

BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE PUEBLA

Facultad de Ciencias de la Electrónica

Ingeniería en Mecatrónica



BUAP

Tesis

“Diseño e implementación de un sistema de navegación y monitoreo ambiental para vehículos autónomos”

Tesis presentada como requisito para obtener el título en:

LICENCIATURA EN INGENIERÍA MECATRÓNICA

Presenta: Julián Ramos García

Asesor: Dr. José Fermi Guerrero Castellanos

Co-Asesor: Dr. Víctor Rodolfo González Díaz

Junio de 2016

Título: “Diseño e implementación de un sistema de navegación y monitoreo ambiental para vehículos autónomos”

Estudiante: JULIÁN RAMOS GARCÍA

COMITÉ

M.C. Carlos García Lucero
Presidente

Dr. Germán Ardul Muñoz
Hernández
Secretario

M.C. Jose Francisco Portillo
Robledo
Vocal

ASESORES

Dr. José Fermi Guerrero
Castellanos

Dr. Víctor Rodolfo González
Díaz

Resumen

En el presente trabajo de tesis se desarrolló un sistema de navegación con GPS, usando sensores inerciales y magnéticos de bajo costo, también se desarrolló un sistema de monitoreo ambiental el cual consta de sensores analógicos y digitales todo controlado con un FPGA (del inglés, Field Programmable Gate Array) utilizando el lenguaje de descripción de Hardware VHDL. El sistema de monitoreo ambiental contiene los siguientes sensores: calidad del aire, índice de rayos UV, temperatura, porcentaje de humedad relativa y presión atmosférica, todos colocadas dentro de un sistema de protección contra contingencias. Un FPGA spartan 6 se comunica con todos estos sensores y envía los datos mediante comunicación serial con la ayuda de un módem bluetooth a una computadora en un lugar fijo, a su vez la computadora muestra estos datos en una interfaz gráfica diseñada en LabVIEW.

EL sistema de navegación con GPS para vehículos no tripulados, consta de la implementación de un sistema de referencia de actitud y rumbo AHRS (del inglés, Attitude Heading and Reference System) el cual está compuesto de sensores inerciales (acelerómetros y girometros), sensores magnéticos de bajo costo y un algoritmo de integración de datos desarrollado en el FPGA. Este sistema AHRS sirve para estimar la orientación de un cuerpo rígido con respecto a un sistema de referencia inercial. Debido a que la orientación no puede ser medida directamente se debe hacer una estimación integrando las señales de los sensores inerciales, girometros, magnetómetros y acelerómetros. El problema del cálculo de la orientación consiste en dos partes fundamentales. La primera parte consiste en estimar la orientación, por medio de vectores de observación (medidas de los acelerómetros y girometros), esto se realiza utilizando un algoritmo de proyección, el cuaternión que se obtiene se puede considerar como una orientación medida pero que se encuentra contaminada por ruido. La segunda parte consiste en un observador no lineal a fin de producir la estimación de un cuaternión obtenido mediante la medida arrojada por los girometros y la integración numérica de la ecuación cinemática asociada al cuerpo rígido. Esta estimación del cuaternión es comparada con el cuaternión obtenido mediante el algoritmo de proyección, el error es retroalimentado nuevamente en el observador no lineal para así obtener el cuaternión de orientación.

Para la elección del FPGA como controlador de señales y procesador de las mismas se consideró su velocidad ya que puede trabajar fácilmente hasta 500 MHz (aunque en este caso utilizamos un reloj de 12 MHz), así como su arquitectura abierta ya que esto abre la posibilidad de seguir añadiendo sensores en el futuro para que nuestros sistemas sean más robustos. En el caso de los sensores seleccionados se consideró que tuvieran un voltaje de alimentación idéntico o que en su caso tuvieran integrados un regulador de voltaje para que se utilizara una misma fuente de voltaje.

En el marco de esta tesis, se desarrollaron dos interfaces gráficas para la visualización de los datos procesados o enviados por los sensores, dichas interfaces se desarrollaron en LabVIEW debido a su gran variedad de herramientas de instrumentación.

Índice general

Resumen	III
Agradecimientos	XIII
Lista de acrónimos	xv
Introducción	xvii
Objetivos	xix
1. Generalidades y estado del arte	1
1.1. Monitoreo ambiental	1
1.1.1. Antecedentes	1
1.1.2. Características de los sistemas de monitoreo	1
1.1.3. Aplicaciones de los sistemas de monitoreo ambiental	3
1.1.4. Desafíos en el monitoreo ambiental	3
1.2. Sistemas de referencia de actitud y rumbo (AHRS)	4
1.2.1. AHRS tradicionales	5
1.2.2. Ventajas y desventajas de los AHRS tradicionales	5
1.2.3. Aplicaciones	6
1.3. AHRS Strapdown	6
1.3.1. Ventajas y desventajas de los AHRS strapdown	7
1.3.2. Aplicaciones de los AHRS strapdown	8
1.4. Algoritmos para la estimación de la actitud y rumbo	8
1.4.1. Parametrización de la actitud y rumbo en término de cuaterniones	8
1.4.2. Estimación de la actitud y rumbo usando Filtros de Kalman	10
1.4.3. Estimación de la actitud y rumbo usando observadores no lineales	12
1.5. AHRS Strapdown comerciales	12
2. Sensores utilizados en el monitoreo ambiental y navegación	13
2.1. Sensores inerciales MEMS	13
2.1.1. Acelerómetros MEMS	14
2.1.2. Girómetros MEMS	15
2.2. Sensor Magneto-resistivo	16
2.3. Sensores utilizados en el monitoreo ambiental	16

2.3.1.	Sensor de humedad	17
2.3.2.	Sensor de calidad del aire	17
2.3.3.	Sensor de temperatura	18
2.3.4.	Sensor de rayos Ultravioleta	19
2.3.5.	Sensor de presión atmosférica	19
2.4.	GPS	20
2.4.1.	Antecedentes	20
2.4.2.	Características	20
2.4.3.	Aplicaciones del GPS	21
3.	Diseño e implementación del sistema de navegación	23
3.1.	Cuaterniones unitarios y cinemática de actitud	23
3.2.	Sensores de velocidad angular	25
3.3.	Sensores de vectores de referencia	25
3.4.	Planteamiento del problema	26
3.5.	Estimación de la actitud usando el vector de observación.	27
3.5.1.	Observador no lineal de la orientación	27
3.6.	Integración numérica usando Runge Kutta de orden 4	28
3.6.1.	Integración por RK4 de la ecuación cinemática del cuaternión	29
3.6.2.	Implementación del Runge Kutta en FPGA	30
3.7.	Unidad de Mediciones Inerciales (IMU)	31
3.8.	Procesador digital de señales	32
3.9.	Instrumentación de módulo receptor GPS	33
3.10.	Sistema de comunicación entre sistema de navegación y estación terrestre	33
3.11.	Batería de polímero de litio	35
3.12.	Desarrollo de interfaz gráfica	36
3.12.1.	Integración de interfaz	38
3.13.	Integración del sistema de navegación	39
3.13.1.	Adquisición de datos de los sensores	40
3.13.2.	Computo del cuaternión de orientación	42
3.13.3.	Diseño del PCB	43
3.14.	Resultados experimentales	44
4.	Implementación de sistema de monitoreo	49
4.1.	Integración del sistema de monitoreo ambiental	50
4.1.1.	Adquisición de los datos de los sensores	51
4.1.2.	Comunicación con la estación de monitoreo	52
4.1.3.	Diseño del PCB y ensamble	53
4.2.	Instrumentación de sensor de humedad y temperatura	54
4.3.	Instrumentación de sensor de calidad del aire	56
4.4.	Instrumentación de sensor de rayos UV	57
4.5.	Instrumentación de sensor de presión atmosférica	58
4.6.	Descripción del módulo Bluetooth utilizado	58
4.7.	Batería	59

4.8. Desarrollo de interfaz gráfica	60
4.9. Resultados experimentales	60
Conclusiones	65
A. Cuaterniones	69
A.1. Álgebra de cuaterniones	69
A.1.1. Suma de cuaterniones	69
A.1.2. Resta de cuaterniones	70
A.1.3. Multiplicación de cuaterniones	70
A.1.4. Conjugación de cuaterniones	71
A.1.5. División de cuaterniones	71
A.1.6. Exponenciación de cuaterniones	72
A.1.7. Cuaterniones a ángulos de Euler	72
B. Protocolos de comunicación	75
B.0.8. Protocolo RS-232	75
B.0.9. Protocolo I^2C	76

Índice de figuras

1.1. Estación metereologica.	2
1.2. Ángulos de alabeo, cabeceo y guiñada	4
1.3. Sistema de referencia inercial NED.	4
1.4. AHRS Tradicional.	5
1.5. Aplicaciones de AHRS.	6
1.6. AHRS strapdown.	7
1.7. AHRS's strapdown de la empresa MEMSIC.	7
1.8. Aplicaciones de AHRS strapdown.	8
1.9. Esquema de las componentes de un cuaternión.	9
1.10. Esquema de las componentes de la velocidad angular $\vec{w}$ del cuerpo rígido.	10
2.1. Imagen de un dispositivo MEMS a la izquierda y la diferencia de tamaño de un ácaro y el dispositivo a la derecha	13
2.2. Estructura básica de un acelerómetro de tecnología MEMS	14
2.3. Fuerza de Coriolis	15
2.4. Magnetómetro magneto-resistivo.	16
2.5. Sensor de humedad por deformación	17
2.6. Funcionamiento de sensor de gas por conductividad térmica	18
2.7. Funcionamiento de sensor de rayos UV	19
2.8. Satélite Vanguard puesto en órbita en marzo de 1958	20
2.9. Funcionamiento de GPS	21
2.10. Aplicaciones del GPS.	22
3.1. Diagrama de integración de AHRS.	23
3.2. Diagrama de integración de AHRS.	28
3.3. Módulo para calcular Runge-Kutta	30
3.4. IMU AltIMU-10.	31
3.5. Xula2 con Spartan 6	32
3.6. GPS-635T.	33
3.7. Dispositivos utilizados para la comunicación	34
3.8. Diferentes antenas utilizadas en XBee	35
3.9. Batería de LiPo	36
3.10. Regulador de voltaje	36
3.11. Diagrama a bloques de la estación de monitoreo	37
3.12. Interfaz Gráfica	37

3.13. Diagrama a bloques del sistema de navegación	40
3.14. Diagrama de flujo del software	41
3.15. Diagrama de flujo del software	42
3.16. Diseño de PCB	43
3.17. PCB	43
3.18. Diagrama a bloques en MatLab del observador no lineal	44
3.19. Gráfica del error estimado	45
3.20. Gráfica comparativa entre el cuaternión estimado de color morado y el cuaternión real de color azul	45
3.21. Advertencia en el compilador	46
3.22. Comparación q_0 teórico y practico	46
3.23. Comparación q_2 teórico y practico	47
3.24. Imagen de ubicación dada por el GPS	47
4.1. Estación de monitoreo ambiental portátil de la empresa Kestrelmeters	50
4.2. Integración del sistema de monitoreo	50
4.3. Diagrama de flujo del sistema de monitoreo	51
4.4. Diseño de PCB para sistema de monitoreo ambiental.	53
4.5. Sistema de monitoreo ambiental	53
4.6. Sistema de monitoreo ambiental	54
4.7. Sensor Humedad y temperatura.	54
4.8. Inicio de comunicación con DHT11.	55
4.9. Bit a uno enviado por el DHT11	56
4.10. Bit a cero enviado por el DHT11	56
4.11. Sensor de Calidad de Aire	57
4.12. Comportamiento de sensor de rayos Uv.	57
4.13. Sensor de rayos Uv.	58
4.14. Sensor de presión atmosférica.	58
4.15. Modulo Bluetooth.	59
4.16. Batería de LiPo de sistema de monitoreo.	59
4.17. Interfaz gráfica del sistema de monitoreo	60
4.18. Hexarotor	61
4.19. Prueba de funcionamiento	61
4.20. Datos recolectados de Internet	62
4.21. Datos recolectados por la estación meteorológica	63
B.1. Protocolo de comunicación serial asíncrono	76
B.2. Conexión I^2C típica	77
B.3. Transmisión de datos I^2C	77

Índice de tablas

1.1. Especificaciones técnicas de MTi y 3DM-GX3-25 (ver [17] y [18] para mas detalles)	12
3.1. Especificaciones técnicas de la IMU AltIMU-10	31
3.2. Especificaciones técnicas FPGA Spartan 6 XL-25 [23]	32
3.3. Especificaciones técnicas de la tarjeta de desarrollo XULA-2 [24]	33
3.4. Especificaciones técnicas diferentes series de XBee [29]	34
3.5. Especificaciones técnicas de la batería de LiPo [26]	36
3.6. Requerimientos mínimos para ejecutar LabVIEW	38
3.7. Envío y recepción de datos de la interfaz	39
4.1. Transmisión de datos del sistema de monitoreo ambiental	52
4.2. Especificaciones técnicas de la batería de LiPo [26] para el sistema de monitoreo	60

Agradecimientos

Gracias a Dios por su fidelidad y misericordia que me llevó más lejos de lo que pude haber imaginado, a mis Padres por su amor , comprensión y sus incontables oraciones, sin ustedes no hubiera sido posible, a mis hermanas por soportarme, sé que a veces suelo ser “hiper-odioso”, a mi familia en general por su cariño y aprecio, a mis amigos que tuvieron que aguantar mi necedad durante la carrera, a mis asesores por la paciencia y esta gran oportunidad, pues aprendí mucho durante la realización de este proyecto. En fin, gracias a todos los que me apoyaron y se acordaron de mi en sus oraciones.

