGA

2.4 Resumen

En este capítulo se exponen los principales artículos que sirvieron como guía para llevar a cabo esta tesis y se da una breve explicación de su principal aportación. Se dividieron estos trabajos a fin de determinar su aportación en los diferentes capítulos mostrados en esta tesis.

Esta investigación fue importante para el desarrollo de este trabajo principalmente para tener mejor entendimiento del prototipo físico y de esta manera proponer un controlador e implementarlo en lenguaje de descripción de hardware.

Capítulo 3

Prototipo

Actualmente los Vehículos Aéreos no Tripulados (UAV, por sus siglas en inglés) han tomado gran auge en diferentes instituciones de investigación y de índole comercial, principalmente por la necesidad de satisfacer requerimientos específicos que van desde pruebas de nuevos materiales estructurales de bajo peso y alta resistencia, nuevas leyes de control y de navegación o inclusive la generación de ejércitos de robots móviles aéreos con fines de seguridad para el resguardo de territorios o búsqueda de sobrevivientes en zonas de desastre.

Debido a que dichas aplicaciones son muy particulares, la cantidad de prototipos desarrollados es vasta, desde los más comunes que son replicas a escala de las aeronaves comerciales tripuladas como aviones y helicópteros, así como vehículos híbridos que obedecen diseños muy característicos. Dentro de esta última categoría residen los vehículos aéreos multirotores, los cuales toman el principio básico de un helicóptero tradicional, despegue y aterrizaje vertical (VTOL, por sus siglas en inglés), pero con una mayor cantidad de rotores. Esto es con la finalidad de:

- Aumentar la capacidad de carga
- Simplificar del diseño mecánico
- Tener redundancia en presencia de fallas

Al aumentar la cantidad de rotores, se asume que la fuerza de empuje y la capacidad de carga aumenta, pero la relación consumo de energía por unidad de tiempo se puede ver comprometida. Un helicóptero tradicional es capaz de generar su movimiento debido a que su rotor principal es de paso variable, en el caso de los multirotores, las hélices están fijas al eje de rotación del rotor. Por esto, su mantenimiento y su control son más sencillos. El hecho de contar con múltiples rotores (más de cuatro) garantiza que cuando alguno de ellos falle, su controlador puede hacer uso de los rotores restantes y continuar con el vuelo previsto hasta su aterrizaje seguro. Es por esto que diferentes UAV multirotor han sido diseñados, algunos ejemplos son: el trirotor, el quadrotor, el hexarotor y el octorotor. Virtualmente no existe una cantidad límite de rotores establecida, pero factores adicionales como la comunicación y el control de estos puede complicarse.

Para efectos de esta tesis se construyó el prototipo de un quadrotor basado en diferentes artículos de investigaciones científicas relacionados con aeronaves multirotor. Tareas como diseño mecánico y eléctrico, dispositivos electrónicos y protocolos de comunicación empleados, así como su integración son cubiertas en este capítulo.

3.1 Diseño mecánico

El diseño estructural del quadrotor está basado en la decisión de usar un tipo de hélice en específico, la Mikrokopter EPP1045 [20], cuyas especificaciones son: 10 pulgadas de largo y una pendiente de 4.5 pulgadas por lado. El motivo de esta decisión es que dichas hélices en conjunción con los motores seleccionados, generan una fuerza de empuje individual de 820 gramos.

Teniendo la longitud de las hélices como restricción fundamental, se hizo uso de SolidWorks 2009 [26], el cual es un programa de diseño asistido por computadora para modelado mecánico. En dicho software se diseñaron dos piezas, los brazos y los marcos, los cuales pueden ser vistos en la figura 3.1. Los planos mecánicos se anexan en el medio que acompaña esta tesis. Terminado el diseño de las piezas, virtualmente se ensamblaron dentro del software para asegurar su alineación y evitar cualquier clase de interferencia.

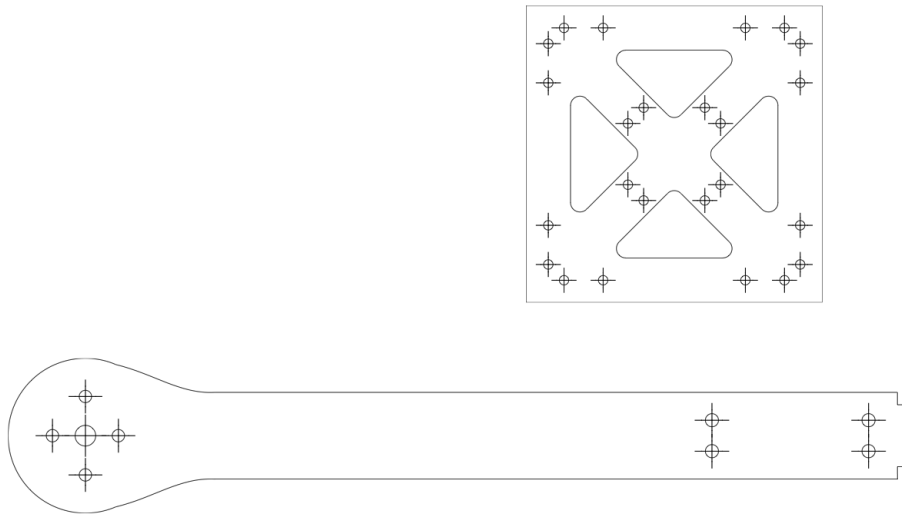


Figura 3.1: Diseño de piezas en SolidWorks 2009

El material seleccionado para dar forma a los brazos fue una placa compuesta por un núcleo de balsa de un cuarto de pulgada y capas delgadas en la parte inferior y superior de fibra de carbono, la conjunción de ambos materiales sirve para darle dureza y ligereza a la estructura, así como para absorber los impactos de posibles caídas y las vibraciones que inducen los rotores a todo el prototipo en sí. Por otro lado, para los marcos estructurales

se escogió aluminio calibre 18, debido a sus propiedades de dureza y ligereza. Estas piezas fueron manufacturadas a través de proceso de hidrocorte debido al tamaño de los barrenos y a la dificultad de cortar fibra de carbono con métodos tradicionales como Control Numérico por Computadora (CNC).

Los dispositivos electrónicos empleados están soldados sobre placas fenólicas perforadas y el ensamble de todo el prototipo se realiza con tornillos de plástico, creando así un sistema modular con el objetivo de que en caso de algún problema técnico pueda ser reparado rápidamente. El peso total de la aeronave es de 1.144 kg y mide 50 x 50 cm. La figura 3.2 muestra un imagen del quadrotor construido.

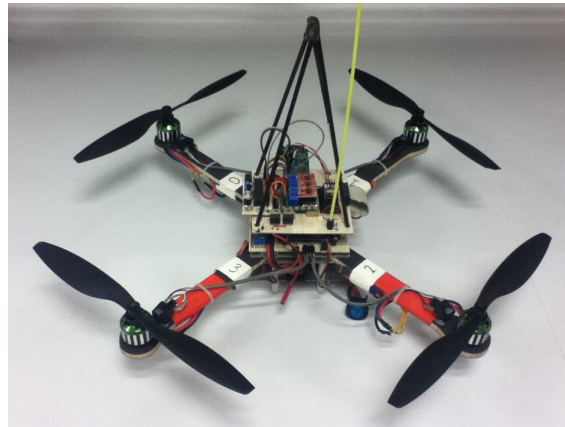


Figura 3.2: Prototipo de quadrotor basado en FPGA

3.2 Diseño eléctrico

El prototipo fue diseñado con componentes económicos y accesibles en cualquier tienda de electrónicos y de aeromodelismo. La siguiente lista muestra los componentes más importantes del sistema:

- Tarjeta de desarrollo DallasLogic CMCS002-2M con Altera Cyclone III EPC325 FPGA [7]
- Central inercial analógica Sparkfun SEN-10010 [27]
- Controlador de velocidad para motores sin escobillas YGE 30i [10]
- Motor sin escobillas ROXXY BL-Outrunner 2827-34 [21]
- Receptor de radio control ACT DSL4-Top [11]
- Módulo WiFi-UART Serialio WiSnap-M1 [30]

El diseño eléctrico original concebía que el sistema iba a ser alimentado por una sola batería de polímero de litio (LiPo, por sus siglas en inglés) de 11 volts y con una capacidad de 2100

mAh. Los dispositivos electrónicos empleados trabajan con un voltaje de 5 y 3.3 volts, razón por la cual se utilizaron convertidores de DC a DC para decrementar el voltaje suministrado por la batería con alta eficiencia. Al realizar las pruebas sobre el plano de tierra se notó que los motores inducían ruido eléctrico, afectando con esto la medición de la central inercial y por consiguiente la obtención valores erróneos principalmente de los giroscopios.

Por esta razón, se decidió aislar ópticamente al circuito analógico del resto del circuito digital a través de un optoacoplador, lo cual ayudó a su correcta estabilización. Para esto fue necesario agregar una nueva batería de 7.4 volts dedicada a la central inercial así como para su filtrado y su discretización. La figura 3.3 muestra el diagrama a bloques del sistema eléctrico del quadrotor.

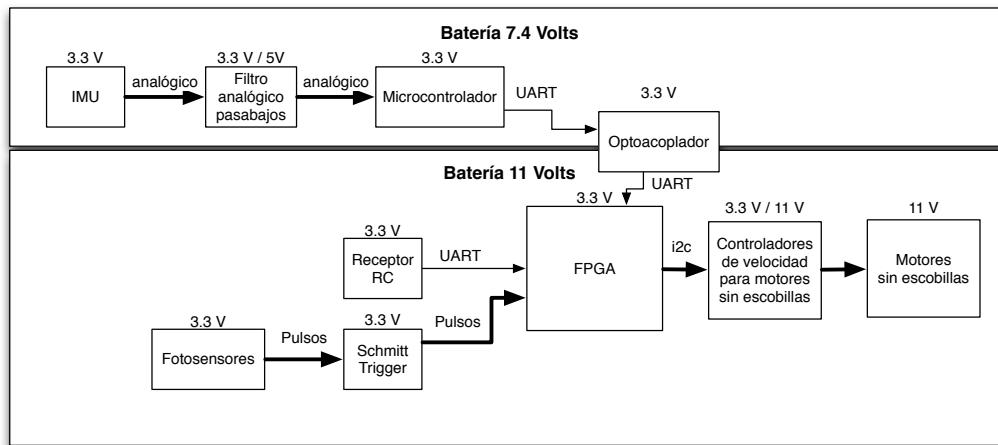


Figura 3.3: Diagrama de bloques del sistema eléctrico del quadrotor

Central inercial

El circuito de medición de la central inercial consiste en la tarjeta Sparkfun SEN-10010, la cual se muestra en la figura 3.4. Posee un sensor giroscópico y un sensor aceleración, ambos triaxiales y con salidas analógicas. Un giroscopio es un dispositivo microelectromecánicos (MEMS) que convierte la velocidad angular de algún eje en específico (x , y o z) en una variación de voltaje a partir de una referencia en dependencia del sentido de giro. Por otro lado, un acelerómetro también es un dispositivo MEMS que mide aceleraciones. Esto no es necesariamente la misma que la aceleración de coordenadas (cambio de la velocidad del dispositivo en el espacio), sino que es el tipo de aceleración asociadas con el fenómeno de peso experimentada por una masa de prueba que se encuentra en el marco de referencia del dispositivo. En base a este fenómeno de peso, un acelerómetro puede ser usado para la medir la inclinación u orientación del marco de referencia en el que esta situado, de igual forma la salida es una variación de voltaje a partir de una referencia en dependencia del sentido de giro.

Las salidas analógicas de los sensores son filtradas activamente a través de un circuito pasa bajos hecho con amplificadores operacionales con frecuencia de corte de 12 Hz. El filtro

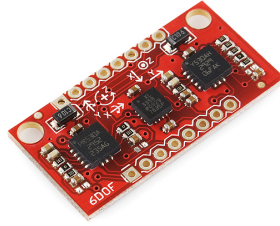


Figura 3.4: Central inercial Sparkfun SEN-10010

elegido a implementar es de tipo Butterworth de segundo orden con estructura Sallen-Key. La figura 3.5 muestra dicha configuración.

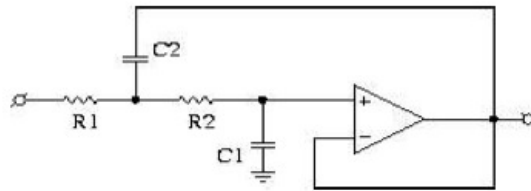


Figura 3.5: Filtro activo pasa bajos Sallen Key

Por tablas, los coeficientes del filtro con ganancia unitaria ($A_0=1$) son $a=1.4142$ y $b=1$. Las ecuaciones que describen el valor de los capacitores y resistencias se muestran a continuación.

Considerando que $C_1=470$ nF

$$C_2 \geq C_1 \frac{4b}{a^2} = 940nF \approx 1\mu F \quad (3.1)$$

$$R_1 = \frac{aC_2 + \sqrt{a^2C_2^2 - 4bC_1C_2}}{4\pi f_c C_1 C_2} = 24841\Omega \approx 27K\Omega \quad (3.2)$$

y

$$R_2 = \frac{aC_2 - \sqrt{a^2C_2^2 - 4bC_1C_2}}{4\pi f_c C_1 C_2} = 15066\Omega \approx 15K\Omega \quad (3.3)$$

Las señales filtradas son conectadas a un microcontrolador Atmel ATMega16l, el cual fue programado para realizar dos tareas específicas, la conversión analógico-digital de las señales filtradas y el envío de dicha conversión en un formato establecido a través de un protocolo serial asíncrono (UART) con un tasa de transferencia de 115,200 bps. Este proceso se realiza con un frecuencia de actualización de 300 Hz. La figura 3.6 muestra el microcontrolador empleado para esta operación.



Figura 3.6: Microcontrolador Atmel ATmega161

Por último, la señal de transmisión UART del microcontrolador es conectada a un optoacoplador 6N137, el cual a su vez lo replica hacia el FPGA. El optoacoplador de alta velocidad de conmutación, se encarga del envío de datos entre los dos circuitos sin alguna referencia a tierra en común y evitando así la contaminación por ruido digital en el circuito analógico. La figura 3.7 muestra el circuito de aislamiento óptico para la transmisión UART.

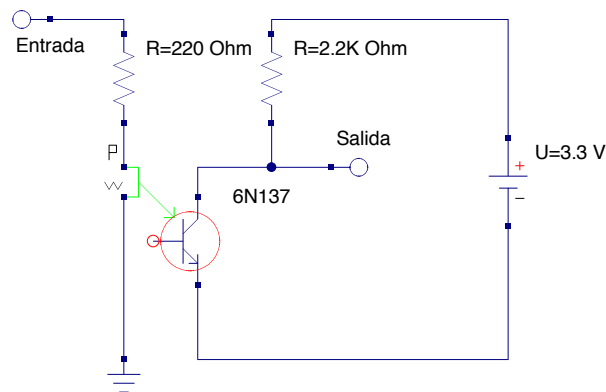


Figura 3.7: Circuito optoacoplado para la transmisión UART

Es importante recalcar que la central inercial está posicionada a 45° de los brazos que conforman la estructura, esta configuración es conocida como X. La figura 3.8 hace referencia a esta configuración.

Transmisión y recepción de radio control

Para el sistema de radio control se utilizó un transmisor Graupner MC-12 [16] y un receptor ACT DSL4-Top en la banda de los 35 MHz. El DSL4-Top es un dispositivo que convierte la señal recibida del transmisor en protocolo UART. De esta forma la lectura e interpretación de los datos es más confiable, ya que al final de la recepción solo hay que comparar con el checksum calculado con el checksum enviado, si el resultado es igual, entonces se actualizan los datos, sino son descartados. La ventaja principal de este receptor contra los demás, es que estos últimos funcionan a través del protocolo de modulación por posición de pulso

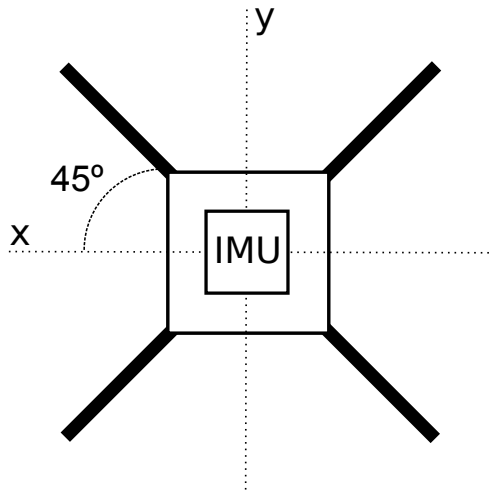


Figura 3.8: Quadrotor con configuración X

(PPM, por sus siglas en inglés), cuya principal desventaja es el uso de contadores para su decodificación que pueden variar su frecuencia en función del cristal empleado para su excitación, temperatura, cantidad de instrucciones del programa en ejecución, etc. La figura 3.9 muestra la imagen del receptor ACT DSL4-Top.



Figura 3.9: Receptor FM ACT DSL4-Top

La señal de recepción UART del DSL4-Top es directamente conectada al FPGA.

Medición de velocidad de los rotores

El circuito para la medición de velocidad de los rotores consta principalmente de un sensor infrarrojo Honeywell HOA1405 [13] y una etiqueta con un patrón diseñado de 32 franjas verticales alternadas (impresas y no impresas) en un material tipo metálico. La figura 3.10 muestra el diseño impreso sobre la etiqueta.

Esta etiqueta fue adherida a la carcasa del rotor, entonces por cada revolución de éste, 16 pulsos son generados por el sensor infrarrojo. La salida del sensor es filtrada a través de



Figura 3.10: Patrón de franjas verticales

un inversor Schmitt Trigger 74HC14 [5]. Este dispositivo usa la histéresis para prevenir el ruido que podría sobreponer a la señal original y que causaría falsos cambios de estado si los niveles de referencia y entrada son parecidos. Este circuito es individual para cada rotor y la señal filtrada es conectada al FPGA, el cual cuenta la cantidad de pulsos en un periodo de tiempo definido, obteniendo así una aproximación a la velocidad real de giro de rotor (ver capítulo 6). La figura 3.11 muestra el circuito diseñado para medir la velocidad de los rotores.

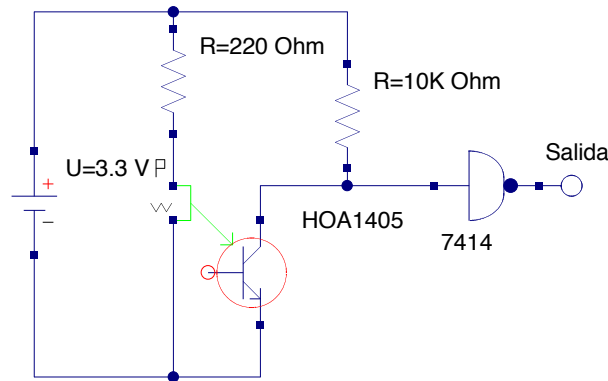


Figura 3.11: Circuito para la medición de velocidad de los rotores

Tarjeta de desarrollo FPGA

El módulo CMCS002-2M permite la implementación de funciones lógicas generales en una tarjeta de desarrollo de tamaño compacto. El modulo provee los siguientes circuitos y características necesarios para conformar un controlador basado en FPGA:

- FPGA Altera EP3C25-C8
- Memoria SRAM 512K X 8 (4Mbit)
- Memoria flash serial EPCS16 (16Mbit)
- 109 pines de entrada y salida
- Puerto periférico USB 2.0
- Reloj oscilador de 25 MHz

La figura 3.12 muestra la imagen de la tarjeta de desarrollo CMCS002-2M.



Figura 3.12: Tarjeta de desarrollo CMCS002-2M

Rotores

El circuito de los rotores está comprendido por dos dispositivos, un controlador de velocidad para motores sin escobillas YGE 30i y un motor sin escobillas ROXXY BL-Outrunner 2827-34. El controlador de velocidad recibe datos del FPGA a través del protocolo I²C, primordialmente el dato de velocidad al que se desea que el rotor gire. Es entonces cuando el controlador de velocidad modifica la forma del pulso contraelectromotriz (EMF) y así ajusta la corriente de salida, lo cual se ve reflejado en la velocidad del rotor. La figura 3.13 muestra la imagen del controlador de velocidad y del motor sin escobillas.

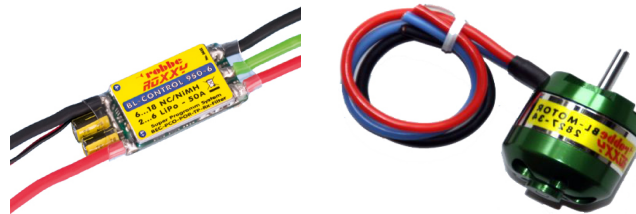


Figura 3.13: Controlador de velocidad y motor sin escobillas

3.3 Protocolos de comunicación

Un protocolo de comunicación se define con un conjunto de normas estipuladas para la representación, señalización, autenticación y detección de errores necesario para enviar información a través de un canal de comunicación. En el caso de la integración de múltiples dispositivos, los protocolos de comunicación seriales juegan un papel muy importante, ya que con pocos cables o hilos, se pueden transmitir datos a alta velocidad incrementando

el desempeño total del sistema. Los dispositivos seleccionados para integrar al quadrotor se comunican a través de dos protocolos de comunicación seriales, el protocolo UART y el protocolo I²C.

Protocolo UART

El protocolo UART (acrónimo en inglés de Universal Asynchronous Receiver Transmitter) es un protocolo de comunicación serial asíncrono. En primera instancia no existe un sistema maestro-esclavo, así como tampoco la transmisión de una señal de reloj que acompañe a los datos transmitidos, es por esto que ambos dispositivos deben tener exactamente la misma configuración de todos sus parámetros para garantizar la comunicación con el menor porcentaje de error posible. Los cantidad de cables requeridos para la comunicación son tres: uno para la transmisión (TXD), uno para la recepción (RXD) y la tierra común (GND) entre los dispositivos a comunicar. Este protocolo cae dentro de la categoría de Full-Duplex (envío y recepción simultánea). La figura 3.14 muestra el circuito de interconexión de dos dispositivos que se comunican a través del protocolo UART.



Figura 3.14: Circuito eléctrico del protocolo UART

Dentro de los parámetros a considerar durante su configuración es la velocidad de transferencia, al ser un estándar, dichas velocidades ya se encuentran predefinidas en múltiplos de 1,200 bps. Otro valor que se tiene que tomar en cuenta es la cantidad de bits que se van a transmitir por cada trama a enviar, el número de bits por omisión es de 8, pero existe la opción de incrementarlo a 9. Si se desea aumentar el grado confiabilidad es posible accionar la opción de paridad, en la cual el sistema hace el calculo de este bit en base a los valores de bits enviados, esta opción por defecto no es usada. Por último, se debe configurar cuantos bits de paro tendrá la trama, este valor es importante para poner en sincronía algunos sistemas con osciladores que generan divisores de frecuencia con considerable porcentaje de error, el valor por omisión es 1, pero puede ser incrementado a 2. La figura 3.15 muestra la trama generada, cabe señalar que la línea siempre se encuentra en alto lógico, en el momento que se inicia la transmisión, se envía un bajo lógico para continuar con el dato a enviar y por último el o los bits de paro.

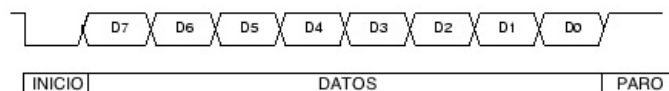


Figura 3.15: Protocolo de comunicación UART

Protocolo I²C

El protocolo I²C (acrónimo en inglés de Inter Integrated Circuit) es un protocolo de comunicación serial síncrono. Para este protocolo en específico existe una relación maestro-esclavo, en la cual el maestro genera y envía una señal de reloj la cual sirve para transaccionar el sentido de la comunicación. Al igual que para el protocolo UART, el I²C requiere de tres cables: uno para la señal de reloj (SCL), uno para los datos (SDA) y la tierra común entre los dispositivos (GND).

Al tener un esquema de maestro-esclavo da la posibilidad de tener hasta 127 dispositivos bajo el mismo bus de datos. En base a lo anterior cada dispositivo tiene una dirección lógica la cual debe ser única. Es importante hacer notar que el protocolo trabaja con las señales de reloj y de datos en modo de pull-up (conectado dichas señales a través de una resistencia a la línea de voltaje) con la intención de proteger a los dispositivos en caso de corto circuito por colisión de datos, es decir, si un dispositivo desea enviar un bajo lógico pues envía a través del medio el valor de la tierra común, pero si se desea enviar en un alto lógico, entonces simplemente el dispositivo pone su salida en alta impedancia. Debido a esto, la comunicación de datos es bidireccional. La velocidad de la señal no está definida a un rango de valores específicos por lo que el dispositivo maestro tiene la facultad de controlarla. Según el estándar, la velocidad máxima es de 4 MHz. La figura 3.16 muestra el circuito eléctrico del protocolo I²C.

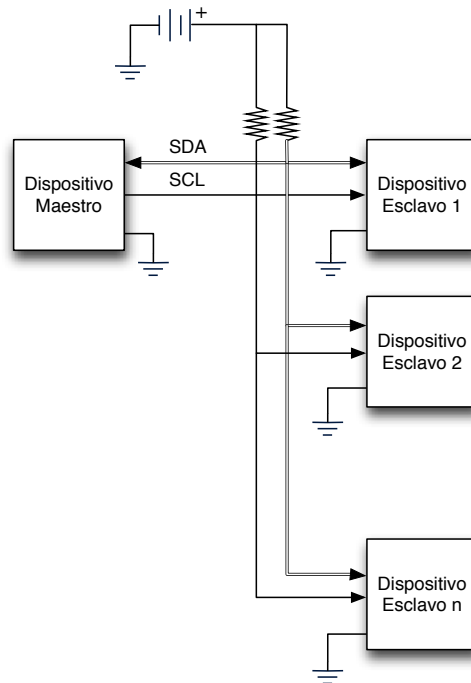


Figura 3.16: Circuito eléctrico del protocolo I²C

El protocolo consiste en que en primera instancia el maestro pone en el bus de datos la

Capítulo 4

Modelado matemático

El modelado matemático de un sistema se puede describir como el conjunto de ecuaciones que denotan la dinámica propia de dicho sistema de forma precisa. Un mismo sistema puede ser representado por diferentes modelos matemáticos que varían principalmente en su nivel de detalle comparado con el sistema real y de los fundamentos físicos empleados para su obtención.

Las ecuaciones de movimiento que describen el comportamiento del quadrotor son básicamente aquellas que se utilizan para definir cualquier sólido en el espacio con seis grados de libertad, y en particular, con cuatro entradas de control sujeto a perturbaciones externas. Estas ecuaciones pueden obtenerse de diversas maneras, pero en base a los textos científicos publicados existen dos métodos que son los más utilizados, el primero nace a partir de las ecuaciones de movimiento de Newton, cuya ventaja principal radica en la posibilidad de conocer las fuerzas generalizadas que se requieren aplicar al vehículo para lograr el movimiento deseado de las coordenadas generalizadas, o bien, a través de las ecuaciones de Euler-Lagrange cuyo enfoque permite describir de una manera más detallada la dinámica de las coordenadas generalizadas.

Dado que uno de los objetivos particulares busca la estabilización de las coordenadas angulares, se utiliza el enfoque de Euler-Lagrange propuesto en [19] como base para el desarrollo de esta tesis.

4.1 Características del quadrotor

El quadrotor, a diferencia de un helicóptero clásico, se controla variando únicamente la velocidad angular de cada uno de sus cuatro rotores. La fuerza producida f_i por el rotor i es proporcional al cuadrado de la velocidad angular del rotor w_i , esto es,

$$f_i = kw_i^2 \tag{4.1}$$

donde $k > 0$ es una constante.

En base a su diseño, un quadrotor se compone básicamente de cuatro brazos en una configuración tipo X y cada brazo sostiene un rotor. Un rotor se conforma de un motor de eléctrico

sin escobillas, una hélice y un cople que une el eje del motor con la hélice. Como se observa en la figura 4.1, los motores 1 y 3 giran en sentido horario, mientras que los motores 2 y 4 giran en sentido antihorario, por ende los efectos giroscópicos y los momentos aerodinámicos tienden a cancelarse en vuelo estacionario. Cabe señalar que el sentido de giro de los rotores es fijo, por esta razón la fuerza de empuje producida siempre será positiva.

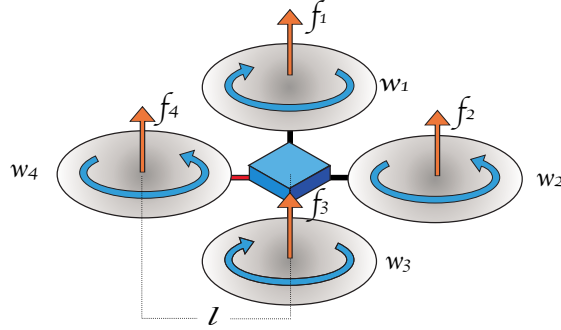


Figura 4.1: Esquema básico de un quadrotor

Si la velocidad total de los rotores es incrementada o decrementada impactará sobre movimiento vertical de la aeronave, dado que la entrada principal de control es la suma de cada fuerza producida por cada rotor, esto es:

$$u = \sum_{i=1}^4 f_i \quad (4.2)$$

Debido a que en el prototipo diseñado, la central inercial está situada a 45° de los brazos que sostienen los rotores (ver figura 3.8), la variación de la velocidad de todos estos es necesaria para lograr el movimiento del quadrotor.

El momento de *pitch* positivo se genera cuando la fuerza generada por los rotores 1 y 2 aumenta, mientras que la de los rotores 3 y 4 disminuye. Si en los rotores 3 y 4 la fuerza aumenta y en la de los rotores 1 y 2 disminuye entonces el momento de *pitch* es negativo. El momento de *roll* positivo ocurre cuando la fuerza de los rotores 2 y 3 aumenta y la de los rotores 1 y 4 disminuye. Si se desea el momento de *roll* negativo, ocurre el caso opuesto. El momento de *yaw* se produce a través de la suma de torques generados por los motores τ_{M_i} . La figura 4.2 hace referencia a estos momentos de forma gráfica.

Las ecuaciones que describen dichos momentos están respectivamente dadas por:

$$\tau_\theta = l(f_1 + f_2 - f_3 - f_4) \quad (4.3)$$

$$\tau_\phi = l(f_2 + f_3 - f_1 - f_4) \quad (4.4)$$

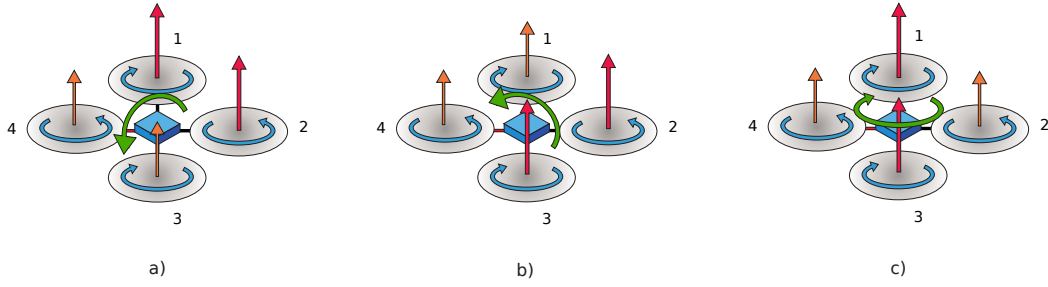


Figura 4.2: a) Movimiento en pitch. b) Movimiento en roll. c) Movimiento en yaw.

$$\tau_{\psi} = \sum_{i=1}^4 \tau_{M_i} \quad (4.5)$$

donde l es la longitud del brazo del quadrotor, la cual equivale a la distancia del centro de masa de la aeronave al centro de rotor.

El torque del rotor es opuesto a la fricción aerodinámica τ_d , tal que:

$$I_r \dot{w}_i = \tau_{M_i} - \tau_d \quad (4.6)$$

donde I_r es el momento de inercia del rotor sobre su eje de rotación.

La fricción aerodinámica se define como:

$$\tau_d = \frac{1}{2} \rho A v^2 \quad (4.7)$$

donde ρ es la densidad del aire, A es el área frontal de la hélice y v es la velocidad relativa respecto al aire.

La velocidad angular w es igual a la velocidad lineal v dividida por el radio de rotación, por lo tanto: $w = v/r$. Reescribiendo,

$$\tau_d = k_d w^2 \quad (4.8)$$

donde $k_d > 0$ es una constante que relaciona la densidad de aire, el radio y el área frontal de la hélice.

Si se considera que el quadrotor se desempeñará en vuelo estacionario, w es una constante, entonces se tiene que:

$$\tau_{M_i} = \tau_d \quad (4.9)$$

Estos momentos se deben realizar manteniendo la fuerza u constante.

4.2 Modelado dinámico

El modelo del quadrotor que a continuación se presenta se obtiene a partir de la aproximación de Euler-Lagrange y se representa como un cuerpo sólido que interactúa tridimensionalmente, está sujeto a una fuerza y tres momentos. La figura 4.3 representa el diagrama de cuerpo libre del quadrotor.