A Dios, la gloria y el honor.

Lista de acrónimos

AHRS: Attitude and Heading Reference System.

DC: Direct Current

GPS: Global Positioning System.

INECC: Instituto Nacional de Ecología y Cambio Climático.

NED: North East Down.

IMU: Inertial Mesure Unit.

MEMS: Micro-Electro-Mechanical Systems. KF: Filtro de Kalman.

EKF: Filtro de Kalman extendido.

KF: Filtro de Kalman.

UHF: Frecuencias Ultracortas.

FPGA: Field Programmable Gate Array.

VHDL: Very High Integrated Circuit and Hardware Description Language.

RS-232: Recommended Standar 232.

UAV: Unmanned aerial vehicle.

VANT: Vehículo Aéreo No Tripulado.

Introducción

La tendencia a nivel mundial, es sin duda dar mayor importancia a problemas ambientales, sobre todo aquellos relacionados con energías renovables. Por ello, la monitorización del medio ambiente es fundamental para no agotar nuestros recursos y asegurar un mejor futuro [1].

La monitorización del medio ambiente en México se remonta al año 1901 cuando se forma el observatorio meteorológico nacional, es entonces cuando se instaló la primera estación meteorológica en México y así se inició el monitoreo ambiental en nuestro país. Comúnmente para monitorear el medio ambiente se utilizan estaciones meteorológicas, que están destinadas a medir parámetros ambientales, por ejemplo: temperatura, presión atmosférica, humedad, calidad del aire entre otros parámetros diversos dependiendo de las necesidades que tenga el usuario. Hoy en día existen una gran variedad estaciones meteorológicas, las cuales tienen un precio desde 26000.00 MN, hasta los 35000.00 MN [2] aproximadamente, lo cual hace que el monitoreo ambiental sea costoso. Las medidas de estas estaciones meteorológicas son de aproximadamente 300 mm x 215 mm x 300 mm y pesan poco más de 2 kilogramos lo que hace un poco inconveniente su portabilidad, teniendo en cuenta que se necesita medir parámetros ambientales a distintas alturas y en diversos lugares para el estudio de un lugar. Con la aparición de nuevos sensores cada vez más pequeños y con un costo accesible, se abre la posibilidad de crear estaciones de monitoreo ambiental más compactas y baratas, también existen dispositivos programables muy eficientes y pequeños que podrían ayudar al procesamiento de estos datos. Además con los vehículos aéreos no tripulados cada vez más variados se podrían tomar datos a distintas alturas y lugares de una manera más fácil y rápida.

La navegación puede definirse como el conjunto de técnicas utilizadas para desplazarse entre un par de puntos conocidos, “origen y destino” siguiendo una trayectoria. En un principio, este término estuvo asociado a los barcos pero el desarrollo de la aviación le agrego una tercera dimensión; además de la posición horizontal (latitud y longitud), se necesita también la altura. Entonces a este tipo de navegación se le conoce como navegación 3D. El desarrollo de Sistemas Micro-electromecánicos (MEMs) ha hecho posible crear sistemas de navegación empotrados; la ventaja de estos sistemas es que son pequeños y tienen bajo consumo de energía [3]. En la actualidad, existen módulos conocidos como “Attitude Heading Reference System” que se identifica por la siglas AHRS y tienen como función principal estimar la orientación y postura de un vehículo con respecto a un sistema de referencia inercial (Norte, Este, Abajo). El AHRS está compuesto por sensores inerciales; un girómetro con tres ejes sensibles matricula L3GD20, acelerómetro con tres ejes sensibles matricula LSM303DLHC y un sensor magnético con tres ejes sensibles matricula LSM303DLHC [32] dispuestos de tal forma que coincidan con los ejes de rotación del vehículo donde esté colocado [5]. La fusión de los datos de estos sensores se hace

regularmente con el algoritmo matemático llamado: filtro de Kalman extendido [4], el cual es solo una ampliación del filtro de Kalman [5] ordinario pero para sistemas no lineales. En este caso, se propone trabajar con un observador no lineal, este observador integrará la señal proveniente de los tres sensores estableciendo una medida fiable de la actitud y rumbo. También se hará uso de un GPS el cual nos dará información de la posición en la que se encuentra el vehículo, el GPS o “Sistema de posicionamiento global” proporciona la localización y orientación mediante coordenadas geográficas latitud y longitud [6]. El presente trabajo consiste en la creación de un sistema de navegación y monitoreo ambiental para vehículos autónomos con base en una tarjeta de programación FPGA con un Spartan 6 llamada XulA2, la cual será programada en VHDL y ésta procesará todos los datos obtenidos de los sensores y los enviará por vía RS232 mediante un dispositivo llamado Xbee serie 2 de la familia de digilent el cual utiliza un protocolo de comunicación Zigbee por radiofrecuencia que es capaz de comunicarse con otro dispositivo similar a 100 metros de distancia, que a su vez se conectará a una computadora para visualizar y guardar todos los datos enviados. Cada uno de los sensores estará conectado al FPGA mediante su respectivo protocolo de comunicación, se tendrá un girómetro, acelerómetro, barómetro y magnetómetro, para el diseño del sistema de navegación del vehículo autónomo y para el monitoreo ambiental se ocupará un sensor de temperatura y humedad, un sensor de presión atmosférica, un sensor de rayos ultravioleta y un sensor de calidad del aire. El presente trabajo de tesis es parte de un proyecto más general que consiste en la implementación de un sistema control de formación de múltiples vehículos aéreos no tripulados mediante control cooperativo: aplicación monitoreo ambiental.

Este trabajo está organizado de la siguiente forma: el capítulo 1, presenta las generalidades y el estado del arte; empezando por los sistemas de monitoreo ambiental, sus antecedentes, características y la tendencia de ellos, también se presenta una descripción de los sistemas AHRS, su principio de funcionamiento y algunas de sus aplicaciones. El capítulo 2, aborda el principio de funcionamiento de cada sensor utilizado en este trabajo, así como sus especificaciones técnicas. En el tercer capítulo, se describe todo el desarrollo del sistema de navegación, iniciando con la descripción del algoritmo matemático y culminando con su síntesis en VHDL, así como los resultados experimentales de esta parte. El capítulo 4, describe la implementación del sistema de monitoreo ambiental, la integración de sus componentes, la programación utilizada y los resultados experimentales. Finalmente, se presenta las conclusiones obtenidas de este trabajo.

Objetivos

Objetivo general:

- Implementar un algoritmo para el sistema de navegación y monitoreo ambiental que pueda ser aplicado en vehículos autónomos terrestres o aéreos utilizando como procesador principal un sistema FPGA.

Objetivos específicos:

- Realizar la programación del dispositivo FPGA Xula2 que contiene un Spartan 6 de la familia Xilinx en VHDL para habilitar la comunicación con el módulo AltIMU-10 el cual contiene sensores inerciales, magnéticos y de presión que permitirá implementar el sistema de navegación inercial (INS). Así mismo, será necesario caracterizar todos los sensores a ocupar.
- Realizar la programación del dispositivo FPGA XULA2 en VHDL para habilitar la comunicación con el módulo GPS GP-635T y que permitirá implementar el sistema de navegación vía GPS.
- Hacer la programación del dispositivo FPGA XULA2 en VHDL para habilitar los módulos Xbee y establecer así la comunicación mediante protocolo Zigbee.
- Diseñar la interfaz de usuario que permita visualizar y analizar las variables del sistema.
- Diseñar y construir un prototipo electrónico que contenga los sensores y la tarjeta de comunicación de datos para probar y depurar el algoritmo en la PC.
- Desarrollar un montaje seguro respecto a contingencias terrestres o aéreas que protejan al sistema.

Capítulo 1

Generalidades y estado del arte

En este primer capítulo se presentan definiciones y conceptos que se utilizarán a lo largo de la tesis. Se comienza con la descripción de los sistemas de monitoreo ambiental convencionales, además se mencionan algunas de sus características así como sus aplicaciones. Se continua con los sistemas de navegación AHRS y sus antecedentes. Finalmente, se describe el funcionamiento del GPS, sus aplicaciones, así como sus ventajas y desventajas.

1.1. Monitoreo ambiental

1.1.1. Antecedentes

En México, en el año de 1992 fue creado el Instituto Nacional de Ecología con atribuciones técnicas y normativas en materia de ecología. El cual en el año 2012 pasa a ser el Instituto Nacional de Ecología y Cambio Climático (INECC), este instituto tiene como una de sus funciones monitorear el medio ambiente y preservarlo. Aunque este instituto se creó hace dos décadas, el monitoreo ambiental tiene antecedentes más longevos en nuestro país, exactamente con las estaciones meteorológicas, lo cual se remonta a 1901 cuando se instaló la primera estación meteorológica en México.

En 1849, la Institución Smithsonian comenzó a establecer una red de observación a través de los Estados Unidos bajo la dirección de Joseph Henry. Redes de observación similares fueron establecidas en Europa en ese momento [7].

1.1.2. Características de los sistemas de monitoreo

Los sistemas de monitoreo ambiental pueden tener distintos y variados sensores medio ambientales, en el caso de las estaciones meteorológicas normalmente utilizan los siguientes sensores medio ambientales:

- **Anemómetro:** mide la velocidad del viento
- **Veleta:** señala la dirección del viento
- **Barómetro:** mide la presión atmosférica.
- **Heliógrafo:** mide la insolación recibida de la superficie terrestre.
- **Higrómetro:** mide la humedad.
- **Piranómetro:** mide la radiación solar.
- **Pluviómetro:** mide el agua caída.
- **Termómetro:** mide la temperatura.

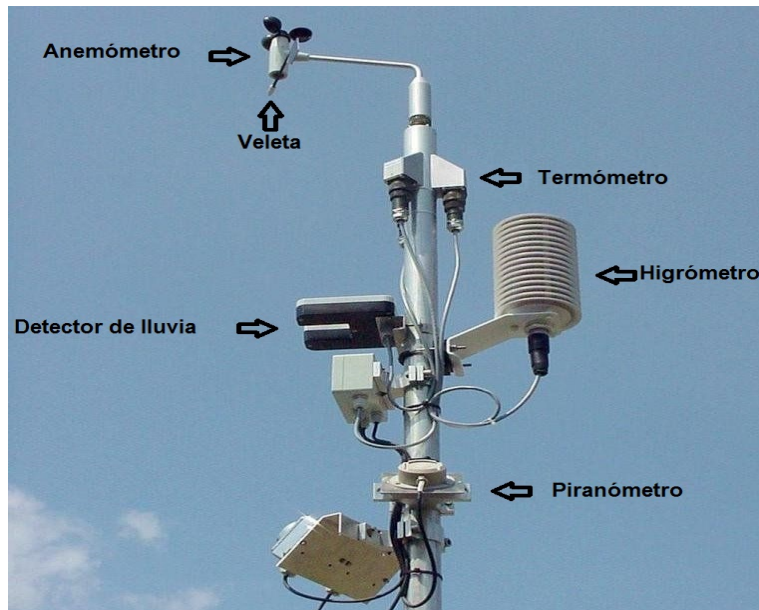


Figura 1.1: Estación metereologica.

En la figura 1.1, se muestra la imagen de una estación meteorológica, estas estaciones meteorológicas pueden tener distintas dimensiones dependiendo de la fiabilidad de los sensores y las aplicaciones requeridas, se pueden agregar más sensores de los ya mencionados; aunque a nuestro conocimiento hasta ahora no se ha explorado la aplicación en vehículos autónomos.

1.1.3. Aplicaciones de los sistemas de monitoreo ambiental

Con la investigación y desarrollo en energías renovables se abre un nuevo campo para aplicar los sistemas de monitoreo, ya que las energías renovables requieren de un estudio del área en donde se desarrollaran para tener mayor eficiencia. Algunos de los aspectos que se consideran para el desarrollo de energías renovables son :

- **Radiación solar:** Es el conjunto de radiaciones electromagnéticas emitidas por el Sol. En este caso se hace referencia a los paneles solares ya que estos producen electricidad a partir la radiación solar que se incide sobre ellos.
- **Temperatura:** En este caso la temperatura afecta al funcionamiento de las celdas solares ya que el voltaje en circuito abierto de una celda disminuye si la temperatura aumenta.
- **Humedad:** En este caso es importante este factor ya que los sistemas de generación de energía renovable pueden ser afectados por la corrosión.
- **Presión atmosférica:** En el caso de energía eólica es necesario saber la presión atmosférica en donde se colocan los generadores ya que la energía del viento está relacionada con el movimiento de las masas de aire que se desplazan desde zonas de alta presión atmosférica hacia zonas adyacentes de menor presión .