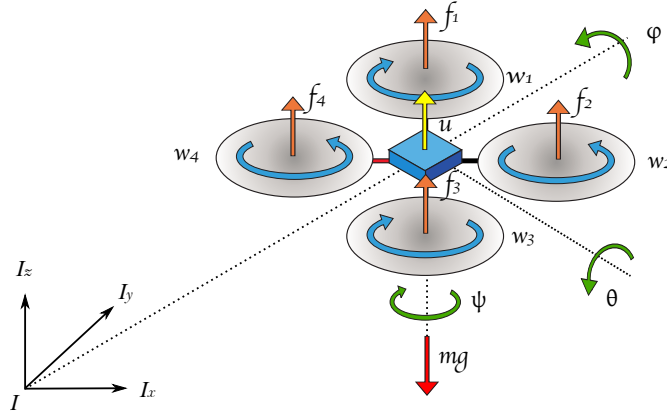


Figura 4.3: Diagrama de cuerpo libre del quadrotor

Las coordenadas generalizadas de la aeronave son:

$$q = (x, y, z, \psi, \theta, \phi) \in \mathbb{R}^6 \quad (4.10)$$

donde $\xi = (x, y, z) \in \mathbb{R}^3$ marca la posición del centro de masa del helicóptero y $\eta = (\psi, \theta, \phi) \in \mathbb{R}^3$ son los ángulos de Euler (*yaw*, *pitch* y *roll* respectivamente) que representan la orientación de la aeronave, ambos (ξ y η) respecto al marco de referencia inercial ($\mathcal{I} = \mathcal{I}_x, \mathcal{I}_y, \mathcal{I}_z$). Bajo esta notación, el modelo se puede dividir en coordenadas traslacionales y rotacionales.

La energía cinética de un sólido rígido en rotación está definida por la siguiente ecuación:

$$T_{tot} = T_{trans} + T_{rot} = \frac{1}{2}mv^T \cdot v + \frac{1}{2}w^T \cdot \mathbf{I}w \quad (4.11)$$

donde m es la masa e \mathbf{I} es la matriz inercial del sólido rígido.

Por lo tanto, aplicando la expresión matemática anterior, la energía cinética traslacional de la aeronave puede ser plasmada de la siguiente manera:

$$T_{trans} = \frac{1}{2}m\dot{\xi}^T \dot{\xi} \quad (4.12)$$

donde m es la masa de la aeronave.

El vector de la velocidad angular w respecto al marco de referencia en el cuerpo de la aeronave se relaciona con las velocidades generalizadas $\dot{\eta}$ (en la región donde los ángulos de Euler son válidos) utilizando una relación estándar cinemática.

$$\dot{\eta} = W_{\eta}^{-1}w \quad (4.13)$$

donde

$$W_{\eta} = \begin{bmatrix} -\text{sen}\theta & 0 & 1 \\ \text{cos}\theta\text{sen}\psi & \text{cos}\psi & 0 \\ \text{cos}\theta\text{cos}\psi & -\text{sin}\psi & 0 \end{bmatrix} \quad (4.14)$$

Definiendo,

$$\mathbb{J} = W_{\eta}^T \mathbf{I} W_{\eta} \quad (4.15)$$

tal que la energía cinética rotacional es

$$T_{rot} = \frac{1}{2} \dot{\eta}^T \mathbb{J} \dot{\eta} \quad (4.16)$$

donde \mathbb{J} actúa como matriz inercial para la energía cinética total rotacional de la aeronave, expresada en términos de las coordenadas generalizadas η .

La única energía potencial que debe de ser considerada es el peso descrito por

$$V = mgz \quad (4.17)$$

donde m es la masa de la aeronave, g es la fuerza de gravedad y z es la altura de la aeronave.

El Lagrangiano (L) de un sistema mecánico es una función que resume las dinámicas del sistema. Está definido como la energía cinética T del sistema menos su energía potencial V .

$$L = T - V \quad (4.18)$$

Por lo tanto, el Lagrangiano del sistema es

$$L(q, \dot{q}) = T_{trans} + T_{rot} - V = \frac{1}{2} m \dot{\xi}^T \dot{\xi} + \frac{1}{2} \dot{\eta}^T \mathbb{J} \dot{\eta} - mgz \quad (4.19)$$

El modelo completo de la dinámicas de la aeronave es obtenido a través ecuación de Euler-Lagrange:

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} = \begin{bmatrix} F_{\xi} \\ \tau \end{bmatrix} \quad (4.20)$$

donde F_{ξ} es la fuerza traslacional y τ representa los momentos de la aeronave. Se ignoran las fuerzas inherentes al cuerpo debido a que son mucho más pequeñas en magnitud que las entradas principales de control u y τ .

De la figura 4.3, se tiene

$$\hat{F} = \begin{bmatrix} 0 \\ 0 \\ u \end{bmatrix} \quad (4.21)$$

donde

$$u = \sum_{i=1}^4 f_i \quad (4.22)$$

y

$$f_i = k_i w_i^2, \quad i = 1, \dots, 4 \quad (4.23)$$

donde k_i es una constante y w_i es la velocidad angular del motor i , entonces $F_\xi = R\hat{F}$ donde R es la matriz de transformación que representa la orientación de la aeronave.

$$R = \begin{bmatrix} \cos\theta\cos\psi & \sin\psi\sin\theta & -\sin\theta \\ \cos\psi\sin\theta\sin\phi - \sin\psi\cos\phi & \sin\psi\sin\theta\sin\phi + \cos\psi\cos\phi & \cos\theta\sin\phi \\ \cos\psi\sin\theta\cos\phi + \sin\psi\sin\phi & \sin\psi\sin\theta\cos\phi - \cos\psi\sin\phi & \cos\theta\cos\phi \end{bmatrix} \quad (4.24)$$

Para esta tesis, la representación de la matriz de rotación R esta basada en el siguiente orden de rotación: la primera rotación sobre el ángulo ψ alrededor del eje z , la segunda rotación sobre el ángulo θ alrededor del nuevo eje y y la tercera rotación sobre el ángulo ϕ alrededor del nuevo eje x .

Los momentos generalizados en la variable η son

$$\tau = \begin{bmatrix} \tau_\psi \\ \tau_\theta \\ \tau_\phi \end{bmatrix} \quad (4.25)$$

donde

$$\tau_\psi = \sum_{i=1}^4 \tau_{M_i} \quad (4.26)$$

$$\tau_\theta = l(f_1 + f_2 - f_3 - f_4) \quad (4.27)$$

$$\tau_\phi = l(f_2 + f_3 - f_1 - f_4) \quad (4.28)$$

Reescribiendo,

$$\tau_\psi = k_d(w_1^2 + w_3^2 - w_2^2 - w_4^2) \quad (4.29)$$

$$\tau_\theta = kl(w_1^2 + w_2^2 - w_3^2 - w_4^2) \quad (4.30)$$

$$\tau_\phi = kl(w_2^2 + w_3^2 - w_1^2 - w_4^2) \quad (4.31)$$

donde l es la distancia desde el motor hasta el centro de gravedad y τ_{M_i} es el torque producido por el motor.

Dado que el Lagrangiano no contiene términos cruzados en la ecuación de energía cinética, la ecuación de Euler-Lagrange puede ser separada en las dinámicas de posición, ξ

$$m\ddot{\xi} + mgE_z = F_\xi \quad (4.32)$$

$$m\ddot{\xi} + mgE_z = R\hat{F} \quad (4.33)$$

$$m \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} = \begin{bmatrix} -u\sin\theta \\ u\cos\theta\sin\phi \\ u\cos\theta\cos\phi \end{bmatrix} \quad (4.34)$$

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = m^{-1} \begin{bmatrix} -u\sin\theta \\ u\cos\theta\sin\phi \\ u\cos\theta\cos\phi - mg \end{bmatrix} \quad (4.35)$$

y las dinámicas de rotación, η .

$$\mathbb{J}\ddot{\eta} + \dot{\mathbb{J}}\dot{\eta} - \frac{1}{2} \frac{\partial}{\partial \eta} (\dot{\eta}^T \mathbb{J} \dot{\eta}) = \tau \quad (4.36)$$

$$\mathbb{J} \begin{bmatrix} \ddot{\psi} \\ \ddot{\theta} \\ \ddot{\phi} \end{bmatrix} + \dot{\mathbb{J}} \begin{bmatrix} \dot{\psi} \\ \dot{\theta} \\ \dot{\phi} \end{bmatrix} - \frac{1}{2} \frac{\partial}{\partial \eta} \left(\begin{bmatrix} \dot{\psi} & \dot{\theta} & \dot{\phi} \end{bmatrix} \mathbb{J} \begin{bmatrix} \dot{\psi} \\ \dot{\theta} \\ \dot{\phi} \end{bmatrix} \right) = \begin{bmatrix} \tau_\psi \\ \tau_\theta \\ \tau_\phi \end{bmatrix} \quad (4.37)$$

$$\begin{bmatrix} \ddot{\psi} \\ \ddot{\theta} \\ \ddot{\phi} \end{bmatrix} = \mathbb{J}^{-1} \left(\begin{bmatrix} \tau_\psi \\ \tau_\theta \\ \tau_\phi \end{bmatrix} - \dot{\mathbb{J}} \begin{bmatrix} \dot{\psi} \\ \dot{\theta} \\ \dot{\phi} \end{bmatrix} + \frac{1}{2} \frac{\partial}{\partial \eta} \left(\begin{bmatrix} \dot{\psi} & \dot{\theta} & \dot{\phi} \end{bmatrix} \mathbb{J} \begin{bmatrix} \dot{\psi} \\ \dot{\theta} \\ \dot{\phi} \end{bmatrix} \right) \right) \quad (4.38)$$

Por simplicidad se considera el cambio de variables

$$\begin{bmatrix} \tilde{\tau}_\psi \\ \tilde{\tau}_\theta \\ \tilde{\tau}_\phi \end{bmatrix} = \mathbb{J}^{-1} \left(\begin{bmatrix} \tau_\psi \\ \tau_\theta \\ \tau_\phi \end{bmatrix} - \dot{\mathbb{J}} \begin{bmatrix} \dot{\psi} \\ \dot{\theta} \\ \dot{\phi} \end{bmatrix} + \frac{1}{2} \frac{\partial}{\partial \eta} \left(\begin{bmatrix} \dot{\psi} & \dot{\theta} & \dot{\phi} \end{bmatrix} \mathbb{J} \begin{bmatrix} \dot{\psi} \\ \dot{\theta} \\ \dot{\phi} \end{bmatrix} \right) \right) \quad (4.39)$$

Finalmente, el modelo matemático de rotación se expresa de forma simplificada como

$$\begin{bmatrix} \ddot{\tilde{\varphi}} \\ \ddot{\tilde{\theta}} \\ \ddot{\tilde{\phi}} \end{bmatrix} = \begin{bmatrix} \tilde{\tau}_\psi \\ \tilde{\tau}_\theta \\ \tilde{\tau}_\phi \end{bmatrix} \quad (4.40)$$

Las ecuaciones 4.35 y 4.40 describen el comportamiento dinámico no lineal del quadrotor.

4.3 Resumen

En este capítulo se detallan las consideraciones, la metodología y el desarrollo del modelo matemático a partir de la aproximación de Euler-Langrange, el cual describe al sistema como una diferencia de energías y permite mostrar de una manera más detallada la dinámica de las coordenadas generalizadas.

El modelo resultante es no lineal, pero se toman algunas consideraciones para su simplificación como la suposición de que no existen efectos giroscópicos ya que la aeronave se desempeña en vuelo suspendido o cuasisuspendido, lo cual disminuye notablemente la complejidad del sistema.

Capítulo 5

Estabilización

Se hace evidente que un vehículo aéreo tipo quadrotor presenta un modelo no lineal, inestable y complejo, dada su naturaleza e interacción con el entorno. Es por esto que dicho vehículo presenta dos lazos de control principales. El primero y fundamental es el control de orientación que persigue la estabilización de cualquier movimiento del vehículo. El segundo lazo de control considera al vehículo estabilizado como un sistema para desempeñar tareas específicas en conjunción con otros sensores, como son el seguimiento de trayectorias o localización y mapeos simultáneos (por sus siglas en inglés SLAM).

El control de orientación es el encargado de estabilizar al vehículo manteniendo la posición angular deseada y evitar que el vehículo se desestabilice en ausencia de las entradas de control enviadas por el piloto. El desempeño de este control está en función del nivel de ruido del sensado y el rechazo a perturbaciones del sistema.

El método de linealización aproximada es utilizado para proponer un controlador de orientación Proporcional-Derivativo (PD) lineal, a partir de un modelo dinámico simplificado alrededor del punto de equilibrio.

5.1 Linealización

El modelo dinámico del quadrotor descrito en la sección anterior es un modelo no lineal con acoplamientos en sus estados. Es importante resaltar que si se desea estabilizar a la aeronave, el modelo descrito puede ser linealizado en la región quadrotor presente condición de vuelo suspendido (hover, en inglés) o cuasisuspendido (pequeños desplazamientos laterales derivados por pequeños ángulos en la inclinación del quadrotor). En este caso, el modelo será linealizado alrededor su punto de equilibrio.

Por facilidad de expresión y manipulación, es conveniente representar las ecuaciones dinámicas 4.35 y 4.40 en forma matricial. Se define el vector de estado como

$$\bar{x} = (x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10}, x_{11}, x_{12}) = (x, \dot{x}, y, \dot{y}, z, \dot{z}, \psi, \dot{\psi}, \theta, \dot{\theta}, \phi, \dot{\phi}) \quad (5.1)$$

y el vector de entrada como

$$\bar{u} = (u_1, u_2, u_3, u_4) = (u - mg, \tilde{\tau}_\psi, \tilde{\tau}_\theta, \tilde{\tau}_\phi) \quad (5.2)$$

En base al vector de estado y al vector de entrada, las ecuaciones dinámicas se pueden reexpresar en forma de variables de estado como,

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \\ \dot{x}_6 \\ \dot{x}_7 \\ \dot{x}_8 \\ \dot{x}_9 \\ \dot{x}_{10} \\ \dot{x}_{11} \\ \dot{x}_{12} \end{bmatrix} = \begin{bmatrix} x_2 \\ \frac{-u_1 \sin x_9}{m} \\ x_4 \\ \frac{u_1 \cos x_9 \sin x_{11}}{m} \\ x_6 \\ \frac{u_1 \cos x_9 \cos x_{11} - mg}{m} \\ x_8 \\ u_2 \\ x_{10} \\ u_3 \\ x_{12} \\ u_4 \end{bmatrix} \quad (5.3)$$

Sea $\tilde{x} = 0$ un punto de equilibrio para la condición de vuelo suspendido, dicha condición ocurre cuando la fuerza de empuje total de los rotores es ligeramente mayor al peso de la aeronave, manteniendo la aeronave en sustentación. La linealización aproximada hace uso de la matriz jacobiana.

La matriz jacobiana $Df(x)$ en un punto \tilde{x} de la función vectorial $f : U \subseteq \mathbb{R}^n \rightarrow \mathbb{R}^m$ con: $f(x_1, x_2, \dots, x_n) = (f_1(x_1, \dots, x_n), f_2(x_1, \dots, x_n), \dots, f_m(x_1, \dots, x_n))$, es la matriz $m \times n$ cuyos elementos d_{ij} son las derivadas parciales de las componentes f_i respecto de las variables x_j ($i = 1, \dots, m; j = 1, \dots, n$), evaluadas en el punto \tilde{x} [25].

$$Df(x) = \begin{bmatrix} \frac{\partial y_1}{\partial x_1} & \dots & \frac{\partial y_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial y_m}{\partial x_1} & \dots & \frac{\partial y_m}{\partial x_n} \end{bmatrix} \quad (5.4)$$

La linealización aproximada alrededor del punto de equilibrio es representada por el siguiente sistema lineal invariante en el tiempo a través de su representación de ecuaciones de estado

$$\dot{x} = A\bar{x} + B\bar{u} \quad (5.5)$$

donde

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -g & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & g & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ y \\ \dot{y} \\ z \\ \dot{z} \\ \psi \\ \dot{\psi} \\ \theta \\ \dot{\theta} \\ \phi \\ \dot{\phi} \end{bmatrix} \quad (5.6)$$

y

$$B = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \frac{1}{m} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & \frac{1}{I_{zz}} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{I_{yy}} & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{I_{xx}} \end{bmatrix} \begin{bmatrix} u - mg \\ \tau_{\psi} \\ \tau_{\theta} \\ \tau_{\phi} \end{bmatrix} \quad (5.7)$$

Las salidas del sistema están representadas por la siguiente ecuación de salida

$$y = C\bar{x} \quad (5.8)$$

donde

$$C = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} z \\ \psi \\ \theta \\ \phi \end{bmatrix} \quad (5.9)$$

En base a las ecuaciones obtenidas a través del procedimiento de linealización, se puede inferir que se eliminaron los acoplamientos entre los estados, además de considerar a cada uno de los grados de libertad como un sistema independiente de tipo una entrada una salida (SISO, por sus siglas en inglés) dentro de los límites en los cuales la linealización aproximada es válida.