Estos son algunas de las consideraciones que se deben de tener para el uso de energías renovables, aunque no son todas; se considera que estas son las más importantes. Otra de las aplicaciones emergentes es el Internet de las cosas que se refiere a la interconexión de objetos en una red. Un ejemplo claro son los “ Smartphone´s”que recopilan información para que el usuario planifique bien su día a día; una de estas informaciones es el estado del tiempo. Este tipo de información es muy útil para todas las personas ya que pueden consultar la temperatura, calidad del aire, presión atmosférica e irradiancia solar. Así, el monitoreo ambiental es muy utilizado también en este rubro. Estos sistemas son utilizados para medir cambios climatológicos en distintas zonas y definir que tan afectadas han sido por el paso del tiempo.

1.1.4. Desafíos en el monitoreo ambiental

Algunos los desafíos que se tienen al trabajar con los sistemas de monitoreo actuales son:

- *Reducir costos*
Los costos de los sistemas de monitoreo ambiental convencionales; en el caso de estaciones meteorológicas estan entre los 26000.00 MN, hasta los 35000.00 MN

■ *Tamaño*

Los sistemas de monitoreo ambiental comerciales son muy grandes y dependen de una toma de corriente por lo que son difíciles de trasladar. Actualmente, en México no se disponen de sistemas de monitoreo ambiental portátiles que sean comerciales. Esto representa un problema en nuestro país ya que se requieren de este tipo de sistemas ya sea para el estudio o preservación del medio ambiente y en el desarrollo de energías renovables.

1.2. Sistemas de referencia de actitud y rumbo (AHRS)

En la navegación la medida precisa de orientación (actitud y rumbo) con respecto a un sistema de referencia inercial es fundamental; ya que esta auxilia al operador para mantener la navegación, sistema conocido como horizonte artificial; en el caso de vehículos aéreos. En el caso de los vehículos autónomos se debe conocer la orientación a cada instante para que el control pueda integrar todas las señales que se enviarán a los actuadores con el fin de mantener la orientación deseada.

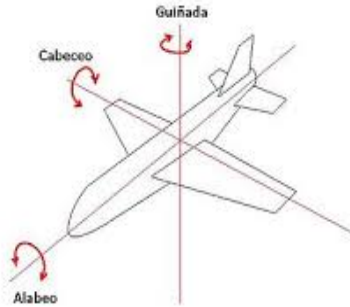


Figura 1.2: Ángulos de alabeo, cabeceo y guiñada .

En la figura 1.2 se muestran los ángulos de Euler usados para describir la orientación de un objeto en tres dimensiones.

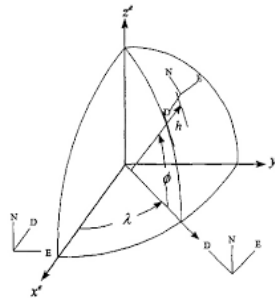


Figura 1.3: Sistema de referencia inercial NED.

Un sistema de Referencia de Actitud y Rumbo (AHRS) es un sistema que estima la orientación de un objeto con respecto a un sistema de referencia inercial. En la figura 1.3 se muestra un sistema de referencia Norte, Este, Abajo, que es el más utilizado en sistemas AHRS.

Los AHRS se componen por los siguientes sensores: acelerómetros que miden aceleración lineal, girómetros que miden velocidad angular y magnetómetros que miden intensidad de campo magnético todos ellos en el sistema de referencia móvil pero con respecto al sistema de referencia inercial. Cada sensor mide en los tres ejes del sistema de referencia inercial y se colocan de manera ortogonal de tal manera que los ejes coincidan con los ejes de rotación del vehículo donde se utilice [4]. Al conjunto de girómetros y acelerómetros se le conoce como Unidad de Mediciones Inerciales (IMU por sus siglas en inglés Inertial Measurement Unit).

1.2.1. AHRS tradicionales

Los primeros AHRS hicieron posible la navegación autónoma y su aparición fue durante la segunda guerra mundial aplicados en las bombas V1 y V2 [8]. Estos son sistemas puramente mecánicos formados principalmente por: gimbales, acelerómetros, girómetros mecánicos estos se encuentran alineados con el sistema de referencia NED. Para que se mantenga esta orientación, los sensores inerciales son aislados de cualquier movimiento exterior mediante un conjunto de tres gimbales que son giro-estabilizados. Los gimbales son estructuras en forma de anillos, anidadas entre si y sus extremos tienen rodamientos que están conectados a un servomotor en un extremo y en el otro a un encoder. Por medio de estos encoder se puede conocer el rumbo y la actitud del vehículo, cada encoder mide un ángulo diferente.

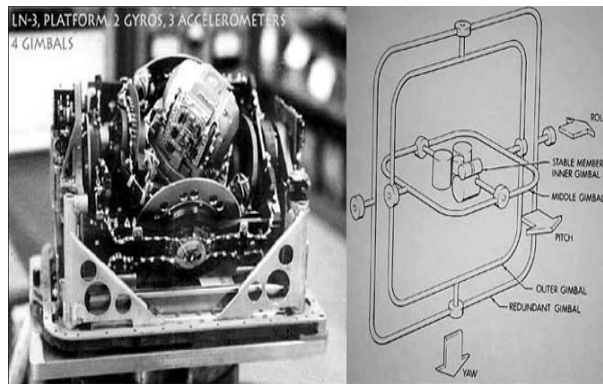


Figura 1.4: AHRS Tradicional.

1.2.2. Ventajas y desventajas de los AHRS tradicionales

Ventajas:

- Aislamiento de los sensores inerciales de altas velocidades angulares, lo que elimina muchos errores de los sensores dependientes de la velocidad y generalmente permiten a los sensores una mayor exactitud.

Desventajas:

- Son sistemas mecánicos complejos
- Son costosos
- Son pesados
- Tienen un alto consumo de energía
- Deben ser calibrados constantemente
- Sufren desgaste mecánico.

1.2.3. Aplicaciones

Las aplicaciones que tienen los AHRS tradicionales es en aeronaves, barcos, submarinos, naves espaciales y misiles. En la figura 1.5 se muestran algunos ejemplos de las aplicaciones de los AHRS tradicionales.



Figura 1.5: Aplicaciones de AHRS.

1.3. AHRS Strapdown

Los sistemas AHRS strapdown están formados principalmente por: un procesador más los sensores inerciales como se aprecia en la figura 1.6. En este tipo de sistemas, los acelerómetros y girómetros no están aislados de las rotaciones del vehículo como sucede en los sistemas AHRS tradicionales[9]. Por lo tanto, los sensores son expuestos a altas velocidades de giro, lo cual provoca errores en los sensores dependientes de la orientación y velocidad. Esto se debe principalmente a que estos sistemas están montados rígidamente dentro del vehículo, alineados con los ejes de giro.

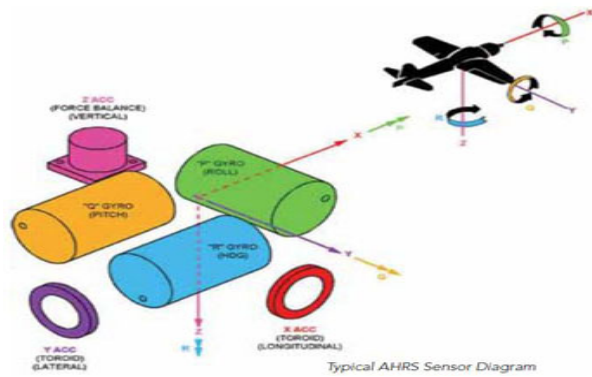


Figura 1.6: AHRS strapdown.

Los sistemas de plataforma estable y los sistemas strapdown están basados en los mismos principios. Los sistemas strapdown han reducido la complejidad mecánica y tienden a ser más pequeños que los sistemas de plataforma estable. Esto se debe principalmente a que los gimbales han sido remplazados por software, que implica un mayor costo computacional al hacer la estimación de la orientación pero es más barato que el uso de gimbales. Como el costo de los procesadores y sistemas embebidos ha disminuido, los sistemas strapdown se han convertido en los AHRS dominantes en el mercado.

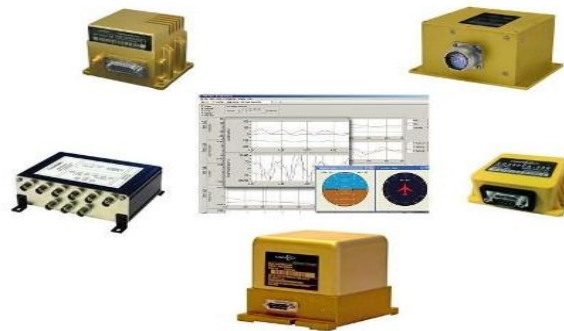


Figura 1.7: AHRS's strapdown de la empresa MEMSIC.

1.3.1. Ventajas y desventajas de los AHRS strapdown

A continuación se muestran las ventajas y desventajas de los sistemas AHRS strapdown.

Ventajas:

- Son más baratos que los sistemas de gimbales
- En aplicaciones donde se requiere
- Son pequeños
- No son pesados
- Tienen un bajo consumo de energía

- No presentan desgaste mecánico

Desventajas:

- Los sensores deben operar a velocidades de rotación más grandes en comparación con los usados en sistemas gimbales.
- La entrada dinámica de los girómetros strapdown deben de ser de magnitudes más grandes que la de los girómetros de los sistemas de gimbales.
- Aunque son más económicos que los AHRS tradicionales su costo aún es elevado para el publico en general.

1.3.2. Aplicaciones de los AHRS strapdown

Los AHRS strapdown tienen las mismas aplicaciones que los AHRS tradicionales. Sin embargo, los AHRS strapdown basados en tecnología MEMS tienen un campo de aplicación más amplio. Dentro de estas aplicaciones podemos mencionar: vehículos no tripulados, plataformas de estabilización, robótica asistida, juegos de vídeo, captura de movimientos, biomecánica, etc. En la figura 1.8 se muestran algunas de las aplicaciones de los AHRS.



Figura 1.8: Aplicaciones de AHRS strapdown.

1.4. Algoritmos para la estimación de la actitud y rumbo

Los AHRS strapdown son en realidad estimadores de la orientación, ya que esta no puede ser medida directamente y debe ser estimada usando diferentes tipos de sensores tales como: acelerómetros, girómetros y magnetómetros. Estas señales nos ayudan para estimar la orientación, la cual se hace por medio de algoritmos matemáticos de estimación programados en microprocesadores o microcontroladores.

1.4.1. Parametrización de la actitud y rumbo en término de cuaterniones

La orientación de cualquier cuerpo rígido puede ser representada por medio de cuaterniones. Esta representación consiste en un vector unitario $\vec{e}$ y un ángulo de rotación β alrededor de dicho eje, tal como se muestra en la figura 1.9 .

El cuaternión q está definido como sigue

$$q = \begin{bmatrix} \cos \frac{\beta}{2} \\ \vec{e} \sin \frac{\beta}{2} \end{bmatrix} = \begin{bmatrix} q_0 \\ \vec{q} \end{bmatrix} \in H \quad (1.1)$$

donde

$$H = \left\{ q \mid q_0^2 + \vec{q}^T \vec{q} = 1, q = \begin{bmatrix} q_0 & \vec{q} \end{bmatrix}^T, q_0 \in \mathbb{R}, \vec{q} \in \mathbb{R}^3 \right\} \quad (1.2)$$

$\vec{q} = [q_1 \ q_2 \ q_3]^T$ y q_0 son las partes vectorial y escalar del cuaternión respectivamente.

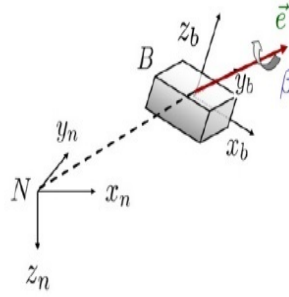


Figura 1.9: Esquema de las componentes de un cuaternión.

En aplicaciones de control de orientación, el cuaternión unitario representa la rotación de un sistema de coordenadas inercial $N(x_b, y_b, z_b)$ localizadas en el centro de masa de un cuerpo rígido. Donde B es un sistema de referencia móvil.