Para que el sistema descrito por la ecuación de estados 5.5 sea de estado completamente controlable, es necesario y suficiente que la siguiente matriz de controlabilidad $S = [B \ AB \ A^2B \ \dots \ A^{11}B]$ sea de rango 12, es decir de rango completo. Esta condición ha sido verificada por lo que el sistema es controlable.

5.2 Control de orientación

El controlador necesario para estabilizar a un quadrotor es un controlador lineal derivativo (controlador D), este controlador responde ante cambios abruptos, debido a su alta sensibilidad, mejorando así la respuesta en estado transitorio.

Para obtener realmente un controlador de orientación es necesario añadir al controlador de estabilización, el término proporcional, obteniendo así un controlador proporcional derivativo (controlador PD).

Control proporcional derivativo

El controlador proporcional derivativo consiste en aplicar la suma de dos tipos de acciones de control, la acción proporcional y la acción derivativa [29].

La acción proporcional de control es proporcional al estado actual de error, de acuerdo a la ecuación 5.10.

$$u(t) = K_p e(t) = K_p (r(t) - y(t)), \quad (5.10)$$

donde K_p es la ganancia proporcional. La desventaja que presenta esta acción por si misma es que genera error en estado estacionario.

La acción derivativa de control realiza la predicción los valores futuros del error, y comprende a la ecuación 5.11

$$u(t) = K_d \frac{de(t)}{dt}, \quad (5.11)$$

donde K_d es la ganancia derivativa. Aunque el control derivativo no afecta en forma directa al error en estado estacionario, añade amortiguamiento al sistema y, por tanto, permite un valor más grande que la ganancia K_p , lo cual provoca una mejora en la precisión en estado estacionario. La principal desventaja está relacionado con la amplificación del ruido medido en la variable de interés.

La conjunción de estas dos acciones de control dan como resultado un controlador PD.

Control proporcional derivativo para el quadrotor

Es importante notar que debido a los sensores que posee el quadrotor, acelerómetros y giroscopios (los primeros dan una medición del el ángulo de rotación, mientras que los segundos dan la razón de cambio de dichos movimientos), hacen que el PD sea un controlador útil ya que los datos medidos de los sensores sean introducidos directamente al lazo de control.

La estructura del controlador es un controlador PD para cada uno de los estados.

El ángulo de *yaw* es el que menor relevancia tiene para el control debido a que no afecta en el movimiento traslacional de la aeronave, ya que este ángulo gira alrededor de z . Por esta razón, el controlador PD de *yaw* puede ser sintonizado de forma independiente a los demás. La ley de control PD para el ángulo de *yaw* está descrita por la ecuación

$$\tau_\psi = k_{p,\psi}(\psi^d - \psi) + k_{d,\psi}(\dot{\psi}^d - \dot{\psi}) \quad (5.12)$$

donde ψ^d y $\dot{\psi}^d$ representan respectivamente el ángulo y la velocidad angular deseados.

Debido a que el diseño del quadrotor es simétrico en el plano xy , se puede suponer que los controladores PD para *pitch* y *roll* son similares, aunque independientes. Además de la estabilización, estos ángulos son encargados de los movimientos laterales x y y de la aeronave. La ley de control PD para el ángulo de *pitch* está descrita por la ecuación

$$\tau_\theta = k_{p,\theta}(\theta^d - \theta) + k_{d,\theta}(\dot{\theta}^d - \dot{\theta}) \quad (5.13)$$

De forma similar, la ley de control PD para el ángulo de *roll* está descrita por la ecuación

$$\tau_\phi = k_{p,\phi}(\phi^d - \phi) + k_{d,\phi}(\dot{\phi}^d - \dot{\phi}) \quad (5.14)$$

Debido a la simetría mencionada anteriormente, se puede suponer las ganancias K_p y K_d para los controladores PD de *pitch* y de *roll* deben ser valores muy cercanos.

Para efectos de esta tesis no se propondrá un control de altura, por lo que la fuerza de empuje total será la necesaria para contrarrestar el peso de aeronave y mantenerla en vuelo.

$$u = mg \quad (5.15)$$

La figura 5.1 muestra el diagrama de bloques correspondiente a las leyes de control en lazo cerrado correspondientes a la estabilización del quadrotor.

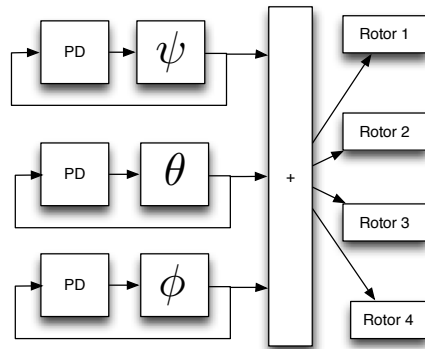


Figura 5.1: Diagrama de bloques del esquema de control

Los valores de la ganancias se muestran en la tabla 5.1, las cuales fueron ajustadas para cumplir las siguientes especificaciones: tiempo de subida $tr < 1s$, tiempo de establecimiento $ts < 5s$ y factor de amortiguamiento $0,4 \leq \xi \leq 1,3$. Debido a que la rotación en ψ no tiene ninguna relación con lo movimientos laterales a través de la inclinación (como lo hace θ o ϕ), se puede tener un controlador PD con ganancias más altas sin que el sistema se vuelva inestable.

Variable	Ganancia proporcional	Ganancia derivativa
Yaw	$k_{p,\psi} = 1$	$k_{d,\psi} = 2$
Pitch	$k_{p,\theta} = 0,4$	$k_{d,\theta} = 1$
Roll	$k_{p,\phi} = 0,4$	$k_{d,\phi} = 1$

Tabla 5.1: Ganancias para la simulación numérica del controlador PD

Tal sistema tiene por objetivo, además de la estabilización horizontal del helicóptero, la modificación de la respuesta dinámica del mismo de tal forma que pueda ser pilotado por un operador competente.

La figura 5.3 muestra las gráficas de salida de la simulación para los ángulos (ψ, θ, ϕ) y las velocidades angulares ($\dot{\psi}, \dot{\theta}, \dot{\phi}$)

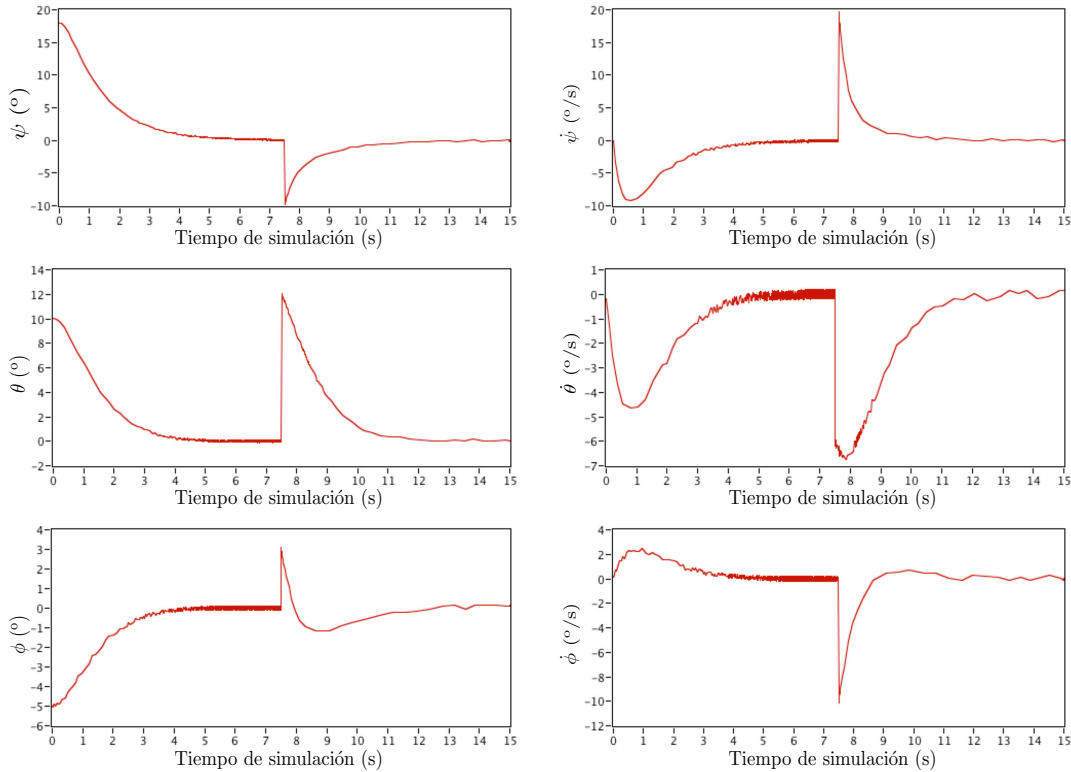


Figura 5.3: Gráficas de simulación de los ángulos y velocidades angular

y la figura 5.4 complementa la simulación con las gráficas de las acciones de control ($\tau\psi$, $\tau\theta$, $\tau\phi$)

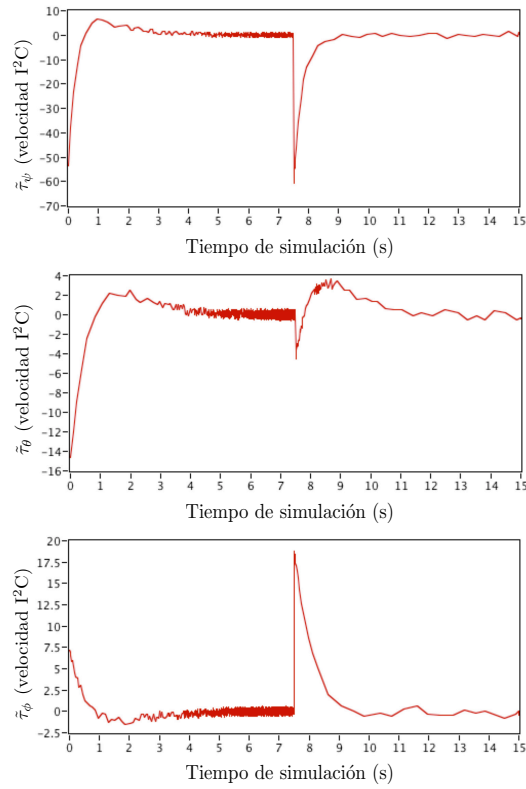


Figura 5.4: Gráficas de simulación de las acciones de control

Se puede ver al inicio la evolución de las salidas (ψ , θ , ϕ), las cuales tienen como valor inicial a $\eta_0 = (18, 10, -5)$ grados, las cuales convergen aproximadamente a cero grados, considerando que el tiempo de establecimiento está alrededor de los dos segundos, el tiempo de subida es menor a un segundo y posee un comportamiento subamortiguado, cumpliendo así el diseño del controlador PD. Es importante notar que existe una perturbación en $t = 7,5$ s, la cual modifica la orientación y velocidad angular de la aeronave, pero el controlador PD actúa sobre la velocidad de los rotores, para estabilizarla y continuar con el vuelo estacionario, por lo que se concluye que el controlador propuesto basado en las especificaciones de diseño es capaz de estabilizar al quadrotor en presencia de perturbaciones en un tiempo finito.

Todas las señales de retroalimentación fueron contaminadas con ruido blanco periódico con la intención de asemejar la simulación numérica a condiciones de operación reales, las señales de los ángulos presentaban ruido blanco con una amplitud de 0.1° , mientras que las velocidades angulares con $0.3^\circ/\text{s}$.

5.4 Resumen

Este capítulo logra obtener un modelo linealizado en un punto de equilibrio inestable pero controlable (estabilizable) modificando las entradas del sistema (velocidades de los motores), a través del método de linealización Jacobiana.

Además se presenta el método de control lineal de orientación utilizado para su estabilización, así como la ganancias para dicho controlador. Este método es validado a través de simulaciones, las cuales serán representadas a través de las gráficas de entrada-salida del sistema. Dichas simulaciones son contaminadas con ruido blanco haciendo referencia a mediciones reales, además de perturbaciones considerables que permiten visualizar la evolución de las leyes de control propuestas ante este tipo de estímulos que suceden en el mundo real.

Capítulo 6

Implementación en hardware

La mayoría de los prototipos existentes integran sus leyes de control, medición y comunicaciones de datos en dispositivos como: microcontroladores [23], microprocesadores [24] o procesadores digitales de señales (por sus siglas en inglés DSP) [17], los cuales a pesar de la frecuencia de oscilación a la cual trabajan, están atados a ejecutar un código de forma serial. Actualmente, la tecnología ha proveído a dichos dispositivos con múltiples núcleos de procesamiento, pero de igual forma las herramientas de programación se han vuelto más complejas en términos de sincronización de tareas y el compartimiento de recursos como el acceso a memoria o a los periféricos que integran al sistema.

Por otro lado existen otras soluciones orientadas a la descripción de hardware en las cuales el diseñador puede virtualmente crear cualquier circuito o circuito que pueden desempeñar tareas concurrente en el mismo dispositivo limitado solo por la capacidad de recursos de éste. Uno de los más importantes es el FPGA que tiene como ventajas principales: funcionalidad exacta sin desperdicio de recursos, concurrencia de tareas y gran desempeño de cálculo sobre tiempo. El FPGA es programado a través de lenguajes de descripción de hardware, siendo el más popular el VHDL (acrónimo que representa la combinación de VHSIC y HDL, donde VHSIC es el acrónimo de Very High Speed Integrated Circuit y HDL es a su vez el acrónimo de Hardware Description Language). El FPGA a usar es producido por Altera, por ello se usa el ambiente de desarrollo llamado Altera Quartus II v10 [1].

6.1 Consideraciones

El modelo matemático linealizado expuesto en el capítulo previo describe que el comportamiento dinámico del quadrotor, está basado en una fuerza de empuje y tres momentos de inercia, y a partir de estos el quadrotor es capaz de generar todos los movimientos de un sólido que se desenvuelve en un medio tridimensional (considerando su orientación y traslación). Existen factores adicionales que no se habían tomado en cuenta al inicio de la construcción del quadrotor, los cuales afectan el desempeño de éste y que hacen prácticamente imposible su estabilización. Estos factores se hacen presentes cuando el sistema se controla en lazo cerrado o retroalimentado y el sistema oscila cada vez con mayor amplitud o cuando una perturbación mínima lo convierte en inestable.

La metodología de diseño y construcción que se siguió para el prototipo fue recopilada

de los resultados publicados en diversos artículos científicos los cuales coinciden en los componentes principales, pero existe un problema relacionado a este último factor, ya que el comportamiento de dos productos de la misma marca y modelo pueden ser tan diferentes que puedan llegar a ocasionar la inestabilidad del sistema.

El circuito de los rotores está comprendido por dos dispositivos: un controlador de velocidad para motores sin escobillas y un motor sin escobillas. A través de diversas pruebas se comprobó que los rotores variaban hasta por 10 revoluciones por segundo, considerando el mismo voltaje de entrada y misma hélice, esta diferencia de velocidad ocasiona un ligero momento de inercia del rotor más potente hacia el más débil; cuando la central inercial detecta la inclinación generada, el controlador interno la compensa y se genera en un vaivén con una mayor amplitud con cada oscilación, haciendo inestable al sistema.

Dada esta razón el controlador PD concebido fue mejorado experimentalmente, agregando un sistema de medición de velocidad para cada uno de los rotores, creando así una nueva ley de control proporcional para regular la velocidad de cada rotor de forma independiente. La figura 6.1 muestra el nuevo esquema de control que logra la estabilización del prototipo.

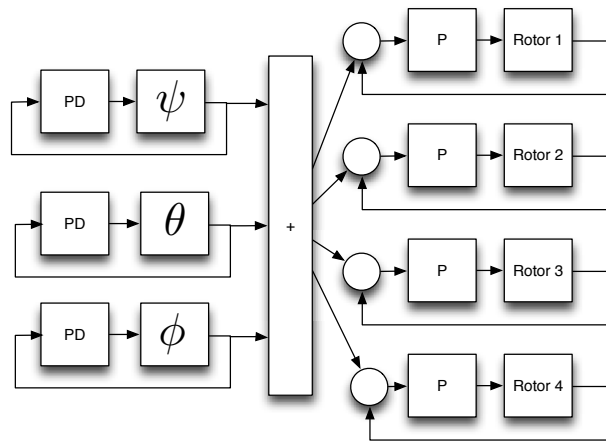


Figura 6.1: Diagrama de bloques de la nueva estructura de control

Aproximación a un motor de CD

Los motores de CD son un elemento básico del modelado de sistemas y pueden servir como aproximación para modelar motores debido a que requieren un voltaje de entrada y generan un par y una velocidad angular a la salida. Para efectos de esta tesis el motor sin escobillas será aproximado a través de un motor de corriente directa.