El producto de cuaterniones se define como:

$$q_1 \otimes q_2 = \begin{bmatrix} q_{10} & -\vec{q}_1^T \\ \vec{q}_1 & I_3 q_{10} + [\vec{q}_1^\times] \end{bmatrix} \begin{bmatrix} q_{20} \\ \vec{q}_2 \end{bmatrix} \quad (1.3)$$

donde $\otimes$ denota la multiplicación de cuaterniones, I_3 es la matriz identidad y $[\vec{\xi}^\times]$ es el tensor antisimétrico asociado con el vector axial $\vec{\xi}$, y esta dado por

$$[\vec{\xi}^\times] = \begin{bmatrix} \xi_1 \\ \xi_2 \\ \xi_3 \end{bmatrix}^\times = \begin{bmatrix} 0 & -\xi_3 & \xi_2 \\ \xi_2 & 0 & -\xi_1 \\ -\xi_2 & \xi_1 & 0 \end{bmatrix} \quad (1.4)$$

El error de orientación es utilizado para cuantificar el desfase entre dos orientaciones. Si q define la orientación verdadera y $\hat{q}$ es el cuaternión estimado, entonces el cuaternión de error que representa el error de orientación verdadera y la estimada viene dado por

$$q_e = \hat{q}^{-1} \otimes q = \begin{bmatrix} q_{e0} & \vec{q}_e^T \end{bmatrix}^T \quad (1.5)$$

donde $\hat{q}^{-1}$ es la rotación complementaria del cuaternión $\hat{q}$, el cual es el cuaternión conjugado.

En el caso de que el cuaternión verdadero y el cuaternión de error se convierte en

$$q_e = \begin{bmatrix} \pm 1 & 0^T \end{bmatrix}^T \quad (1.6)$$

Esto se debe al hecho de que q y $-q$ representan la misma orientación física. En realidad, uno se encuentra girado 2π relativamente al otro alrededor del vector unitario $\vec{e}$.

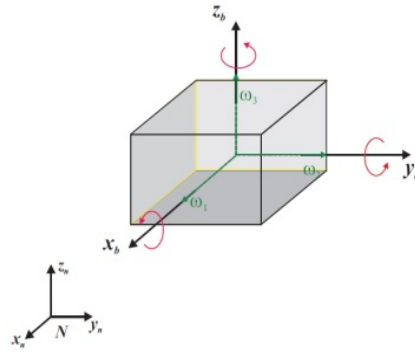


Figura 1.10: Esquema de las componentes de la velocidad angular $\vec{w}$ del cuerpo rígido.

Denotando con $\vec{\omega} = [\omega_1 \omega_2 \omega_3]^T$ el vector de velocidad angular del marco de referencia en el cuerpo B relativo al marco inercial N , se muestra en la figura 1.10 la ecuación cinemática está dada por:

$$\begin{bmatrix} \dot{q}_0 \\ \vec{\dot{q}} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} -\vec{q}_1^T \\ I_3 q_0 + [\vec{q}_1 \times] \end{bmatrix} \vec{\omega} = \frac{1}{2} \Xi(q) \vec{\omega} \quad (1.7)$$

1.4.2. Estimación de la actitud y rumbo usando Filtros de Kalman

Uno de los principales usos de los observadores es la estimación de variables de estado de un sistema en el cual las medidas están contaminadas con ruido. Considere un sistema lineal discreto con la siguiente dinámica:

$$x_{k+1} = Ax_k + Bu_k + w_k \quad (1.8)$$

$$y_k = Cx_k + v_k \quad (1.9)$$

En las ecuaciones anteriores A , B y C son matrices, k es el índice de tiempo, x_k es llamado el estado del sistema, u_k es una entrada conocida del sistema, y_k es la salida medida, w_k y z_k

son los ruidos blancos Gaussianos de media cero. La variable w_k se llama el ruido del proceso y la variable z_k se llama ruido de la medición. El vector x_k contiene toda la información sobre el estado actual del sistema, pero en general no puede ser medido directamente. En su lugar, medimos y_k que es una función de x_k y que está dañada por el ruido z_k y por medio de ésta se estima x_k . Por lo anterior, w_k y z_k satisfacen:

$$E(w_k) = 0 \quad E(z_k) = 0 \quad (1.10)$$

$$E(w_j w_k^T) = S_w \quad \text{para } j = k \text{ y cero en otro caso} \quad (1.11)$$

$$E(z_j z_k^T) = S_z \quad \text{para } j = k \text{ y cero en otro caso} \quad (1.12)$$

S_w y S_z son las matrices de covarianza para la perturbación w en el proceso y el ruido z en la media. El observador óptimo que permitirá estimar el estado x está dado por las siguientes ecuaciones y es conocido como el Filtro Kalman (KF):

$$\hat{x}_{k+1} = (A\hat{x}_k + Bu_k) + K_k(y_{k+1} - C\hat{x}_k) \quad (1.13)$$

$$K_k = AP_k C^T (C P_k C^T + S_z)^{-1} \quad (1.14)$$

$$P_{k+1} = AP_k A^T + S_w - AP_k C^T S_z^{-1} C P_k A^T \quad (1.15)$$

Donde $\hat{x}$ es el vector de estados estimado, K es la matriz de covarianza del error de estimación ($\hat{x} - x$). El KF provee un eficiente método computacional para combinar señales de salida de sensores contaminados por ruido para estimar el estado de un proceso lineal estocástico, bajo la suposición que el ruido que afecta las medidas es blanco y presenta una distribución Gaussiana. Debido a que en la naturaleza la mayoría de los sistemas son no lineales, la condición de linealidad no puede ser satisfecha todo el tiempo y ni para todas las aplicaciones. La estimación de la actitud y rumbo es un ejemplo de cuando el sistema es no lineal debido a las ecuaciones involucradas.

Para el caso de los sistemas no lineales se puede usar el Filtro de Kalman Extendido (EKF), el cual es una ampliación del Filtro de Kalman ordinario [5]. En el sistema en el que se diseña un EKF se asume que todos los componentes del ruido tienen media cero y siguen una función de probabilidad de distribución Gaussiana. Sin embargo, ésta no es una suposición válida, especialmente cuando el ruido es parte de una función no lineal. Por ejemplo, el problema de la estimación de la actitud y rumbo usando EKF consiste en la linealización de ecuaciones dinámicas altamente no lineales. El filtro de Kalman Extendido trabaja de manera eficiente en las zonas en las que se realizó la linealización, fuera de ellas presenta problemas de divergencia. Además no garantiza la convergencia global del sistema.

1.4.3. Estimación de la actitud y rumbo usando observadores no lineales

En los últimos años, importantes esfuerzos en la investigación se han dirigido hacia la síntesis de observadores no lineales de la orientación, con el fin de abordar los problemas de divergencia, la distribución no Gaussiana del ruido y alto costo computacional que presenta el Filtro de Kalman Extendido. El primer trabajo de este tópico fue presentado en [9]. Posteriormente, en [10] y en [11] observadores no lineales basados en cuaterniones y observadores del bias de los girómetros son propuestos en el marco de vehículos de navegación de la marina y en la calibración de sensores de satélites, respectivamente. Además, la formulación de observadores no lineales de la orientación basados en el grupo ortogonal de matrices $SO(3)^2$ se han propuesto recientemente en [12], [13], [14]. Estos observadores no lineales (a excepción de [14]) no toman en cuenta el ruido presente las mediciones de girómetros ni el término del bias variante en el tiempo. Además, la complejidad computacional de estos enfoques es una desventaja, ya que estos algoritmos trabajan con matrices de 3×3 que tienen que conservar su ortogonalidad.

1.5. AHRS Strapdown comerciales

En el mercado existen diversos sistemas que permiten la estimación de la posición y la orientación de un cuerpo en el espacio [15] [16]. Sin embargo, estos sistemas son caros y además el algoritmo de integración de datos diverge cuando experimentan movimientos de alta frecuencia. Debido a estos inconvenientes los AHRS comerciales no han tenido una gran penetración en el mercado. En la tabla se hace la comparación de dos AHRS comerciales: MTi de Xsens y 3DM-GX3-25 de MicroStrain.

Características	MTi (kg)	3DM-GX3-25
Rango de orientación	360°	360°
Rango de vel. angular	$\pm 360^\circ/s$	$\pm 360^\circ/s$
Resolución de la orientación (c. estáticas)	$\pm 0,5^\circ$	$\pm 0,5^\circ$
Resolución de la orientación (c. dinámicas)	$\pm 2,0^\circ$	$\pm 2,0^\circ$
Frecuencia de la salida de datos	512HZ	1000HZ
Voltaje de alimentación	4.4V -6.0V	4.5V -30.0V
Dimensiones	$58 \times 58 \times 22mm$	$44 \times 25 \times 11mm$
Interfaz digital	USB 2.0 y/o RS-232	USB2.0 y/o RS-232
Peso	50g	18g
Precio	2334 USD	2295 USD

Tabla 1.1: Especificaciones técnicas de MTi y 3DM-GX3-25 (ver [17] y [18] para más detalles)

Capítulo 2

Sensores utilizados en el monitoreo ambiental y navegación

En el presente capítulo se exponen los diversos tipos de sensores utilizados para los sistemas de monitoreo ambiental y navegación. Comenzando con una breve descripción de la tecnología utilizada en la realización de los sensores tipo MEMS que son los sensores utilizados para el sistema AHRS, también se describe brevemente el principio de funcionamiento de cada sensor para el sistema de monitoreo así como algunas de sus versiones disponibles en el mercado y su principio de funcionamiento.

2.1. Sensores inerciales MEMS

Los sistemas microelectromecánicos (Microelectromechanical Systems, MEMS) son dispositivos de escala micrométrica que integran elementos mecánicos y electrónicos. El tamaño de estos dispositivos puede variar desde un micrómetro hasta un milímetro. (ver figura 2.1).

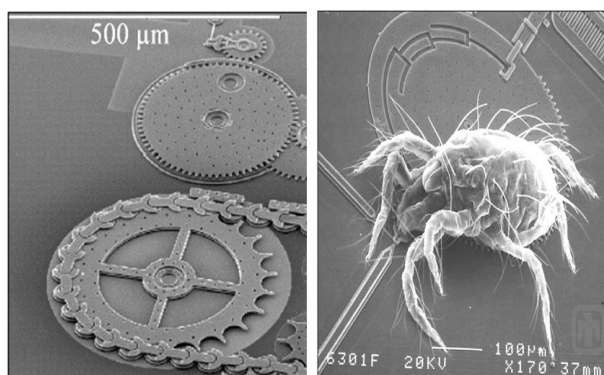


Figura 2.1: Imagen de un dispositivo MEMS a la izquierda y la diferencia de tamaño de un ácaro y el dispositivo a la derecha

Los dispositivos MEMS superan muchas de las características que habían impedido la adopción de sistemas inerciales en muchas aplicaciones potenciales especialmente donde: el costo, el tamaño y el consumo de energía se encuentran limitados. Las fuerzas externas que

actúan sobre un cuerpo causan una aceleración y/ó un cambio de su orientación (posición angular). La rapidez del cambio de la posición angular es la velocidad angular. Un sensor inercial que mide la aceleración lineal de un cuerpo sobre su eje sensible se llama acelerómetro. Del mismo modo, un sensor que mide la velocidad angular de un sistema inercial sobre su eje sensible se llama girómetro. Los sensores inerciales MEMS ofrecen una alta resolución y un bajo consumo de energía, en un tamaño extremadamente pequeño. Debido a esto se han convertido en los sensores más populares para medir movimiento, aceleración, inclinación y vibración.

2.1.1. Acelerómetros MEMS

Un acelerómetro es un sensor que mide aceleraciones o movimientos de traslación. Los acelerómetros MEMS están formados principalmente por tres elementos: una masa de prueba que genera una fuerza inercial debido a una aceleración o desaceleración en el acelerómetro, un resorte que soporta la masa de prueba y la regresa a su posición inicial, una vez que la aceleración termina, un sistema de medición relaciona la señal de salida con la aceleración inducida. Como se muestra en la figura 2.2, la masa de prueba tiende a resistirse al cambio de movimiento debido a su propia inercia. Como resultado, la masa es desplazada en sentido contrario respecto al cuerpo. La extensión neta del resorte provee una medida de la fuerza aplicada, que es proporcional a la aceleración.

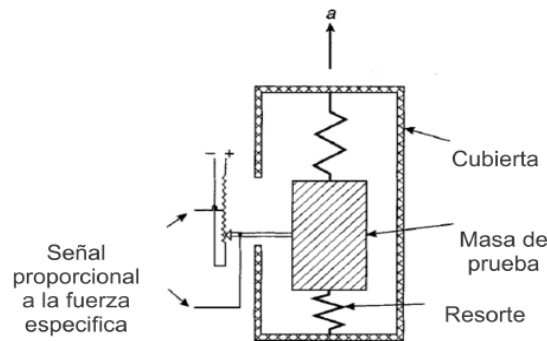


Figura 2.2: Estructura básica de un acelerómetro de tecnología MEMS

Una aceleración en dirección positiva causará que la masa de prueba se mueva hacia abajo con respecto a la posición de equilibrio. Cuando el acelerómetro se encuentra estático y bajo un campo gravitacional. La masa de prueba se desplaza hacia abajo debido a la fuerza gravitacional. Como el acelerómetro se encuentra estático, una aceleración en dirección contraria debe ser ejercida por éste. Por lo tanto, éste ejerce una menor aceleración gravitacional.

Para orientarse con respecto a un sistema de referencia inercial, se requiere conocer las aceleraciones del cuerpo a lo largo de tres ejes. Comúnmente se utilizan tres acelerómetros de un eje montados ortogonalmente para proveer mediciones independientes de una fuerza específica.

2.1.2. Girómetros MEMS

Un girómetro es un sensor de velocidad angular que mide la velocidad angular de rotación. Los girómetros MEMS usan elementos mecánicos vibrantes llamados: masas de prueba para monitorear el movimiento rotacional. Estos sensores no tienen ninguna parte móvil que requiera rodamientos y por lo tanto pueden ser miniaturizados para reducir el costo del sensor. Los girómetros MEMS basan su funcionamiento en la detección de una fuerza. Dicha fuerza actúa sobre la masa de prueba que está sujeta a un movimiento lineal vibratorio dentro de un sistema de referencia inercial; el cual está girando alrededor de un eje perpendicular al eje del movimiento lineal. La fuerza resultante se conoce como fuerza de Coriolis y actúa en una dirección perpendicular al eje de rotación y al eje de vibración. Este fenómeno se muestra en la figura 2.3.

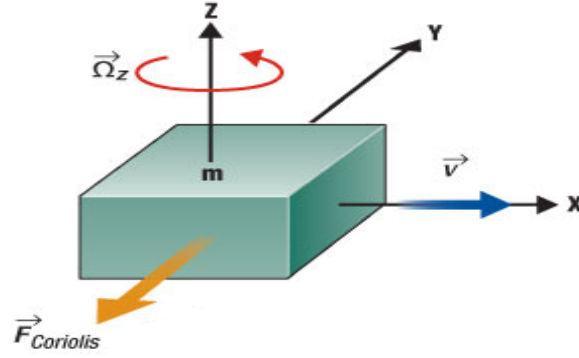


Figura 2.3: Fuerza de Coriolis

La fuerza de Coriolis es llamada muchas veces como una fuerza ficticia por que no aparece cuando el movimiento es expresado en un sistema de referencia inercial [19]. La magnitud de la fuerza de Coriolis está dada por

$$\vec{F}_{Coriolis} = -2m\vec{\omega} \times \vec{v} \quad (2.1)$$

donde $\vec{v}$ y $\vec{\omega}$ son la velocidad lineal y la velocidad angular respectivamente, a las que es sometida la masa de prueba. Todos los girómetros vibratorios hacen uso del efecto de Coriolis para medir la velocidad angular.

Cuando un girómetro es acelerado linealmente, la masa de prueba se mueve en la misma dirección. Por lo tanto, no se genera la fuerza de Coriolis y la salida del girómetro será el valor correspondiente a una velocidad angular cero, esto muestra que los girómetros MEMS no son sensibles a aceleraciones lineales, tales como inclinaciones, golpes o vibraciones.

La velocidad angular ω_G es medida por tres girómetros montados ortogonalmente. La señal de salida del girómetro se ve influenciada por un ruido de baja frecuencia conocido como deriva, esto es

$$\vec{\omega}_G = \vec{\omega} + \vec{v} + \vec{\eta}_G \quad (2.2)$$

$$\dot{\vec{v}} = -T^{-1}\vec{v} + \vec{\eta}_v \quad (2.3)$$

donde $\vec{\eta}_G$ y $\vec{\eta}_v$ se suponen como ruidos blancos y Gaussianos; y $T = \tau I_3$ es una matriz diagonal de constantes de tiempo.

2.2. Sensor Magneto-resistivo

Los magnetómetros son sensores que generalmente miden el vector de campo magnético. La medición puede incluir la magnitud del vector del campo magnético, su dirección o ambos.

Los magnetómetros pueden ser clasificados de acuerdo a su rango de medición del campo magnético[20]. Los magnetómetros que detectan campos magnéticos menores a 1μ Gauss se consideran de bajo campo y aquellos que tienen un rango de medición de 1μ G a 10 Gauss se conocen como sensores de deriva del campo magnético.

A su vez existen distintos tipos de magnetómetros que se diferencian entre si por el efecto que usan para medir el campo magnético, entre ellos podemos encontrar los magnetómetros magneto-resistivos que se basan en el cambio de la resistencia que experimenta un material cuando es expuesto a un campo magnético, si bien este efecto esta presente en toda la materia se puede distinguir mayormente en materiales ferromagnéticos [21]. También existen los magnetómetros magneto-inductivos que consiste en una bobina con un núcleo ferromagnético el cual se usa para medir el campo magnético.

El sensor magneto-resistivo está hecho de una delgada película de una aleación de níquel e hierro (Permalloy). En presencia de un campo magnético el permalloy cambia su resistencia en un 2-3 %. Generalmente, estas resistencias se usan en un arreglo de puente de Wheatstone como se muestra en la figura 2.4; para formar un sensor magneto-resistivo de un eje de medición.

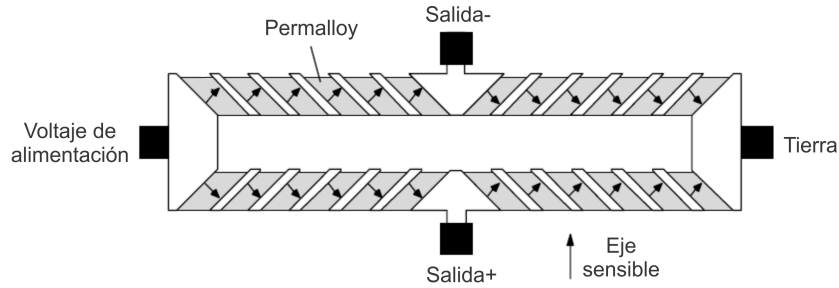


Figura 2.4: Magnetómetro magneto-resistivo.

Los sensores magneto-resistivos tienen un gran sensibilidad y un amplio rango de temperatura de operación. Su alta sensibilidad se presta para aplicaciones de medición del campo magnético terrestre, brújulas electrónicas y sistemas de navegación.

2.3. Sensores utilizados en el monitoreo ambiental

Existen una gran variedad de sensores utilizados en el monitoreo ambiental, pero para uso específico de ésta tesis solo se utilizan cuatro tipo de sensores: los cuales son humedad,

temperatura, calidad del aire y presión atmosférica. Por lo tanto, estos son los sensores de los cuales se hablarán en la siguiente sección.

2.3.1. Sensor de humedad

La humedad es un fenómeno natural que se presenta a nivel molecular y se encuentra básicamente relacionada con la cantidad de moléculas de agua presentes en una determinada sustancia, la cual puede estar en estado sólido o gaseoso. Típicamente los sensores de humedad miden el porcentaje de humedad relativa, que es la relación entre la presión parcial de vapor actual y la presión de vapor de saturación expresada en porcentaje o de otra manera, es la cantidad de agua que contiene un gas expresada en porcentaje a la misma temperatura y presión absoluta.

Los sensores de humedad se pueden clasificar por el principio que utilizan para medir el porcentaje de humedad, aunque el principio más común entre estos sensores es el sensor por deformación. El principio de funcionamiento de este sensor es aprovechar los cambios en las dimensiones que sufren ciertos tipos de materiales en presencia de humedad, al aumentar la humedad algunas fibras orgánicas o sintéticas aumentan de tamaño, esta deformación es proporcional a la humedad percibida. Así que para medir esta deformación se utiliza una banda extensiométrica que al cambiar su forma también cambia su resistencia la cual puede ser mediada aplicando un voltaje conocido y medido por un controlador externo que traduce esta variación de voltaje en porcentaje de humedad relativa. En la figura 2.5 se ilustra lo anterior mencionado.

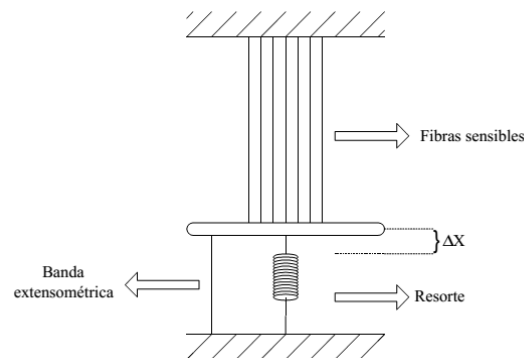


Figura 2.5: Sensor de humedad por deformación

2.3.2. Sensor de calidad del aire

Este sensor mide la cantidad de concentración de gases contaminantes en el aire, es decir, en realidad es un detector de gases, que se basan en general en propiedades características de los gases, tales como la conductividad térmica y el paramagnetismo del oxígeno.

La conductividad térmica se usa principalmente para la detección de gases con una conductividad térmica mayor a la del aire, para percibir esta diferencia se colocan dos elementos

metálicos en el sensor, uno de referencia en un espacio cerrado herméticamente y otro para expuesto al medio ambiente (ver figura 2.6). Si la conductividad térmica del gas es menor a la del elemento de referencia la temperatura del elemento expuesto incrementa e inversamente. Estos cambios de temperatura son proporcionales a la concentración de gas presente en el elemento expuesto.

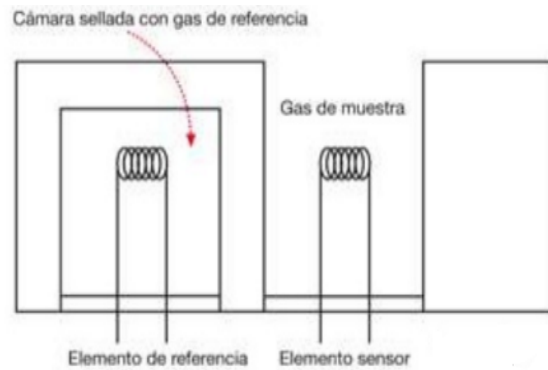


Figura 2.6: Funcionamiento de sensor de gas por conductividad térmica

La mayoría de los gases son diamagnéticos y son repelidos del campo magnético. El oxígeno es diferente, es un gas paramagnético, lo que significa que es atraído por un campo magnético. El paramagnetismo depende mucho de la temperatura. Las moléculas de oxígeno frío son atraídas por un campo magnético fuerte y cuando se calientan dejan el campo magnético. Esto da lugar a una corriente, la cual genera mediciones de oxígeno.

Existe una diversidad de gases que se pueden medir en el medio ambiente y normalmente se miden en partes por millón, entre los gases que se pueden medir están: monóxido de carbono (CO), Dióxido de carbono (CO₂), Óxidos de nitrógeno (NO_x) y Ozono (O₃).

2.3.3. Sensor de temperatura

Los sensores de temperatura son dispositivos que transforman los cambios de temperatura en cambios en señales eléctricas que son traducidos por un procesador. Existen distintos tipos de estos sensores pero comúnmente se utilizan los siguientes con algunas variantes dependiendo del fabricante.

- **Termistor:** está basado en el comportamiento de los semiconductores que varían su resistencia eléctrica en función de la temperatura. Existen termistores de tipo NTC que al aumentar la temperatura disminuye la resistencia y de tipo TPC que al incrementar la temperatura aumenta la resistencia.
- **RTD *Resistance Temperature Detector*:** Un RTD es un sensor de temperatura basado en la variación de la resistencia de un conductor con la temperatura. Los metales empleados normalmente son platino, cobre, níquel y molibdeno.
- **Termopar:** está formado por dos metales distintos que generan una diferencia de potencial en función de la temperatura percibida, al medir esta diferencia de potencial se puede conocer la temperatura.

2.3.4. Sensor de rayos Ultravioleta

Los sensores de luz ultravioleta se basan en la luminiscencia de los materiales, la luz ultravioleta cubre un espectro de los 400 nm a los 15 nm.

Los sensores Uv detectan la fluorescencia de un objeto sometido a la luz ultravioleta. Por lo tanto, una vez que se emite una luz ultravioleta de cualquier fuente como el sol este rayo luminoso pasa a través de una lente que lo focaliza hacia un objeto y este a su vez refleja esta luz hacia un fotodiodo que detecta esta luz y envía una señal proporcional a la cantidad de rayos Uv detectada (ver figura 2.7).

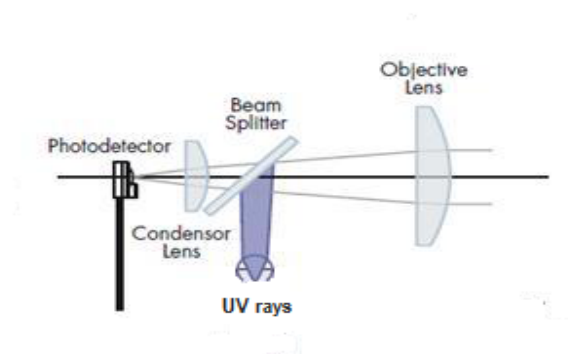


Figura 2.7: Funcionamiento de sensor de rayos UV

2.3.5. Sensor de presión atmosférica

En general estos sensores convierten la presión medida en una señal eléctrica, para realizar esta función se emplean varios materiales en el sensor como silicio, cerámica o metal, los tres principios más comunes desarrollados en laboratorios para estos sensores son los siguientes.