Los motores son dispositivos que transforman la energía eléctrica en energía mecánica, lo cual ocasiona un par de giro y una velocidad en el eje del motor. Por tanto, un motor tiene 2 subsistemas: uno eléctrico y otro mecánico. La figura 6.2 muestra un esquema general de un motor de CD con una hélice de carga.

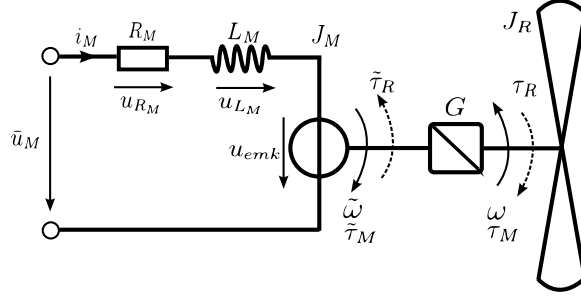


Figura 6.2: Esquema de un motor de CD

El subsistema eléctrico se denota por la ecuación:

$$\tilde{u}_M(t) = R_M i_M(t) + L_M \frac{di_M(t)}{dt} + K_e \tilde{\omega}(t) \quad (6.1)$$

donde \tilde{u}_M es la fuerza electromotriz, i_M es la corriente que circula por el circuito eléctrico, R_M es la resistencia del motor, L_M es la inductancia del motor, K_e es la constante electromotriz y $\tilde{\omega}$ es la velocidad angular del motor.

El subsistema mecánico se puede denotar como:

$$\tilde{\tau}(t) = \tilde{J}_{tot} \dot{\tilde{\omega}}(t) + b \tilde{\omega}(t) \quad (6.2)$$

donde el $\tilde{\tau}$ es el par generado por el motor, \tilde{J}_{tot} representa la inercia total del sistema y b es el coeficiente de fricción viscosa.

Al circular una corriente i_M por el devanado inducido, como resultado de la interacción con el campo magnético se ejerce sobre él un par $\tilde{\tau}$ que es directamente proporcional al campo magnético y a la propia corriente de inducido i_M . Considerando que el campo magnético es constante, se tiene que:

$$\tilde{\tau} = K_t i_M \quad (6.3)$$

Por otra parte, el giro del devanado en presencia del campo magnético, produce una caída de tensión o fuerza contraelectromotriz u_{emk} , proporcional a su velocidad de giro $\tilde{\omega}$, dado por:

$$u_{emk} = K_e \tilde{\omega} \quad (6.4)$$

Cambiando las ecuaciones antes descritas a dominio de Laplace y haciendo las sustituciones pertinentes podemos obtener la ecuación de transferencia del motor que vincula la velocidad angular $\tilde{\omega}$ del motor con el voltaje de alimentación \tilde{u}_M .

$$\frac{\tilde{\omega}(s)}{\tilde{u}_M(s)} = \frac{K_t}{(\tilde{J}_{tot}s + b)(L_M s + R_M) + K_t K_e} \quad (6.5)$$

La relación entre el par del eje del motor y el de la carga, así como las velocidades del motor y de la carga se dan por las siguientes ecuaciones:

$$\tilde{w} = Gw \quad (6.6)$$

y

$$\tau = G\tilde{\tau} \quad (6.7)$$

donde G representa la relación de transmisión entre ambos ejes.

Si se considera que L_M y b son muy pequeños y por lo tanto despreciables, se obtiene que:

$$\frac{\tilde{w}(s)}{\tilde{u}_M(s)} = \frac{K_t}{R_M \tilde{J}_{tot} s + K_t K_e} \quad (6.8)$$

Finalmente, haciendo los cambios de variable, la función de transferencia del motor de cd puede denotarse como:

$$\frac{\tilde{w}(s)}{\tilde{u}_M(s)} = \frac{K}{\alpha s + 1} \quad (6.9)$$

6.2 Componentes de hardware

El motivo principal de emplear un dispositivo de descripción de hardware como lo es el FPGA, se debe a la capacidad de crear módulos o componentes totalmente independientes entre sí, los cuales en conjunto son capaces de atender simultáneamente múltiples periféricos, calcular leyes de control concurrentes sensibles al tiempo de computo y la capacidad para hacer uso exclusivo de la cantidad de recursos de hardware necesarios, incluyendo a los buses de datos y señales de control.

El diseño realizado consiste en cinco componentes principales, de los cuales tres se encargan de los sensores, uno a los actuadores y el último calcula las leyes de control y sincroniza a los componentes mencionados previamente, además de un componente adicional usado para la depuración de errores y verificación de funcionamiento. A continuación se listan los componentes que conforman al diseño de los componentes de hardware:

1. imu
2. dsl
3. spdfb
4. ctrl
5. esc
6. smalltest

La figura 6.3 muestra un diagrama de bloques sintetizado de como se interconectan todos los componentes de hardware, así como el flujo de datos y protocolos de comunicación empleados.

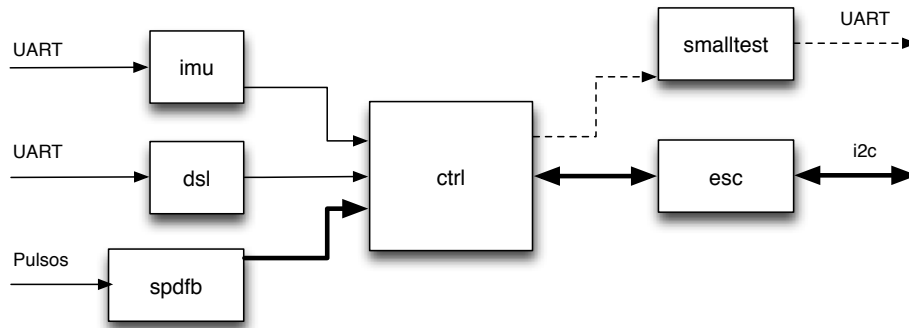


Figura 6.3: Diagrama de bloques de los componentes de hardware generales

A continuación se describen los componentes de hardware generales que le dan funcionalidad al sistema.

imu

El componente *imu* es el encargado de recibir una trama por el protocolo UART, la trama es enviada por el circuito de la central inercial con una frecuencia de actualización de 300 Hz. La figura 6.4 muestra el diagrama de bloques de los subcomponentes de *imu*.

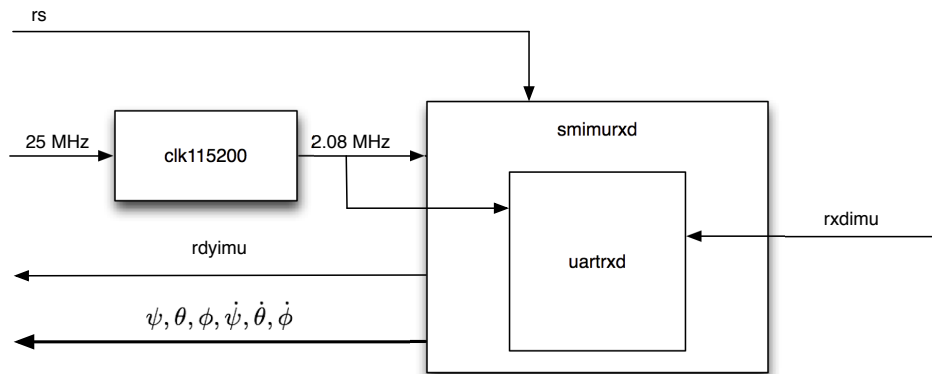


Figura 6.4: Diagrama de bloques de los subcomponentes de *imu*

El componente *clk115200* es un divisor de frecuencia a 2.08 MHz, necesario para emparar el algoritmo de captura de una trama UART que se envía a 115200 kbps. Con dicha frecuencia ajustada, el componente *uartrxd* recibe un byte de datos acompañados por bit de inicio y de fin de transmisión, activando así una bandera y es entonces cuando el dato es tomado por el componente *smimurxd*. Este componente se basa en una máquina de estados que

realiza la evaluación byte a byte de la trama, si ésta es correcta, actualiza los valores, sino son descartados.

La trama tiene un longitud de 10 bytes y se compone de los siguientes valores:

$$'s' \ 'n' \ 'p' \ \psi \ \theta \ \phi \ \dot{\psi} \ \dot{\theta} \ \dot{\phi} \ xor$$

donde $'s' \ 'n' \ 'p'$ son caracteres ASCII (acrónimo inglés de American Standard Code for Information Interchange) que actúan como cabecera de inicio de la trama, continúan $\psi, \theta, \phi, \dot{\psi}, \dot{\theta}, \dot{\phi}$, que son los ángulos y velocidades de navegación de la aeronave (filtrados y discretizados) a 8 bits; al terminar se recibe el valor calculado del xor de todos los datos de la trama a excepción de la cabecera. Es entonces cuando se realiza el cálculo del xor, si el valor recibido y el calculado son iguales, la trama se da por aceptada y se actualizan los datos, además de enviar una señal indicando que existen datos nuevos, sino, simplemente se ignoran y la máquina de estados espera a evaluar la siguiente trama.

Este componente al ser un sistema basado en máquina de estados, requiere una señal de reset rs para su correcta inicialización.

dsl

El componente *dsl* se encarga de recibir a través de un puerto de recepción UART, toda la información proveniente del equipo de radio control, con el cual el piloto puede lograr movimientos sobre la aeronave. La figura 6.5 muestra los subcomponentes que lo conforman.

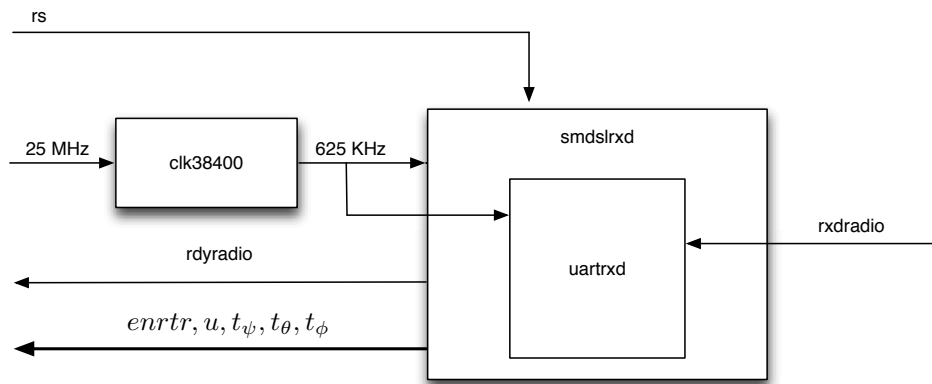


Figura 6.5: Diagrama de bloques de los subcomponentes de *dsl*

La velocidad de transferencia del receptor de radio control está ajustada a 38,400 bps, por ello es necesario la incorporación del componente *clk38400* cuyo trabajo es el de dividir la frecuencia para ajustarse a las características de la comunicación. El componente *uartrxd* es el encargado de recibir byte a byte, cada vez que se ha completado la recepción, éste la envía al componente *smdslrxd*, el cual por medio de una máquina de estado verifica que los datos sean correctos y en el orden adecuado. La trama que el componente toma como

válida es:

$$0xFF \ 0xFF \ 0xFF \ u \ t_{\psi} \ t_{\theta} \ t_{\phi} \ enrtr \ checksum \quad (6.10)$$

Los primeros tres bytes (denotados por $0xFF$) forman la cabecera de la trama, después se reciben los valores de las palancas de mando de transmisor de radio control ($u, t_{\psi}, t_{\theta}, t_{\phi}, enrtr$), para culminar con el *checksum* de dichos valores. Si el *checksum* calculado y el recibido concuerdan, entonces los datos contenidos en la trama son validos, sino son descartados. Si la trama es válida se activa una bandera indicando que existen nuevos datos disponibles en el registro.

Este componente al ser un sistema basado en máquina de estados, requiere una señal de reset *rs* para su correcta inicialización.

spdfb

El componente *spdfb* es el encargado de convertir pulsos de entrada de los circuitos de medición de velocidad de los rotores en revoluciones por unidad de tiempo. La figura 6.6 muestra el diagrama de bloques de los subcomponentes de *spdfb*.

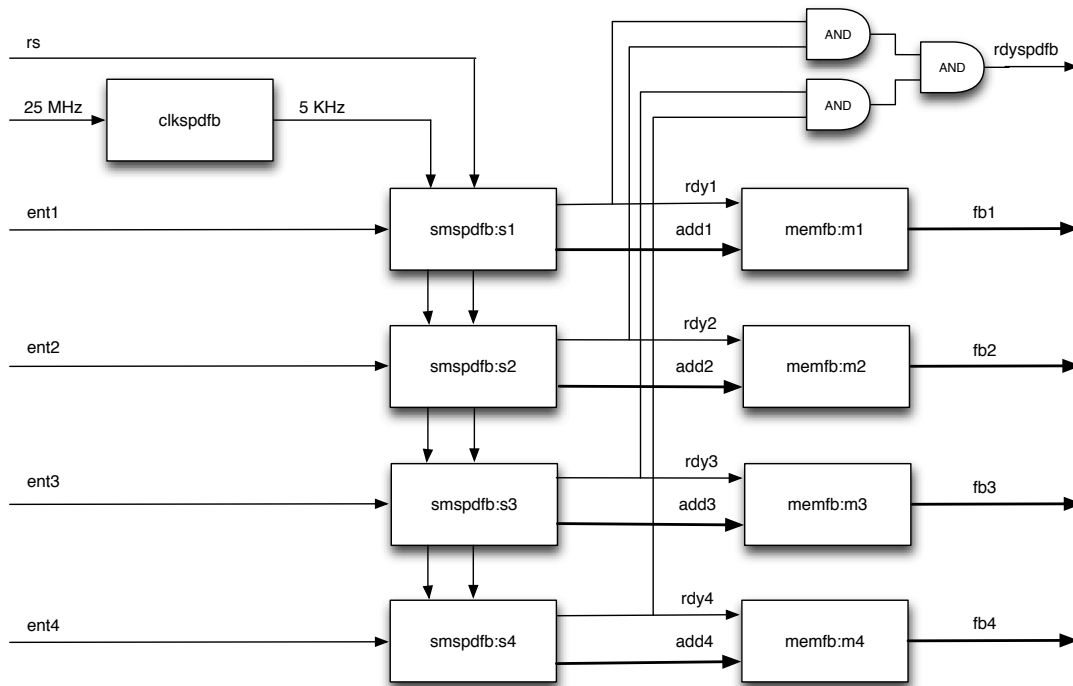


Figura 6.6: Diagrama de bloques de los subcomponentes de *spdfb*

El componente *clkspdfb* recibe la señal del reloj principal del FPGA y la divide para obtener un reloj de 5 KHz. Es entonces cuando el componente *smspdfb* hace uso de este nuevo reloj para realizar el muestreo de las entradas de pulsos, es decir, aumenta un contador cada vez

que se detecta un flanco positivo durante un periodo de 25 ms, cuando este temporizador expira, el nuevo valor es puesto en un registro y poniendo en alto una bandera que indica que existe un nuevo dato; en el siguiente ciclo de reloj se reinicia el contador para obtener una nueva medición y se restablece la bandera. Es importante recalcar que el periodo de 25 ms se obtuvo experimentalmente debido a que por especificaciones del fabricante el rotor tiene un tiempo de respuesta mínimo de 20 ms para cambiar de velocidad, entonces en base a este valor se decidió optar por 5 ms adicionales para garantizar la estabilización del propio rotor.

Es importante tener en mente el funcionamiento del rotor en este componente, ya que su comportamiento es no lineal y depende principalmente de tres factores: tipo de hélice, voltaje de alimentación y peso de la aeronave. Debido a esta no linealidad, es necesario encontrar una relación entrada (valor I²C) - salida (pulsos por unidad de tiempo), con la intención de utilizar datos siempre con las mismas unidades y reduciendo así el tiempo de computo. Es por esto que los rotores fueron caracterizados simultáneamente introduciendo el rango completo de valores I²C (0 - 120) y almacenando los resultados obtenidos a través del circuito de medición de velocidad. La figura 6.7 muestra las gráficas que relacionan los valores de entrada-salida de los rotores.

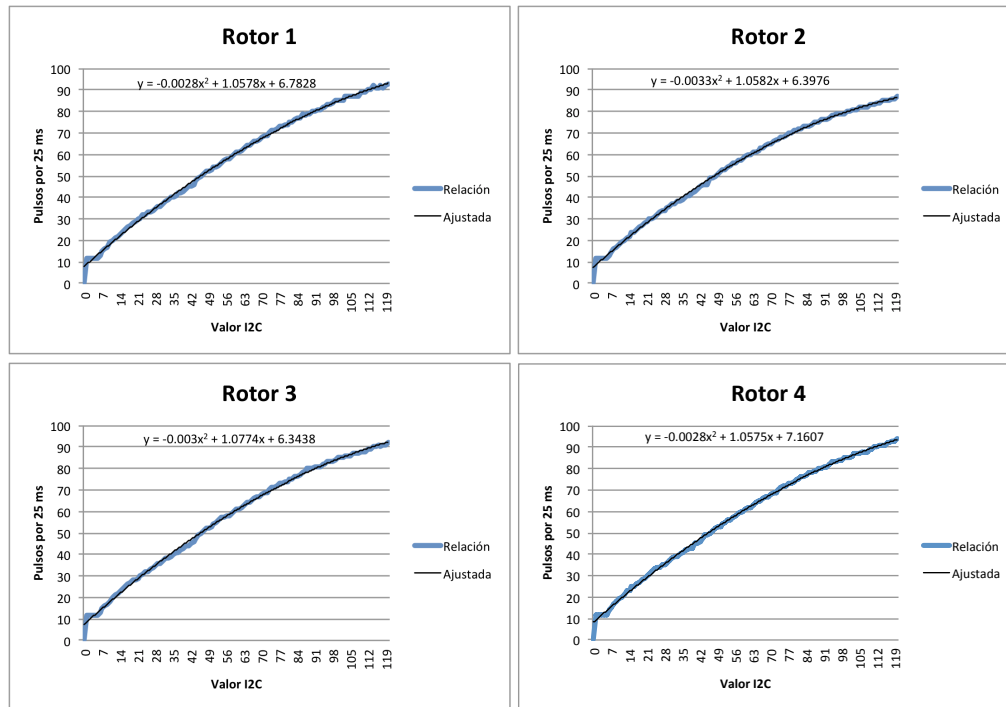


Figura 6.7: Gráficas de la relación de los valores de entrada-salida de los rotores

A dichas gráficas se les realizó un ajuste de curva aproximándolas a un polinomio de segundo grado; la ecuación característica de dicho polinomio fue despejada para así obtener la relación inversa es decir, pulsos por unidad de tiempo por valor I²C. Al ser una fun-

ción cuadrática solo se tomó en cuenta los resultados positivos, para ya al final crear el componente *memfb*, el cual es una memoria que contiene dicha relación inversa, la cual se direcciona por la cantidad de pulsos por unidad de tiempo y arroja como resultado el valor de I²C correspondiente de cada rotor.

Cuando la operación es completada por todos los componentes *smspdfb*, cada uno prende una bandera; la conjunción de todas las banderas a través de compuertas *and* conforman la señal de listo del componente *spdfb* completo.

Al ser un circuito secuencial, debe ser inicializado por medio de una señal de reset *rs*.

ctrl

ctrl es el componente descrito en hardware más importante, ya que se encarga de sincronizar al resto componentes que atienden periféricos (sensores), a través de los estados de sus banderas y por consiguiente el paso de datos por medio de registros. Con la información contenida en dichos registros hace el cálculo de las leyes de control y envía los resultados de velocidad a los componentes que se relacionan con los actuadores (rotores).

El concepto general reside que en cada momento que se actualizan los datos de la central inercial (300 Hz), entran a un controlador PD que se encarga de orientar a la aeronave; en conjunción con las salidas del controlador de orientación y con la velocidad medida de los rotores se calcula la salida para cada uno de los motores a través de un controlador proporcional (P). A esta salida calculada se le suman los valores recibidos por el piloto para inclinar a la aeronave y lograr el movimiento deseado, ya sea en orientación o en traslación. La figura 6.8 muestra a diagrama de bloques la idea general del componente *ctrl*, principalmente el flujo de datos para hacer el cálculo de las leyes de control y sus entradas y salidas correspondientes.

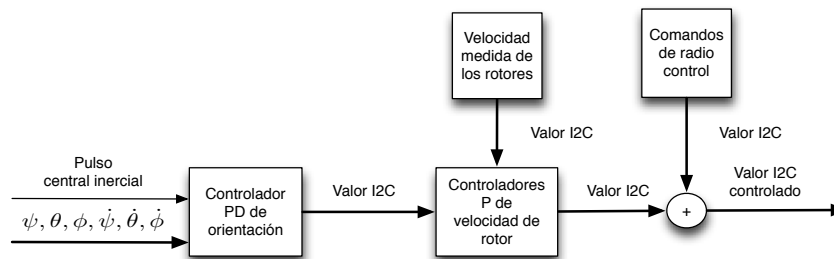


Figura 6.8: Diagrama de bloques simplificado del controlador de estabilización

Ampliando un poco la descripción del componente *ctrl*, éste se conforma de múltiples subcomponentes que se ejecutan secuencialmente a la frecuencia del reloj de la tarjeta CMCS002-2M (25 MHz), por lo cual, se consideraron etapas de acoplamiento ya que los componentes que atienden a los sensores y actuadores trabajan a frecuencias considerablemente más lentas. La ventaja principal de esto es aprovechar la frecuencia de oscilación mayor para hacer la mayor cantidad de operaciones en el menor tiempo posible, dándole así al sistema la posibilidad futura de probar leyes de control más complejas y que requieran

de más operaciones de procesamiento.

El componente *pulser* se encarga de hacer el acomplamiento en dos componentes con frecuencia de oscilación distinta, es decir, tiene como entrada la señal de nuevos datos en el registro de la central inercial, el cual es un pulso a 2.08 MHz y lo convierte en un pulso de 25 MHz, con la finalidad de solo disparar el proceso del computo cada vez que el evento sucede y no múltiples ocasiones, debido a la diferencia con la longitud del periodo del pulso original. Cuando los nuevos datos son recibidos por el componente *pdctrl*, este copia los registros de los datos actualizados de la central inercial ($\psi, \theta, \phi, \dot{\psi}, \dot{\theta}, \dot{\phi}$) y las ganancias de los controladores PD ($K_{p,\psi}, K_{p,\theta}, K_{p,\phi}, K_{d,\psi}, K_{d,\theta}, K_{d,\phi}$), los cuales se aplican directamente y en el orden descrito. Este controlador busca que la orientación de la aeronave tanto en sus ángulos como en sus velocidades angulares siempre tiendan a 0. Terminado este proceso se activa una bandera que le indica al componente *spderror* que los datos calculados ya se encuentran en el registro y están disponibles.

spderror realiza el computo de la velocidad deseada en cada uno de los rotores en base al controlador PD de orientación además de la velocidad adicional relacionada con la fuerza de empuje u para mantener el helicóptero suspendido y se le resta la velocidad actual de giro en los rotores, con esto se obtiene un valor de error. Las ecuaciones que describen el calculo de este error se basan en el diagrama de cuerpo libre presentado en la figura 4.3 y son:

$$\begin{aligned} errspd1 &= (u + c_{p,\psi} + c_{p,\theta} - c_{p,\phi} + c_{d,\psi} + c_{d,\theta} - c_{d,\phi}) - fb1 & (6.11) \\ errspd2 &= (u - c_{p,\psi} + c_{p,\theta} + c_{p,\phi} - c_{d,\psi} + c_{d,\theta} + c_{d,\phi}) - fb2 \\ errspd3 &= (u + c_{p,\psi} - c_{p,\theta} + c_{p,\phi} + c_{d,\psi} - c_{d,\theta} + c_{d,\phi}) - fb3 \\ errspd4 &= (u - c_{p,\psi} - c_{p,\theta} - c_{p,\phi} - c_{d,\psi} - c_{d,\theta} - c_{d,\phi}) - fb4 \end{aligned}$$

Una vez que el cálculo ha concluido, el componente activa una bandera que indica que los errores en los registros ya están disponibles para que el componente *spdpcrtl* copie dichos registros al igual que la ganancia de controlador de velocidad P ($K_{p,errspd}$) y los multiplique, creando así el lazo retroalimentado de cada uno de los rotores. Cuando la operación ha concluido, se prende la bandera para que el componente *spdtrtr* copie los registros que contienen los nuevos resultados ($cm1, cm2, cm3, cm4$). Con estos valores en adición a los comandos recibidos del sistema de radio control ($u, t_\psi, t_\theta, t_\phi$) describen las ecuaciones finales que dotan al sistema no sólo de estabilidad sino también de operabilidad para el piloto; las ecuaciones son:

$$\begin{aligned} spm1 &= u + t_\psi + t_\theta - t_\phi + cm1 & (6.12) \\ spm2 &= u - t_\psi + t_\theta + t_\phi + cm2 \\ spm3 &= u + t_\psi - t_\theta + t_\phi + cm3 \\ spm4 &= u - t_\psi - t_\theta - t_\phi + cm4 \end{aligned}$$

Cuando se ha concluido el cálculo, *spdtrtr* verifica el estado el componente *esc*, este componente se encarga de la transacción de datos I²C con los controladores de velocidad para

motores sin escobillas. Si el componente no está en espera del dato, entonces *spdrtr* actualiza el registro con los valores calculados (*spm1*, *spm2*, *spm3*, *spm4*), en caso contrario el valor del registro no es actualizado y los datos son desechados. En este punto también se realiza un acoplamiento de frecuencias debido a que el lazo de control se ejecuta a 25 Mhz, mientras que el componente *esc* tiene un divisor a 403 KHz.

La figura 6.9 muestra un diagrama de bloques detallado del componente *ctrl*, considerando las entradas, las salidas, señales de interconexión y el flujo de datos entre subcomponentes.

Cabe resaltar que los componentes descritos son secuenciales por lo que hacen uso de la señal de reset *rs* para su correcta inicialización.

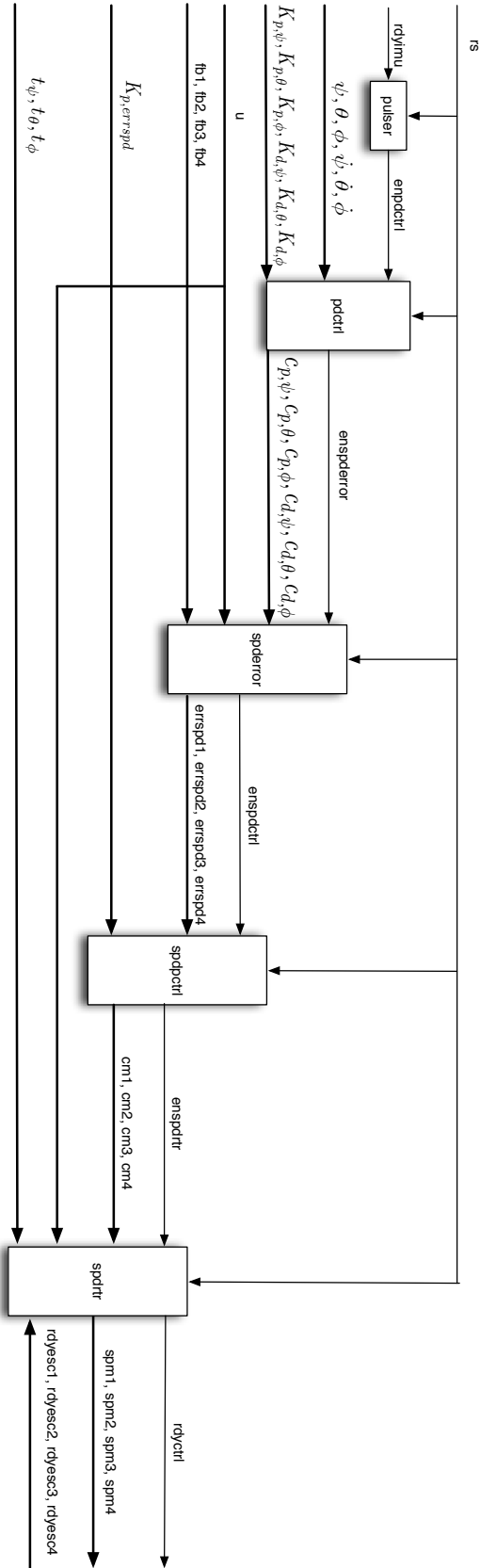


Figura 6.9: Diagrama de bloques de los subcomponentes de *ctrl*

esc

El componente *esc* se encarga de recibir los nuevos datos de velocidad para cada uno de los rotores y enviarlos por medio del protocolo I²C paralelamente y en sincronía. La figura 6.10 muestra el diagrama de bloques de los subcomponentes de *esc*.

El componente *clki2c* es un divisor de frecuencia a 403 KHz, este reloj es usado por el resto de los subcomponentes. El componente *smesc* es una máquina de estados la cual tiene diversas funciones, como habilitar o deshabilitar los motores por medio de la señal *enrtr* (paro de emergencia), inicializar los controladores de velocidad para motores sin escobillas para su arranque y actualizar la velocidad de los rotores.

Cuando el componente *smesc* no está enviando una trama y ocurre un evento a partir del cálculo del control, entonces copia el registro con los valores actualizados de los rotores (*spm1*, *spm2*, *spm3*, *spm4*) y los pasa al componente *i2cesc*. Este último componente se encarga de generar la trama I²C, la cual se basa en dos líneas de transmisión-recepción para llevar a cabo la comunicación, la de datos (*sdaesc*) y el reloj (*sclesc*). Propiedades adicionales acordes al protocolo como bit de inicio y fin de transmisión, *acknowledge* y *clock-stretching* fueron implementados, si se desea conocer a profundidad dichas características será necesario revisar el RFC de I²C de Philips.

Es importante hacer notar que el protocolo fue diseñado para trabajar en un esquema maestro-esclavo, pero para efectos prácticos de esta tesis, los buses se hicieron dedicados con la finalidad de actualizar todos los rotores síncronamente. Debido a esto, los dispositivos que están dentro del bus deben utilizar direcciones diferentes para poder ser accedidos de forma correcta, pero al tener un bus dedicado los controladores de velocidad para motores sin escobillas conservaron su dirección original (en hexadecimal 0x80). En la figura 6.11 se muestra la trama para actualizar la velocidad de los motores.

Donde el vector [D6 ... D0] indica la nueva velocidad dentro del rango de 0 a 120. En función a la naturaleza del protocolo, ambas señales *sdaesc* y *sclesc* son de entrada-salida, multiplexándose según la posición de la trama. En caso de que no exista señal de *acknowledge*, en alguna de las fases de la trama, se envía inmediatamente el bit de paro y se reintenta.

Al ser un circuito descrito a través de máquinas de estado, debe ser inicializado por medio de una señal de reset general *rs*.

smallest

El componente *smallest* se encarga de revisar datos de todos los componentes que integran el diseño completo y enviarlos por protocolo UART con la intención de depurar al sistema y crear un historial de vuelo instantáneo y automático. La figura 6.12 muestra los subcomponentes que lo conforman.