- **Sensores de película delgada:** están basados en mismo principio que las galas extensiométricas, que son estructuras de rejillas conductoras que cambian su resistencia en función de las variaciones de expansión o grosor inducidos por el cambio de presión. Estas galgas extensiométricas están colocadas y estructuradas sobre un elemento metálico.
- **Sensores de película gruesa:** funciona con el mismo principio que el anterior solo que las galgas son más gruesas y están estructuradas sobre una base cerámica.
- **Sensores piezoresistivos:** en este caso se aplica una membrana de un material semiconductor (silicio) con estructuras selectivamente distribuidas. El funcionamiento de los sensores está basado en el efecto piezoresistivo que consiste en una variación de la resistencia en el semiconductor causado por su expansión y compresión que influye en la movilidad de los electrodos bajo carga mecánica.

2.4. GPS

2.4.1. Antecedentes

En el año de 1959 después del lanzamiento del satélite espacial estadounidense *Vanguard* se puso en manifiesto que la transmisión de señales de radio desde el espacio podría servir para orientarnos en la superficie terrestre.

Anteriormente, los sistemas de posicionamiento empleaban estaciones terrestres de amplitud modulada cubrían un área mayor que los de UHF(frecuencias ultracortas), pero ninguno de estos sistemas podían determinar con exactitud la posición, esto debido a las interferencias atmosféricas que afectan las señales de radio de amplitud modulada y a la propia curvatura de la tierra que desvía las ondas.

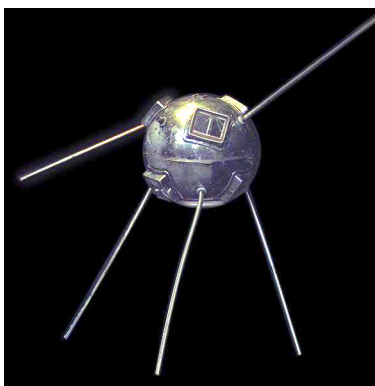


Figura 2.8: Satélite Vanguard puesto en órbita en marzo de 1958

Por tanto, la única forma de solucionar este problema era colocando transmisores de radio en el espacio, los cuales emitieran constantemente señales codificadas en dirección a la tierra. De hecho, esas señales cubrirían un área mayor que las de amplitud modulada.

Sin embargo, no fue hasta 1993 que el departamento de Defensa de los Estados Unidos de América, basado en la información recolectada del satélite vanguardia, puso en funcionamiento un sistema de localización por satélite conocido por GPS.

La primera prueba exitosa del sistema GPS como instrumento de navegación, se realizó a bordo del transbordador espacial *Discovery* en 1993. Actualmente los satelites GPS pertenecen a una segunda generación denominada *Block II*.

2.4.2. Características

Los receptores GPS están preparados para determinar con un margen mínimo de error la latitud, longitud, y altura desde cualquier punto en la tierra.

El funcionamiento del GPS se basa al igual que los sistemas de navegación antiguos en el principio de triangulación. Esto significa que el receptor de GPS determine con exactitud la distancia que lo separa de los satélites.

Cada satélite GPS emite continuamente una señal con un mensaje de navegación de 50 bits por segundo en dos frecuencias transportadoras a 1575,42 MHz y 1227,6 MHz.

La señal proveniente del GPS proporciona la hora precisa de acuerdo con el reloj atómico a bordo del satélite, el número de semana GPS y un informe de estado para el satélite. Cada transmisión dura 30 segundos y lleva 1500 bits de datos codificados y que es diferente para cada satélite. Los receptores GPS pueden decodificar la señal y distinguir entre diferentes satélites.

La primera parte de la señal GPS indica al receptor la relación entre el reloj del satélite y la hora GPS. La siguiente serie de datos proporciona al receptor información de órbita precisa del satélite que sirve para triangular la posición del receptor.

Para ubicar la posición exacta donde nos encontramos situados, el receptor GPS tiene que localizar por lo menos 3 satélites que le sirvan de puntos de referencia y uno más para correcciones tal como se muestra en la figura 2.9. Para determinar el lugar exacto de la órbita donde se encuentran los satélites el receptor tiene una memoria que contiene datos sobre cada satélite en órbita.



Figura 2.9: Funcionamiento de GPS

2.4.3. Aplicaciones del GPS

Debido a que es un sistema gratuito, ininterrumpido y fiable, el Sistema de Posicionamiento Global (GPS) ha permitido desarrollar cientos de aplicaciones que afectan casi todas las facetas de la vida moderna.

Las aplicaciones son muy variadas ya que cada día se le dan nuevos usos al GPS, la única limitante es la creatividad humana. Entre las aplicaciones más comunes están: agricultura, aviación, carreteras y autopistas, cronometría, medio ambiente, navegación marítima, recreación, seguridad pública, topografía, vías férreas, etc.



Figura 2.10: Aplicaciones del GPS.

Capítulo 3

Diseño e implementación del sistema de navegación

En este capítulo se describe la integración de la programación del FPGA y los sensores que conforman al sistema AHRS. En la figura 3.1 se muestra el diagrama de esta integración. Primero, se describe el observador no lineal que es el algoritmo de estimación de la orientación. Después, se describen la integración de cada sensor y su configuración, también se describe la integración del modulo XBee, del GPS y la interfaz gráfica. Por último, se describe la integración de todos los elementos del hardware y software.

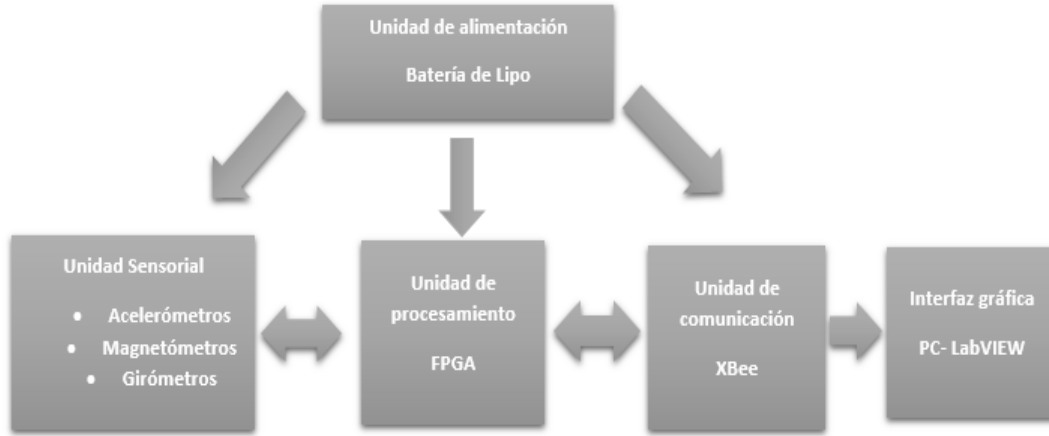


Figura 3.1: Diagrama de integración de AHRS.

3.1. Cuaterniones unitarios y cinemática de actitud

Considera dos sistemas ortogonales, de marco de coordenadas de la mano derecha: el cuerpo del marco de coordenadas, $E^b = [\vec{e}_1^b, \vec{e}_2^b, \vec{e}_3^b]$, localizado en el centro de masa del cuerpo rígido, y el marco de coordenadas inercial $E^f = [\vec{e}_1^f, \vec{e}_2^f, \vec{e}_3^f]$, localizado en algún punto fijo en el espacio. La rotación del marco del cuerpo, E^b , con respecto al cuadro fijo, E^f , está representado por la matriz de actitud $R \in SO(3) = R \in \mathbb{R}^{3 \times 3} : R^T R = I, \det R = 1$.

El producto cruz entre dos vectores, $\vec{\chi}, \vec{\xi} \in \mathbb{R}^3$, está representado por una multiplicación de

matriz $[\vec{\xi}^\times]\chi = \vec{\xi} \times \chi$, donde:

$$[\vec{\xi}^\times] = \begin{pmatrix} 0 & -\xi_3 & \xi_2 \\ \xi_3 & 0 & -\xi_1 \\ -\xi_2 & \xi_1 & 0 \end{pmatrix} \quad (3.1)$$

La unidad esférica n-dimensional embebida en $\mathbb{R}^{n+1}$ esta denotada como $\mathbb{S}^n = \{x \in \mathbb{R}^{n+1} : x^T x = 1\}$. Miembros de $SO(3)$ son frecuentemente parametrizados en términos de la rotación, $\beta \in \mathbb{R}$, acerca de un eje fijo, $\vec{e} \in \mathbb{S}^2$, por el mapa, $\mathcal{U} : \mathbb{R} \times \mathbb{S}^2 \rightarrow SO(3)$, definido como:

$$\mathcal{U}(\beta, \vec{e}) := \cos\beta I - \sin\beta[\vec{e}^\times] + (1 - \cos(\beta))[\vec{e}^\times] \quad (3.2)$$

Por lo tanto, el cuaternión unitario, $q \in [\mathbb{S}]^3$, está definido como:

$$q := \begin{pmatrix} \cos\frac{\beta}{2} \\ \vec{e}\sin\frac{\beta}{2} \end{pmatrix} = \begin{pmatrix} q_0 \\ \vec{q} \end{pmatrix} \in \mathbb{S}^3 \quad (3.3)$$

$\vec{q} = [q_1 \ q_2 \ q_3]^T \in \mathbb{R}$ y $q_0 \in \mathbb{R}$ son conocidos como partes escalares del cuaternión, respectivamente. q representa un elemento de $SO(3)$ a través del mapa, $\mathcal{R} : \mathbb{S}^3 \rightarrow SO(3)$, definido como:

$$\mathcal{R} := (q_0^2 - \vec{q}^T \vec{q})I - 2q_0[\vec{q}^\times] + 2\vec{q}\vec{q}^T \quad (3.4)$$

Note que $R = \mathcal{R}(q) = \mathcal{R}(-q)$ por cada $q \in \mathcal{S}^3$, i.e., quaterniones q y $-q$ representan la misma actitud física. La unidad inversa del cuaternión esta dada por $q^{-1} := [q_0 - \vec{q}^T]^T$ y el producto del cuaternión esta definido por:

$$q_1 \otimes q_2 := \begin{pmatrix} q_{1_0} & -\vec{q}_1^T \\ \vec{q}_1 & I_3 q_{1_0} + [\vec{q}_1^\times] \end{pmatrix} \begin{pmatrix} q_{2_0} \\ \vec{q}_2 \end{pmatrix} \quad (3.5)$$

Sea $\vec{\omega} = [\omega_1 \ \omega_2 \ \omega_3]^T$ que denota el vector de velocidad angular del cuerpo de marco de coordenadas, $\mathbf{E}^b$, con respecto al marco de referencia inercial, $\mathbf{E}^f$, expresado en $\mathbf{E}^b$, la ecuación cinemática está dada por:

$$\begin{pmatrix} \dot{q}_0 \\ \dot{\vec{q}} \end{pmatrix} = \frac{1}{2} \begin{pmatrix} -\vec{q}^T \\ I_3 q_0 + [\vec{q}^\times] \end{pmatrix} \vec{\omega} = \frac{1}{2} \Xi(q) \vec{\omega} \quad (3.6)$$

El error de actitud es usado para cuantificar el desfase entre las dos actitudes.

Si q define el cuaternión real de la actitud y $\hat{q}$ es el cuaternión estimado, entonces el cuaternión de error que representa el error de actitud entre la orientación real y la estimada viene dado por:

$$q_e := \hat{q}^{-1} \otimes q = (q_{e0} \vec{q}_e^T)^T \quad (3.7)$$

En el caso que el cuaternión real y el estimado coinciden, el cuaternión de error se convierte:

$$q_e = \pm(1, 0)^T \quad (3.8)$$

3.2. Sensores de velocidad angular

La velocidad angular $\vec{\omega} = (\omega_1 \ \omega_2 \ \omega_3)^T$ se mide por tres girómetros montados ortogonalmente. La señal de salida del girómetro esta contaminada por el ruido y por una función de variación lenta llamada bias, esto es:

$$\vec{\omega}_G = \vec{\omega} + \vec{\nu} + \vec{\eta}_G \quad (3.9)$$

$$\dot{\vec{\nu}} = -T^{-1}\vec{\nu} + \vec{\eta}_\nu \quad (3.10)$$

Donde η_G y $\eta_\nu \in \mathbb{R}^3$ están delimitadas por el ruido y $T = \tau I_3$ es una matriz diagonal con las constantes de tiempo.