El componente *clk115200* está presente ya que se requiere una velocidad de transmisión de 115,200 bps. El componente *smallestrx* recibe todas las variables de entrada, señales intermedias y salidas (u , t_ψ , t_θ , t_ϕ , ψ , θ , ϕ , $\dot{\psi}$, $\dot{\theta}$, $\dot{\phi}$, *spm1*, *spm2*, *spm3*, *spm4*, *fb1*, *fb2*, *fb3*, *fb4*), las almacena en un *buffer* y va pasando dato por dato al componente *uarttx* para su serialización de acuerdo al protocolo. En cuanto el proceso termina, se reinicia de nuevo

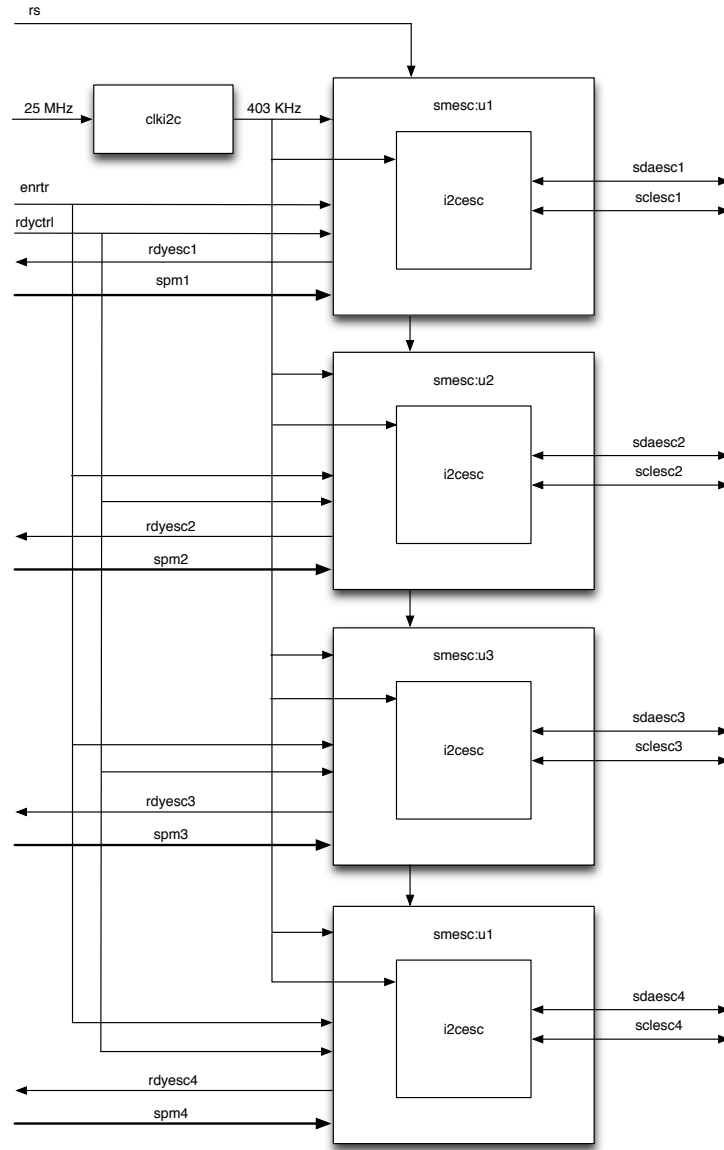


Figura 6.10: Diagrama de bloques de los subcomponentes de *esc*

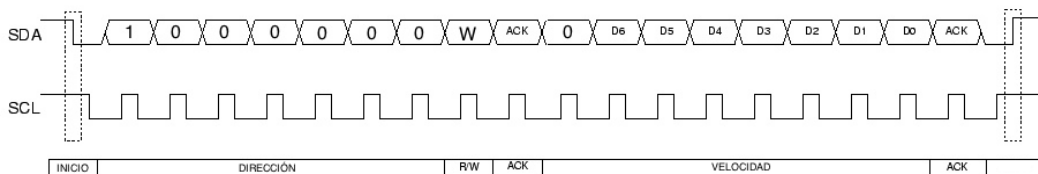


Figura 6.11: Trama I²C para actualizar la velocidad de los motores

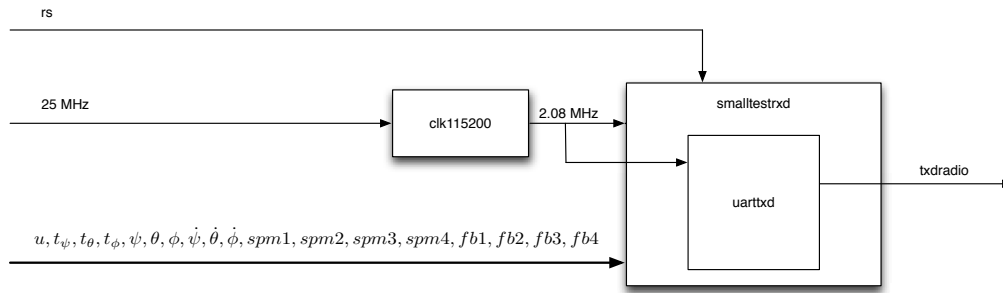


Figura 6.12: Diagrama de bloques de los subcomponentes de *smalltest*

con los nuevos parámetros. La trama a enviar se define a continuación:

's' 'n' 'p' u t_ψ t_θ t_φ ψ θ φ ψ̇ θ̇ φ̇ spm1 spm2 spm3 spm4 fb1 fb2 fb3 fb4 xor

Los caracteres ASCII 's' 'n' 'p' forman la cabecera de la trama; a todos los datos que indican el estado del vehículo se les realiza la operación xor de forma recursiva y se anexa al final de trama, así el destinatario puede comprobar la integridad de los datos y saber así si son útiles o no.

Este módulo es independiente a los demás por lo que no aporta retardos al sistema, simplemente monitorea y envía el estado general del quadrotor, por lo que éste puede no ser sintetizado en el diseño ganando así recursos adicionales. Al igual que los componentes anteriores, al tener máquinas de estado para la resolución, se debe inicializar a través del señal general de reset *rs*.

6.3 Recursos utilizados

Esta sección hace referencia a la tabla 6.3, la cual muestra la relación de los recursos usados del FPGA Altera EP3C25-C8 después de la síntesis del diseño. Es importante notar que la mayoría de los recursos utilizados son elementos lógicos, ya que el sistema en su totalidad se basa en circuito secuenciales y máquinas de estado, las cuales aparte de lógica combinatoria hacen uso de los registros como almacenamiento temporal.

Recursos	Porcentaje usado	Cantidad usada
Total de elementos lógicos	28 %	6,815/24,624
Total de pines	16 %	25/157
Total de bits de memoria	1 %	3,960/608,256
Multiplicadores de 9 bits embebidos	11 %	14/132
Total de PLLs	0 %	0/4

Tabla 6.1: Recursos del FPGA utilizados por el diseño

Se puede observar que existe una gran cantidad de recursos dentro del FPGA que pueden

ser usados para aumentar la versatilidad del quadrotor o agregar nuevas funciones sin que el tiempo del procesamiento de la ley de control se vea comprometido.

6.4 Resultados experimentales

La sintonización de las ganancias de las leyes de control en el prototipo puede llegar a ser difícil dado que, si las ganancias de control están mal ajustadas pueden dar lugar a colisiones. Mas aún, una vez que se encuentra un conjunto de ganancias aceptable, no es posible asegurar que dichas ganancias son las óptimas. Es por esto que el desarrollo de la simulación numérica del sistema en lazo cerrado ayudó a determinar los valores iniciales de las ganancias, para después realizar el ajuste de estas directamente en el prototipo, logrando así la estabilización del quadrotor.

Los valores finales de las ganancias obtenidas de manera experimental se muestran en la tabla 6.2.

Variable	Ganancias proporcionales	Ganancias derivativas
Yaw	$k_{p,\psi} = 1,3$	$k_{d,\psi} = 2$
Pitch	$k_{p,\theta} = 0,23$	$k_{d,\theta} = 1,2$
Roll	$k_{p,\phi} = 0,2$	$k_{d,\phi} = 1$

Tabla 6.2: Ganancias experimentales para controlador embebido en FPGA

Estas ganancias fueron ajustadas después de 53 vuelos y reflejan un punto donde se alcanza la estabilización del prototipo. Adicionalmente a las ganancias de los controladores descritas en la tabla 6.2, la ganancia del controlador proporcional de velocidad ($K_{p,errspd}$) fue establecida en 1. Este valor de ganancia unitario fue obtenido experimentalmente durante los vuelos de la aeronave.

Después de ajustar las ganancias, los datos útiles que describen el comportamiento interno fueron enviados y almacenados por periodo aproximado de 5 minutos en vuelo cuasiestacionario. La intención de este vuelo, es que un operador competente sea capaz de pilotear la aeronave principalmente haciendo movimientos sobre el ángulo de *yaw* y pequeños movimientos sobre *pitch* y *roll*, alcanzando así desplazamientos laterales sin que el sistema se desestabilice.

La figura 6.13 muestra las gráficas del vuelo suspendido para los ángulos (ψ, θ, ϕ) y las velocidades angulares ($\dot{\psi}, \dot{\theta}, \dot{\phi}$) medidas en referencia al marco inercial (\mathcal{I}).

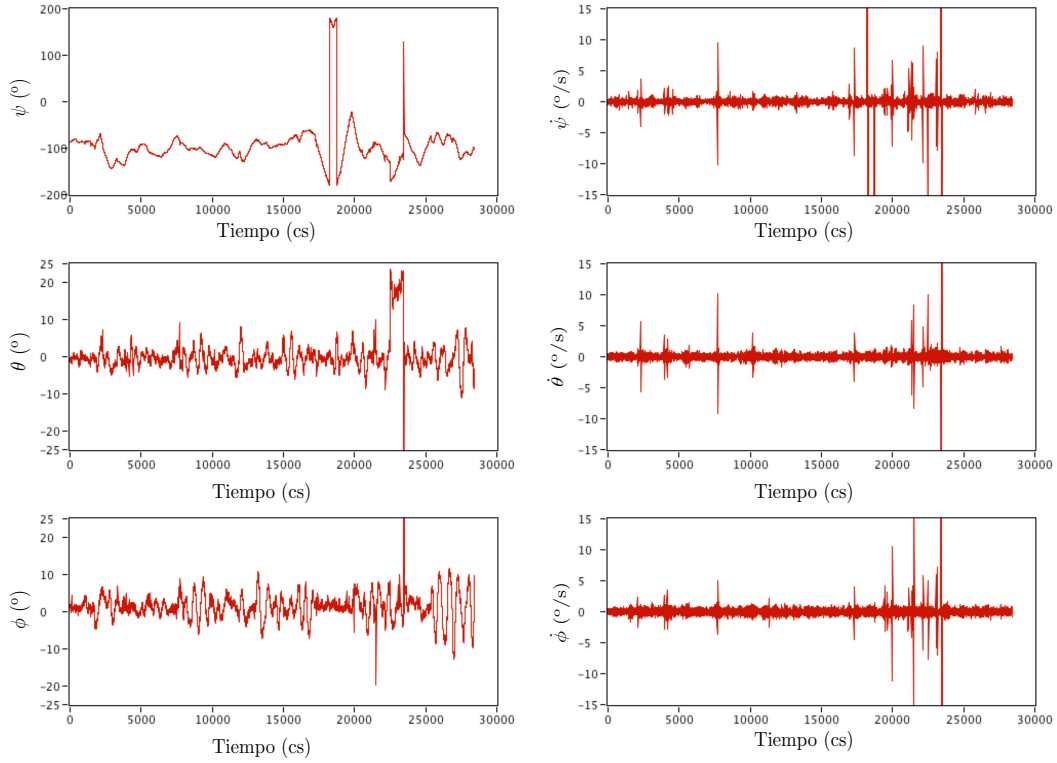


Figura 6.13: Gráficas experimentales de los ángulos y velocidades angulares

La figura 6.14 muestra las gráficas de las velocidades I²C generadas a partir de las acciones de control ($\tau_\psi, \tau_\theta, \tau_\phi$).

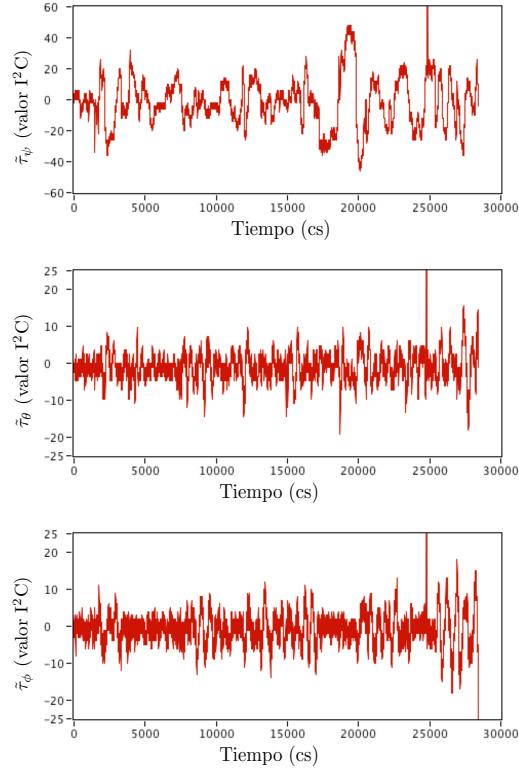


Figura 6.14: Gráficas de la evolución de las acciones de control en el prototipo

Cabe resaltar que como el controlador actúa ante los cambios o perturbaciones de las variables medidas, esto ocurre cuando $t \approx 18000$ cs, el operador presiona la palanca asignada a *yaw* con sentido negativo en el transmisor de radio control logrando así un mínimo menor a -180° , pero debido a la convención usada por el sistema, se convierte en un máximo tras superar el rango que lo delimita, después de ese momento el operador regresó la aeronave a la posición inicial. Es importante destacar el desempeño del controlador, ya que para el mismo tiempo propuesto, se observa un incremento en la ley de control τ_ψ , ocasionado por la perturbación del operador a través del transmisor de radio control. En este punto, el sistema ya está expuesto al ruido generado por múltiples factores, pero principalmente atribuido a la vibración del prototipo, errores de medición, errores de comunicación en el envío de datos e inclusive del propio filtrado, pero su evolución demuestra su capacidad para compensarlos.

Debido a que el sistema de control fue ampliado con respecto a la aproximación derivada del modelo matemático, a continuación se muestra en la figura 6.15 las gráficas de la velocidad medida para cada uno de los motores.

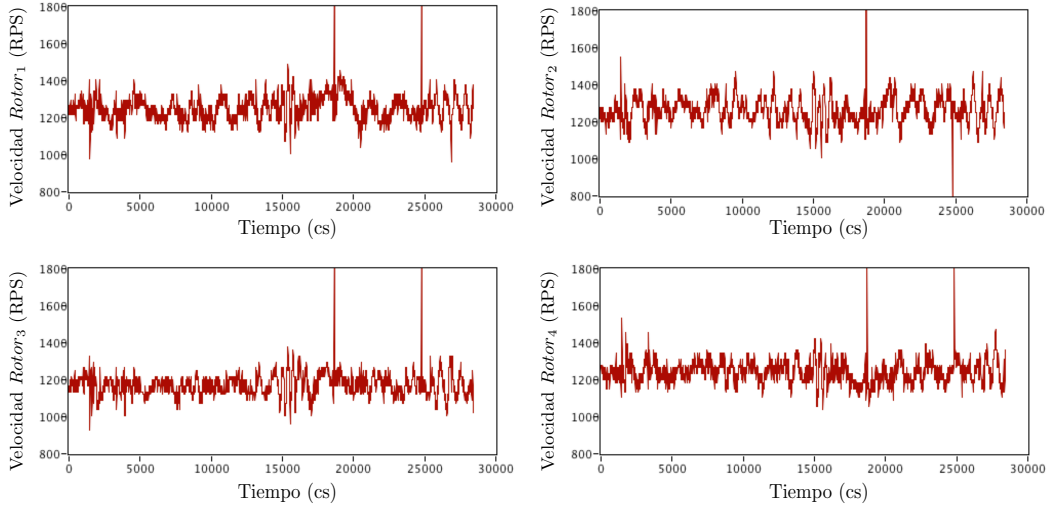


Figura 6.15: Gráficas experimentales de la velocidad angular medida de los motores

Las entradas de control para las velocidades motores se ilustran en la gráficas de la figura 6.16.

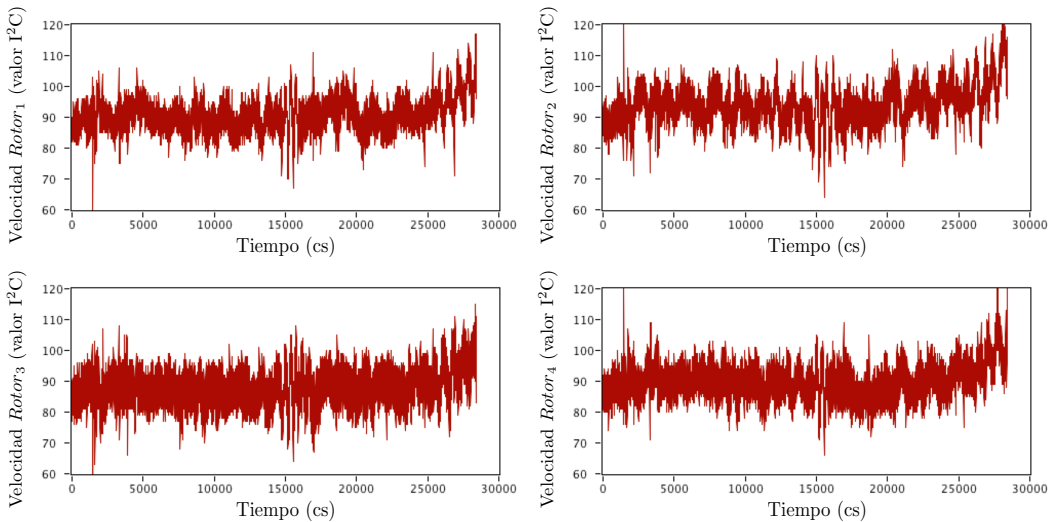


Figura 6.16: Gráficas experimentales de los valores I²C de los motores

Las gráficas de las entradas de control para los motores muestran un valor I²C promedio de 84, el cual es el valor mínimo necesario para la sustentación del helicóptero, este valor genera la fuerza de empuje necesaria para contrarrestar el peso de la aeronave. Al final de la

prueba, se observa que este valor se incrementa debido a pérdida de energía de la baterías, por lo que el piloto se ve en la necesidad de aumentar la fuerza de empuje para compensar este fenómeno.

Para la medición de los ángulos y velocidad angulares referenciados al marco inercial (T), se uso un sistema de captura de movimiento MotionAnalysis [3], en el cual por medio de marcadores reflectivos sujetos a la aeronave y a través de un sistema de cámaras se puede conocer y almacenar la dinámica de dicho objeto. La conexión hacia el sistema se hace a través del Robot Operating System (por sus siglas en inglés ROS) [9] en Linux, en el cual se crearon los nodos que centralizan los datos internos del quadrotor y del sistema de cámaras. La figura 6.17 presenta gráficamente un mapa tridimensional del vuelo realizado, en donde se encuentra el marco de referencia inercial (denotado como `/mocap`) y el marco de referencia del cuerpo (denotado por `/quadrotor`).