3.3. Sensores de vectores de referencia

Consideremos la representación de un vector, $\vec{x}_i$, con respecto a $\mathbf{E}^f$ y $\mathbf{E}^b$ y denotado por $\vec{r}_i$ y $\vec{b}_i$, respectivamente. Los vectores $\vec{r}_i$, son también llamados vectores de referencia, y en general se conocen por su precisión. Los vectores, $\vec{b}_i$ son conocidos como vectores de observación y son obtenidos de la la tarjeta de sensores (por ejemplo magnetómetros y acelerómetros). Estas dos representaciones están vinculadas a través de la matriz de rotación $R \in SO(3)$

$$\vec{b}_i = R\vec{r}_i \quad (3.11)$$

Por otro lado, se deja a b_{i_q} y r_{i_q} como el cuaternión asociado a los vectores, $\vec{b}_i$ y $\vec{r}_i$:

$$b_{i_q} = (0 \ \vec{b}_i^T)^T \quad r_{i_q} = (0 \ \vec{r}_i^T)^T \quad (3.12)$$

Estos cuaterniones están relacionados por la rotación del cuaternión, q , de modo que:

$$b_{i_q} = q^{-1} \otimes r_{i_q} \otimes q \quad (3.13)$$

De la ecuación 3.5 y usando el álgebra de cuaternión, se puede obtener el siguiente modelo de medición.

$$\begin{pmatrix} 0 & -(\vec{b}_i - \vec{r}_i)^T \\ \vec{b}_i - \vec{r}_i & -[(\vec{b}_i + \vec{r}_i)^T] \end{pmatrix} q = Hq = 0 \quad (3.14)$$

con $H \in \mathbb{R}^{4 \times 4}$. Note que 3.14 es un sistema lineal de ecuaciones bien estructurado. En la aplicación de los sensores inerciales y magnéticos, el sistema de coordenadas inercial, $\mathbf{E}^f$ es el escogido para ser el NED (North, East, Down). En este trabajo, el origen de $\mathbf{E}^f$ está localizado en la ciudad de Puebla, México (GPS:19° 01' 41.78" N 98° 11' 16.81" O). Los vectores de referencia son los vectores gravitacionales y magnéticos que son bien conocidos. Los vectores de observación, es decir, el gravitacional y el magnético expresado en cuerpo de referencia $\mathbf{E}^b$, son obtenidos de los tres ejes del acelerómetro y los tres ejes del magnetómetro. Las mediciones del acelerómetro expresadas en el cuerpo de de referencia B se pueden escribir como:

$$\vec{b}_A = R(\vec{a} - g\vec{e}_3^f) + \eta_s \quad (3.15)$$

donde $g\vec{e}_3^f = [0 \ 0 \ \bar{g}]^T$ y $\vec{a}$ son el vector de gravedad y aceleración inercial del cuerpo respectivamente expresado en el marco de referencia inercial $\mathbf{E}^f$. $\bar{g} = 9,81m/sec^2$ denota la constante de gravedad y η_s se asume como el ruido acotado. Si la aceleración absoluta del

cuerpo rígido en el sistema de coordenadas inercial es tenue ($\|\vec{a}\| \ll \|g\vec{e}_3^f\|$), el vector de observación debido al acelerómetro es:

$$\vec{b}_A = R\vec{r}_A + \eta_A \quad (3.16)$$

donde $\eta_M \in \mathbb{R}^3$ denota la perturbación magnética acotada por el ruido. De acuerdo con 3.11 y 3.13 la ecuación de medida de los acelerómetros y magnetómetros se convierte en:

$$\begin{pmatrix} 0 & -(\vec{b}_A - \vec{r}_A)^T \\ \vec{b}_A - \vec{r}_A & -[(\vec{b}_A + \vec{r}_A)^\times] \\ 0 & -(\vec{b}_M - \vec{r}_M)^T \\ \vec{b}_M - \vec{r}_M & -[(\vec{b}_M + \vec{r}_M)^\times] \end{pmatrix} q = \bar{H}q = 0 \quad (3.17)$$

3.4. Planteamiento del problema

El objetivo es diseñar un observador que estime la actitud de un cuerpo rígido haciendo una compensación entre una buena aproximación a corto plazo dado por la integración del giroscopio y una estimación fiable a largo plazo dado por las observaciones del vector.

Dado que las observaciones obtenidas de los tres vectores del acelerómetro y del magnetómetro son no colineales, entonces el cuaternión se puede calcular con estas mediciones simultáneas de vectores. Para este propósito la ecuación 3.18 de medición se define como

$$\bar{H}(t)q = \begin{pmatrix} \varphi_1^T(t) \\ \vdots \\ \varphi_8^T(t) \end{pmatrix} q = 0 \quad (3.18)$$

El cuaternión desconocido q , al cual llamaremos *QPS*, se puede estimar al minimizar la siguiente función de costo:

$$J(q) = \frac{1}{2} \sum_{i=1}^8 (\varphi_i^T(t)q)^T (\varphi_i^T(t)q) = \frac{1}{2} \|\bar{H}(t)q\|^2 \quad (3.19)$$

Por otro lado, si el valor inicial, $q(t_0)$, es conocido y las medidas de la velocidad angular son perfectas, es decir, $\vec{\omega} = \vec{\omega}_G$, es posible obtener $q(t)$ mediante la integración de la ecuación cinemática 3.6. Sin embargo, $q(t_0)$ es en general desconocido. Por otro lado, las medidas de las velocidades angulares están contaminadas por el bias y el ruido, lo que resulta en una divergencia lenta de actitud respecto al tiempo. Por lo tanto, la idea de la observación se pueda utilizar para corregir la divergencia en la integración, $\hat{q}(t)$ de algunos $\hat{q}(t_0)$ erróneos y errores de actitud causados por el bias y el ruido presentes en las medidas del girómetro. Esta corrección puede ser llevada a cabo como una función obtenida del error $q_e = \hat{q}^{-1} \otimes q_{ps}$, donde q_{ps} es considerado la medida de la actitud obtenida de los vectores de observación.

El problema de la observación puede ser formulado como sigue: Dada la ecuación cinemática de un cuerpo rígido y el modelo variante en el tiempo de la evolución del bias, se encuentra el $\hat{q}(t)$ estimado y $\hat{\nu}(t)$ del conocido $\omega_G(t)$ y $q_{ps}(t)$ para todo $t \geq t_0 \in \mathbb{R} \geq 0$

3.5. Estimación de la actitud usando el vector de observación.

El método de Kaczmarz [31] es una de las soluciones más populares de sistemas lineales sobredeterminados y tiene numerosas aplicaciones en la tomografía computarizada para el control de imágenes y control adaptativo. Es un método iterativo y por lo tanto es práctico en el ámbito de grandes sistemas de ecuaciones. El algoritmo consiste en una serie de proyecciones alternas, y esto a menudo es considerado como un tipo de método de proyección en conjuntos convexos. En este trabajo, el método de Kaczmarz es usado para estimar la actitud del cuaternión usando el vector de observación. Para describir este algoritmo, se debe considerar la ecuación dada por 3.18 y la primera línea de $\tilde{H}(t)$

$$0 = \varphi_1(t)^T q \quad (3.20)$$

Supongamos que un q_{ps}^0 es conocido y la nueva medida es obtenida como (3.20). Una propuesta para actualizar la estimación es:

$$q_{ps}^1 = q_{ps}^0 + \alpha \varphi_1(t) \quad (3.21)$$

donde el parámetro α es elegido de la siguiente forma

$$0 = \varphi_1^T(t) q_{ps}^1 = \varphi_1^T(t) q_{ps}^0 + \alpha \varphi_1^T(t) \varphi_1(t) \quad (3.22)$$

De esto se obtiene:

$$\alpha = -\frac{\varphi_1^T(t)}{\varphi_1^T(t) \varphi_1(t)} q_{ps}^0 \quad (3.23)$$

La formula actualizada se convierte en

$$q_{ps}^1 = q_{ps}^0 - \frac{\varphi_1(t) \varphi_1^T(t)}{\varphi_1^T(t) \varphi_1(t)} q_{ps}^0 = \left(I_4 - \frac{\varphi_1(t) \varphi_1^T(t)}{\varphi_1^T(t) \varphi_1(t)} \right) q_{ps}^0 \quad (3.24)$$

que es el método de Kaczmarz. Si este proceso se repite en las ocho líneas de 3.18, entonces los QPS estimados del cuaternión obtenido utilizando las mediciones del magnetómetro y del acelerómetro en el instante t se convierte en:

$$q_{ps}^i = \left[\prod_{j=1}^8 \left(I_4 - \frac{\varphi_j(t) \varphi_j^T(t)}{\varphi_j^T(t) \varphi_j(t)} \right) \right] q_{ps}^{(i-1)} \quad (3.25)$$

Con el fin de evitar un problema potencial que pueda surgir cuando $\varphi(t) = 0$ con $i \in 1, \dots, 8$ por ejemplo un defecto en los sensores, el algoritmo en la practica se modifica como sigue:

$$q_{ps}^i = \left[\prod_{j=1}^8 \left(I_4 - \frac{\gamma \varphi_j(t) \varphi_j^T(t)}{\alpha + \varphi_j^T(t) \varphi_j(t)} \right) \right] q_{ps}^{(i-1)} \quad (3.26)$$

donde $\alpha \geq 0$ y $0 < \gamma < 2$ [31]

3.5.1. Observador no lineal de la orientación

El observador no lineal produce una estimación del bias que varía en el tiempo y una estimación del cuaternión de orientación. El observador no Lineal propuesto para la estimación de la orientación esta dado por:

$$O_A = \begin{cases} \dot{\hat{q}} = \frac{1}{2} \Xi(\hat{q}) [\vec{\omega}_G - \hat{\vec{v}} + \text{sign}(q_{e0}) K_1 \vec{q}_e(t)] \\ \hat{\vec{v}} = -T^{-1} \hat{\vec{v}} - \text{sign}(q_{e0}) K_2 \vec{q}_e(t) \end{cases} \quad (3.27)$$

donde $T \in \mathbb{R}^{3 \times 3}$ fue definido en la ecuación (2.3) y K_1 y K_2 son constantes positivas. $\hat{q}$ es la predicción de la orientación en el tiempo t . Este se obtiene a través de la integración de la ecuación 3.27, usando la velocidad angular medida ω_G , el bias estimado $\vec{v}$ y $\vec{q}_e$ el cual es el vector del cuaternión de error q_e que mide la diferencia entre $\hat{q}$ y la orientación medida q_{ps} obtenido por medio de la matriz de medición $\bar{H}$, esto es

$$q_e = \hat{q}^{-1} \otimes q_{ps} = [q_{e_0} \ \vec{q}_e]^T \quad (3.28)$$

Para una mayor claridad, el término t es omitido en el siguiente análisis. En realidad, la ecuación del observador 3.27 permite integrar los datos surgidos de los vectores de observación y las mediciones del girómetro. La estructura del observador se muestra en la figura 3.2. Note que el cuaternión, q_{ps} , puede ser expresado por:

$$q_{ps} = q \otimes \Delta q \quad (3.29)$$

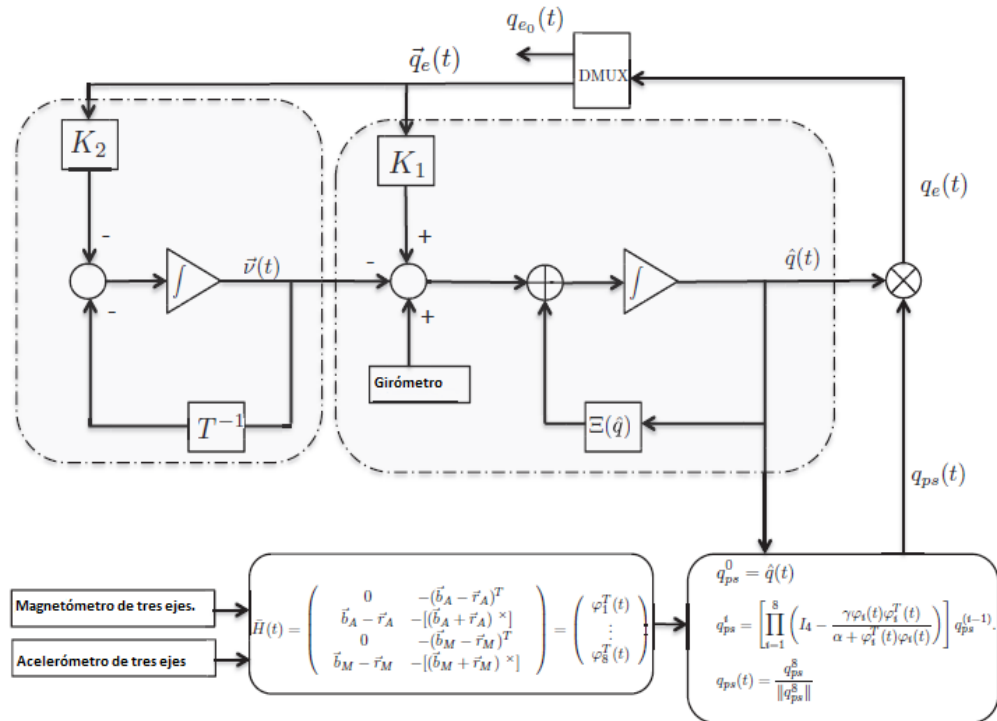


Figura 3.2: Diagrama de integración de AHRS.

donde q denota la actitud real y Δq representa el cuaternión de perturbación, que es causada por el ruido de la medida del sensor en la practica.