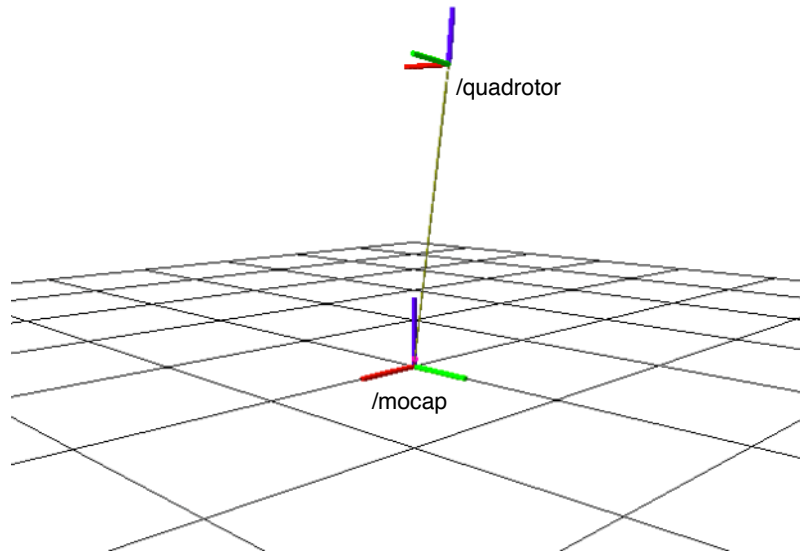


Figura 6.17: Visor tridimensional rviz de ROS

La figura 6.18 muestra el lugar de pruebas donde se realizaron los vuelos del quadrotor, el cual es una locación cerrada, con iluminación controlada para no afectar la medición de las cámaras infrarrojas.



Figura 6.18: Hangar de vuelo en la Universidad de Freiburg, Alemania

Vídeos

Al momento de redactar esta tesis se presentan dos vídeos disponibles en la red los cuales muestran el desempeño en diferentes locaciones cerradas, el primero es dentro de un cuarto reducido con múltiples objetos y el segundo es dentro del hangar de vuelo en la Universidad de Freiburg, Alemania donde el sistema de captura movimiento registra los datos de vuelo. Estos vídeos se encuentran en la siguientes ligas: http://www.youtube.com/watch?v=j_rdtAPrw0A y http://www.youtube.com/watch?v=_QEtRv9-Zqo. De igual forma, los vídeos están contenidos en el medio digital que acompaña a esta tesis.

6.5 Resumen

En este capítulo se detallan los componentes de hardware usados para la integración del quadrotor, que involucran la medición, la acción de control y las comunicaciones remotas. Se hacen también ciertas consideraciones relativas al modificar el controlador descrito en el capítulo anterior, así como su descripción extensa del lazo retroalimentado, incluyendo un análisis del consumo de recursos que ocasiona el diseño en el FPGA.

Se presentan las nuevas ganancias ajustadas al prototipo, así como las gráficas de entrada-salida, medidas por un sistema de captura de movimiento que comprueba de forma eficiente el desempeño del controlador, pero desde el marco de referencia inercial, obteniendo así las coordenadas generalizadas sin ruido o perturbaciones.

Capítulo 7

Conclusiones y trabajo futuro

7.1 Conclusiones y aportaciones

En México existe la necesidad de realizar tareas de búsqueda y reconocimiento de terreno cuando se presentan desastres naturales, para ello se hace uso de aeronaves tripuladas. Sin embargo, este tipo de operaciones tienen un alto impacto económico, además de poner en riesgo la vida de la tripulación. El presente trabajo muestra una primera aproximación del diseño, la construcción, el control y la implementación de un quadrotor de bajo costo que puede ser usado con dichos fines y así ayudar al ser humano en tareas difíciles como lo son los desastres. Todas las tareas listadas fueron llevadas a cabo sistemáticamente, constantando así que el prototipo final es capaz de permanecer en vuelo suspendido y con ayuda de un piloto generar movimientos orientación y traslación.

La construcción del quadrotor tuvo retos importantes ligados con la selección de materiales principalmente para la construcción de los brazos y la central inercial. Al iniciar el proyecto se escogió madera de balsa de 1/8 de pulgada, la cual fue forrada manualmente con una placa de fibra de carbono de 1/32 de pulgada de cada lado, después con un material compuesto de dos cubiertas de fibra de carbono de 1/16 de pulgada y un centro de espuma industrial con espesor 1/4 de pulgada; ambos materiales se rompieron después de que el helicóptero cayó al piso en repetidas ocasiones. Finalmente se escogió un material compuesto que incorpora madera de balsa con un espesor de 1/4 de pulgada y cubiertas de fibra de carbono de 1/16 de pulgada. Por otro lado, la central inercial serial CHRobotics CHR-6dm AHRS [22] escogida al inicio del proyecto, era extremadamente ruidosa, principalmente en las señales provenientes de los giroscopios, para contrarrestar este problema se diseñaron filtros digitales, se aisló ópticamente a la central inercial y se probaron diferentes retenedores antivibración, pero ninguno de estos métodos funcionó, la solución fue construir la central inercial con filtrado analógico.

El proceso de implementación fue más complicado que haciendo uso de cualquier otro dispositivo de procesamiento embebido, comúnmente un microcontrolador o un DSP, ya que estos poseen una serie de funciones, librerías y tipos de datos ya preestablecidos por el lenguaje de programación, lo cual hace que realizar una conversión de tipo de dato o una simple operación aritmética de tipo flotante puede describirse con una sola línea de texto, en cambio la descripción en VHDL es sumamente detallada y requiere que se cumplan ciertas característi-

cas principalmente relacionadas con el tiempo de propagación en las máquinas de estado que secuenciaban los procesos de ejecución y cálculo. Esto representó un gran factor en el tiempo de desarrollo de esta tesis. Pero por otro lado, justamente esta implementación representa la principal aportación de este trabajo, formando así los cimientos para futuros trabajos, que deseen ampliar el alcance del prototipo basado en un proyecto sólidamente construido y probado.

La adición de un controlador proporcional para cada rotor fue necesaria, ya que el prototipo fue construido en base a los artículos científicos reportados en el estado del arte, los cuales no hacían mención a problemas de estabilización por diferencias físicas inherentes a la mecánica de los rotores, con ésta modificación, se añadieron elementos electromecánicos no contemplados al inicio del proyecto, así como modificaciones en el diseño VHDL.

7.2 Trabajo futuro

La recompensa del trabajo bien hecho es la oportunidad de hacer más trabajo bien hecho. (Jonas Edward Salk). Suponiendo que la primera parte de esta célebre frase sea cierta, entonces no queda más indicar algunos de los trabajos a realizar en un futuro inmediato esperando que se cumpla la segunda parte de la afirmación del Dr. Salk y puedan ser una buena continuación de esta tesis.

Uno de los puntos más interesantes a continuar de esta tesis es culminar la obtención de parámetros a través de la técnica de identificación de sistemas, con la finalidad de conocer el modelo no lineal a detalle y mediante los datos de entrada y salida del sistema retroalimentado, esta técnica es capaz de ajustar de manera precisa aquellos parámetros que son complicados o prácticamente imposibles de medir, siendo benéfico por dos razones: proponer un control ya ajustado listo para probarse en el prototipo con ganancias óptimas y conocer las limitaciones del prototipo para llevarlo a condiciones de vuelo extremo en función de dichos parámetros para realizar piruetas o acrobacias, debido a que se conoce la dinámica no lineal, para esto es necesario proponer una estrategia de control diferente.

Otro punto a desarrollar es la autonomía del prototipo para cubrir una trayectoria específica sin la intervención de un piloto, principalmente para labores de reconocimiento y fotografía. El diseño sintetizado en FPGA consumió solo el 28 por ciento de sus recursos, es por esto que la adición de múltiples sensores es factible para expandir la operabilidad del quadrotor. Sistemas de localización como GPS (acrónimo en inglés de Global Positioning System), radares, sistemas de captura de movimiento, son ejemplos de nuevos sensores que pueden enriquecer la maniobrabilidad, integrando esto con técnicas de visión artificial y reconocimiento de patrones, principalmente flujo óptico, las opciones se multiplican para desarrollar una gran cantidad de tareas específicas. La parte interesante de lo descrito anteriormente es la capacidad de hacer toda esta integración sin afectar el tiempo de cómputo debido al paralelismo inherente al FPGA.

Por último integrar los dos puntos anteriores para crear escuadrones de quadrotores descentralizados los cuales realicen tareas de cooperativas, como lo son el levantar objetos pesados, búsqueda extensiva en zonas delimitadas o el mismo rescate de personas en situación de peligro, capaces de ser desplegados ante cualquier situación y con bajo costo de producción.

Nomenclatura

Acknowledge En I²C, es el acuse de recibo por parte del dispositivo destino cuando se ha transmitido en el bus un byte de dirección o de dato.

Buffer Ubicación en la memoria de un instrumento digital reservada para el almacenamiento temporal de información digital, mientras que está esperando ser procesada.

Clock-stretching En I²C, un dispositivo esclavo puede mantener la línea de reloj en bajo antes de recibir o mandar un byte, indicando así que no está listo para procesar más datos.

DSP Digital Signal Processor, es un sistema basado en un procesador o microprocesador que posee un juego de instrucciones, un hardware y un software optimizados para aplicaciones que requieran operaciones numéricas a muy alta velocidad.

Encoder Es un dispositivo electromecánico usado para convertir la posición angular de un eje a un código digital, lo que lo convierte en una clase de transductor.

FPGA Field Programmable Gate Array, es un dispositivo semiconductor que contiene bloques de lógica cuya interconexión y funcionalidad puede ser configurada en sitio mediante un lenguaje de descripción especializado.

I²C Inter-Integrated Circuit, es un protocolo de comunicaciones en serie.

MEMS Microelectromechanical Systems, se refieren a la tecnología electromecánica, micrométrica y sus productos.

Motor sin escobillas Es un motor eléctrico que no emplea escobillas para realizar el cambio de polaridad en el rotor.

Multirotor Helicóptero que posee tres o más rotores.

PPM Pulse Position Modulation, es un tipo de modulación en la cual una palabra de R bits es codificada por la transmisión de un único pulso que puede encontrarse en alguna de las 2^M posiciones posibles.

Trama Conjunto de datos con un formato predefinido empleado en comunicaciones entre dispositivos.

UART Universal Asynchronous Receiver-Transmitter, es un protocolo de comunicaciones en serie.

Bibliografía

- [1] Altera. *Introduction to the Quartus II Software*. URL: http://www.altera.com/literature/manual/archives/intro_to_quartus2.pdf.
- [2] E. Altug, J.P. Ostrowski y R. Mahony. “Control of a quadrotor helicopter using visual feedback”. En: *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No.02CH37292)* 1.May (2002), págs. 72-77. DOI: 10.1109/ROBOT.2002.1013341. URL: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1013341>.
- [3] Motion Analysis. *Raptor-H Digital RealTime System*. URL: http://www.motionanalysis.com/pdf/RaptorH_System.pdf.
- [4] Y. Bouktir, M. Haddad y T. Chettibi. “Trajectory planning for a quadrotor helicopter”. En: *2008 16th Mediterranean Conference on Control and Automation* (jun. de 2008), págs. 1258-1263. DOI: 10.1109/MED.2008.4602025. URL: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4602025>.
- [5] NXP B.V. *74HC14 Hex inverting Schmitt trigger*. URL: http://www.nxp.com/documents/data_sheet/74HC_HCT14.pdf.
- [6] P. Castillo y col. “Modelado y estabilización de un helicoptero con cuatro rotores”. En: *RIAI 4.1* (2009), págs. 41-57. URL: <http://84.88.0.49/index.php/RIAI/article/view/8046>.
- [7] DallasLogic. *CMCS002-2M User Guide Rel. 2*. URL: http://www.dallaslogic.com/prod_cmcs002m.htm.
- [8] Centro Nacional de Prevención de Desastres (CENAPRED). *Acerca de CENAPRED*. URL: <http://www.cenapred.gob.mx/es/>.
- [9] Documentation. *ROS (Robot Operating System)*. URL: <http://www.ros.org/wiki/>.
- [10] Young Generation Electronics. *Bedienungsanleitung und Datenblatt für die Drehzahlsteller YGE 18i und 30i Quadrocopter*. URL: http://lipoly.de/modellbau/steller/yge/manual_yge30i.pdf.
- [11] ACT Europe. *RC-Empfänger ACT DSL4top (Typ MK)*. ACT Europe. URL: <http://www.acteurope.de/Handbuchdtengl.pdf>.
- [12] H. Guanglin, G. Rujun e Y. Shi. “Application of FPGA in Small UAV Autopilot Based on Embedded Linux System”. En: *Industrial Electronics Society, 2007. IECON 2007. 33rd Annual Conference of the IEEE*. IEEE, 2007, págs. 731-734. ISBN: 1424407834. URL: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4459959.

- [13] Honeywell. *HOA1405 Reflective Sensor*. URL: http://sensing.honeywell.com/index.php?ci_id=50407.
- [14] INEGI. *Riesgos naturales*. URL: <http://www.inegi.org.mx/default.aspx>.
- [15] National Instruments. *LabVIEW User Manual*. URL: <http://www.ni.com/pdf/manuals/320999e.pdf>.
- [16] Graupner JR. *Remote Control MC-12 Rotary Select Programming Manual*. URL: <http://noticias.drac.com/.docs/anguera/tramc12.pdf>.
- [17] L. Klauske y col. "DSP-Copter-A Quadrotor Helicopter Controlled by a Digital Signal Processor". En: *ELITE-Project Report Summary* (2002), págs. 1-4. URL: <http://www.go-gddq.com/down/2011-05/11052821107366.pdf>.
- [18] J. Gordon Leishman. *Principles Helicopter Aerodynamics*. Cambridge University Press, 2005.
- [19] Rogelio Lozano. *Unmanned Aerial Vehicles Embedded Control*. Wiley-Iste, 2010.
- [20] Maxx Products. *Maxx Products 10x4.5 EPP1045 Counter Rotating Pair Propellers for RC Airplanes + RC Helicopters*. URL: http://www.quadrocopter.com/Propeller-pair-EPP1045-CF_p_319.html.
- [21] Robbe. *ROXXY BL-Outrunner datasheet*. Robbe. URL: <http://www.mikrokopter.us/shop/roxy-2827-35-robbe-roxy-2827-35>.
- [22] CH Robotics. *CHR-6dm Attitude and Heading Reference System*. URL: http://www.chrobotics.com/docs/chr6dm_datasheet.pdf.
- [23] Hugo Romero y col. "A new UAV configuration having eight rotors: Dynamical model and real-time control". En: *2007 46th IEEE Conference on Decision and Control* 2011.4 (2007), págs. 6418-6423. DOI: 10.1109/CDC.2007.4434776. URL: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4434776>.
- [24] Jared Rought y col. "Self-Stabilizing Quad-Rotor Helicopter". En: *eecs.ucf.edu* 2 (2010). URL: <http://www.eecs.ucf.edu/seniordesign/su2010fa2010/g07/user/image/conferencepaper.pdf>.
- [25] Universidad de Salamanca. *La matriz jacobiana*. URL: http://portalevlm.usal.es/INDEX_FILES/BASES/ARCHIVOS/TEMA6/MJACOBIANA.PDF.
- [26] *SolidWorks Professional*. DDS SolidWorks Corp., 2009. URL: http://www.solidworks.com/sw/products/10155_ENU_HTML.htm.
- [27] Sparkfun. *IMU Analog Combo Board Razor - 6DOF Ultra-Thin IMU*. Sparkfun. URL: <http://www.sparkfun.com/products/10010>.
- [28] A. Tayebi y S. McGilvray. "Attitude stabilization of a VTOL quadrotor aircraft". En: *Control Systems Technology, IEEE Transactions on* 14.3 (2006), págs. 562-571. URL: <http://www.nt.ntnu.no/users/skoge/prost/proceedings/ifac2008/data/papers/2578.pdf>http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1624481.
- [29] Antonio Visioli. *Practical PID control*. London: Springer, 2006.

- [30] *WiSnap User Guide*. Serialio. URL: http://serialio.com/support/wifi/WiSnap_User_Guide_2.21.1.pdf.
- [31] Hongxi Yang y Q. Geng. "The design of flight control system for small UAV with static stability". En: *Mechanic Automation and Control Engineering (MACE), 2011 Second International Conference on*. 201011080. IEEE, 2011, págs. 799-803. ISBN: 9781424494392. URL: http://ieeexplore.ieee.org/xpls/abs/_all.jsp?arnumber=5987048.

Apéndice A

Planos de dibujo

Anexo en CD.

Apéndice B

Código fuente

Anexo en CD.