3.6. Integración numérica usando Runge Kutta de orden 4

En esta sección se desarrolla la solución numérica para el sistema no lineal variable en el tiempo $\vec{q}_e^T$. La integración numérica constituye un gran número de métodos matemáticos para

la resolución de integrales definidas o ecuaciones diferenciales entre ellos están los métodos de Runge Kutta que son métodos iterativos, en especial se elige el Runge Kutta de orden 4 también conocido como RK4 el cual se describe a continuación.[30]

$$k_1 = hf(x(k), u(k), t_k) \quad (3.30)$$

$$k_2 = hf(x(k) + k_1/2, u(k), t_k + h/2) \quad (3.31)$$

$$k_3 = hf(x(k) + k_2/2, u(k), t_k + h/2) \quad (3.32)$$

$$k_4 = hf(x(k) + k_3/2, u(k), t_k + h) \quad (3.33)$$

$$x(k+1) = x(k) + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) \quad (3.34)$$

El error global de truncamiento para el método de RK4 es de orden $O(h^4)$ y la región de estabilidad viene dada por:

$$|r_s| = \left| 1 + \bar{h} + \frac{1}{2}\bar{h}^2 + \frac{1}{6}\bar{h}^3 + \frac{1}{24}\bar{h}^4 \right| < 1 \quad (3.35)$$

3.6.1. Integración por RK4 de la ecuación cinemática del cuaternión

De la ecuación 3.27 el vector $\dot{\hat{q}}$ se puede representar de la siguiente forma:

$$\begin{pmatrix} \dot{q}_0 \\ \dot{\vec{q}} \end{pmatrix} = \frac{1}{2} \begin{pmatrix} 0 & -\omega^T \\ \omega & -[\omega^x] \end{pmatrix} \begin{pmatrix} q_0 \\ \vec{q} \end{pmatrix} \quad (3.36)$$

Donde,

$$[\omega^x] = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix} \quad (3.37)$$

Sustituyendo 3.37 en la ecuación 3.36 tenemos la matriz M_s

$$\begin{pmatrix} -[\omega^x] & \omega \\ \omega^T & 0 \end{pmatrix} = \begin{pmatrix} 0 & -\omega_1 & -\omega_2 & -\omega_3 \\ \omega_1 & 0 & \omega_3 & -\omega_2 \\ \omega_2 & -\omega_3 & 0 & \omega_1 \\ \omega_3 & \omega_2 & -\omega_1 & 0 \end{pmatrix} \quad (3.38)$$

$$\Rightarrow \dot{q} = \frac{1}{2} M_s q \quad (3.39)$$

Una vez definido el Tiempo de muestreo aplicamos el método de RK4 para integra a $\dot{q}$ quedando las siguientes ecuaciones:

$$k_1 = \frac{1}{2} M_s \cdot q_k \cdot \Delta T \quad (3.40)$$

$$k_2 = \frac{1}{2} M_s (q_k + \frac{1}{2} k_1) \cdot \Delta T \quad (3.41)$$

$$k_3 = \frac{1}{2} M_s (q_k + \frac{1}{2} k_2) \cdot \Delta T \quad (3.42)$$

$$k_4 = \frac{1}{2} M_s (q_k + \frac{1}{2} k_3) \cdot \Delta T \quad (3.43)$$

$$q_{k+1} = \frac{1}{6} (k_1 + 2k_2 + 2k_3 + k_4) + q_k \quad (3.44)$$

3.6.2. Implementación del Runge Kutta en FPGA

Una vez obtenidas las ecuaciones del RK4 se trasladan al FPGA en lenguaje VHDL, para esto se hace una máquina de estados la cual se puede resumir en los siguientes pasos.

1. Convertir ω_1 , ω_2 y ω_3 provenientes de los sensores inerciales a variables de tipo flotante, esto para que tenga una buena precisión y que las variables mantengan el mismo tamaño.
2. Se utiliza un módulo creado en *System Generator* de MATLAB el cual se puede ver en la figura 3.3 y equivale a la siguiente ecuación:

$$k_x = M_s(q_k + \frac{1}{2}k_{x-1}) \quad (3.45)$$

En este módulo se introducen las variables de ω_1 , ω_2 y ω_3 y la variable q_k correspondiente

3. Una vez que el módulo hace las operaciones se toman las salidas de éste y se multiplican por $\frac{\Delta T}{2}$ y la salida es el primer k_x el cual se guarda en una variable específica.
4. Se identifica en que K_x se está calculando para regresar si es necesario al paso 2 o si se continua para hacer la suma de los k_x
5. Una vez calculados todos los k_x se suman como en la ecuación (3.44)

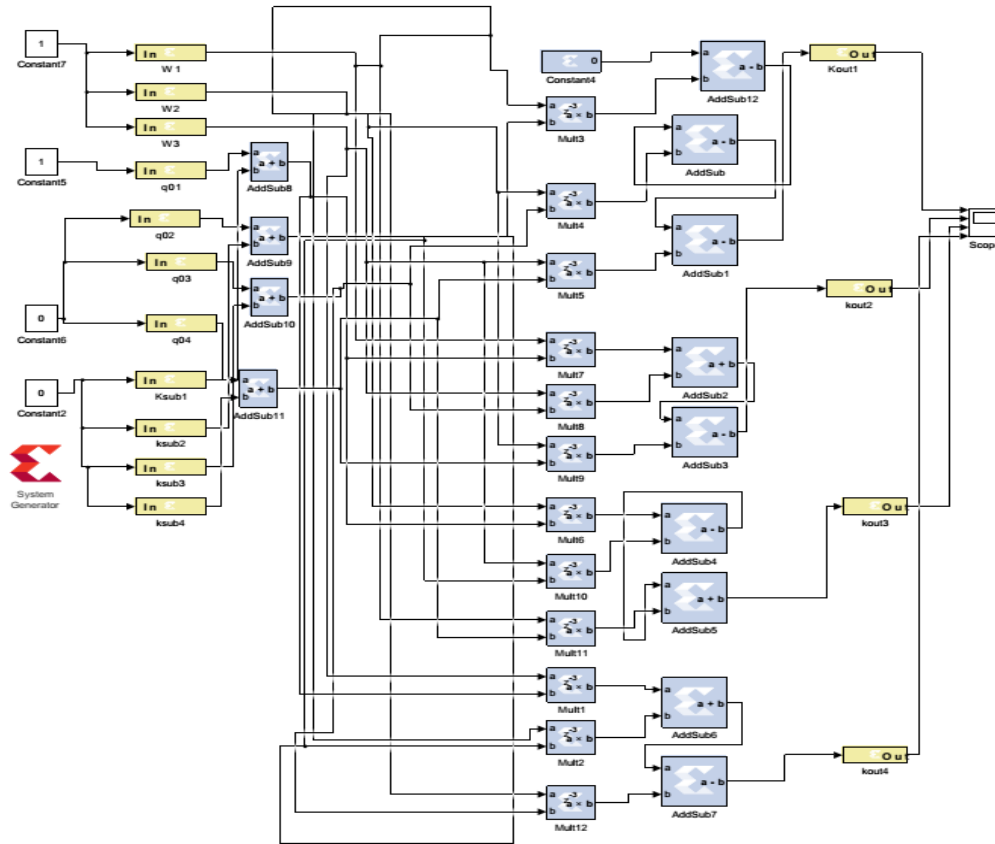


Figura 3.3: Módulo para calcular Runge-Kutta

3.7. Unidad de Mediciones Inerciales (IMU)

Para realizar las mediciones de la aceleración lineal, la velocidad angular y el campo magnetico terrestre en el sistema AHRS se utiliza una IMU. La unidad de mediciones inerciales que se emplea para este propósito es la IMU AltIMU-10 mostrada en la figura 3.4, la cual hace uso de un girómetro *L3GD20*, acelerómetro *LSM303DLHC*, magnetómetro *LSM303DLHC* cada uno con tres ejes de medición; las principales características de estos sensores se muestran en la tabla 3.1, tambien incorpora un barometro *LPS331AP* pero no se utiliza en este proyecto ya que no se contempla en el observador propuesto. Se utiliza esta IMU en el presente trabajo de tesis debido a que los sensores que incorpora son de tecnología MEMS y están colocados en un solo PCB, formando así un sistema de diez grados de libertad aunque solo ocupemos nueve. Además los sensores inerciales presentan un bajo consumo de energía [17] y todos los sensores ocupan un mismo tipo de comunicación que es I^2C teniendo un bus de datos en común y reduciendo el número de conexiones físicas.

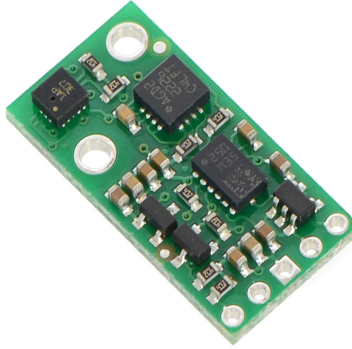


Figura 3.4: IMU AltIMU-10.

Características	Gyro (L3GD20)	Accelerometer (LSM303DLHC)	Magnetometer (LSM303DLHC)
Salida	16 bits con signo	12 bits justificado a la izquierda	12 bits justificados a la derecha
Ejes sensibles	roll,pitch y yaw	roll,pitch y yaw	roll,pitch y yaw
Rango de medición	± 250 , ± 500 o $\pm 2000^\circ/s$	± 2 , ± 4 , ± 8 o ± 16 g	± 1.3 , ± 1.9 , ± 2.5 , ± 4.0 , ± 4.7 , ± 5.6 , o ± 8.1 gauss
Consumo de energía	6.1 mA	110 μ A	110 μ A
Voltaje de alimentación	2.4v - 3.6v	2.16V - 3.6v	2.16v - 3.6v

Tabla 3.1: Especificaciones técnicas de la IMU AltIMU-10

3.8. Procesador digital de señales



Figura 3.5: Xula2 con Spartan 6

Para hacer el procesamiento de señales y el cálculo del sistema AHRS se utiliza un FPGA el cual se detallará en esta sección. Los FPGAs (Field Programmable Gate Arrays) son dispositivos semiconductores que funcionan en torno a una matriz de bloques lógicos configurables (CLBs) conectados a través de interconexiones programables. Los FPGAs actuales pueden operar hasta una velocidad de 500 MHz, tienen bloques DSP (procesamiento digital de señales) para hacer operaciones matemáticas y su programación al ser estructural puede hacer operaciones en paralelo, esto es una gran ventaja para nuestro sistema ya que al mismo tiempo que este recolecta los datos de los sensores puede hacer el cálculo del cuaternión al mismo tiempo, también al ser de estructura abierta se pueden incorporar los módulos necesarios para los protocolos de comunicación que a diferencia de otros procesadores o controladores que están limitados por el número de puertos asignados para estas tareas. Debido a la arquitectura abierta y tipo de programación los FPGAs son el complemento ideal para muchas aplicaciones, tales como: aeroespacial, defensa, audio, centros de datos, seguridad, etc. Para el desarrollo del sistema AHRS se usa un FPGA Spartan 6-LX25 de la familia Xilinx incorporado en una tarjeta de desarrollo Xula 2 la cual se programó en la memoria flash que esta incorpora las características del FPGA y de la tarjeta de desarrollo se pueden observar en la tabla 3.2 y 3.3. La programación se hizo en VHDL en conjunto con bloques de System Generator de Matlab.

Características	FPGA Spartan 6 XL-25
Salida	16 bits con signo
Celdas Lógicas	24,051
Memoria	936 Kb
DSP Slices	38
Entradas/Salidas Maximas	266

Tabla 3.2: Especificaciones técnicas FPGA Spartan 6 XL-25 [23]

Características	Xula 2-XL25
Memoria	32 Mbyte SDRAM, 8Mbit Flash
Voltaje de alimentación	5v,3.3v y 1.2v
Pines configurables	30 I/O
Programación Slices	USB/ JTAG port

Tabla 3.3: Especificaciones técnicas de la tarjeta de desarrollo XULA-2 [24]

3.9. Instrumentación de módulo receptor GPS



Figura 3.6: GPS-635T.

EL GPS utiliza un protocolo de comunicación RS-232, para comunicarse con el FPGA se integra otro modulo de comunicación serial. El GPS actualiza la información de la posición cada segundo y es enviada al FPGA. La información que envía al FPGA no es controlada, el GPS envía los datos desde el momento que este es alimentado, el entandar que el GPS utiliza para enviar los datos es el NMEA (National Marine Electronic Asociaton) con un tipo de mensaje GPRMC (Recommended minimum specific data) y los datos que envía son: tiempo, fecha, posición, curso y velocidad. El trabajo del FPGA es clasificar estos datos, y enviarlos a la interfaz cuando esta lo requiera. En caso de que el GPS no detecte los satélites necesarios para calcular la posición este envía caracteres nulos lo que provoca que le FPGA envíe caracteres nulos a la interfaz. El GPS utilizado para este proyecto se muestra en la imagen 3.6.

3.10. Sistema de comunicación entre sistema de navegación y estación terrestre

La comunicación hacia la interfaz gráfica se hace mediante el protocolo de comunicación RS-232, utilizando un XBee S1 y un convertidor Serial- USB. Es decir, la comunicación se puede realizar con un cable a través del convertidor o de manera inalámbrica mediante el XBee dependiendo de las necesidades que se tengan.

Tanto el XBee como el convertidor comparten el mismo tipo de protocolo de comunicación, son fáciles de integrar dentro de la programación del FPGA ya que las señales de comunicación

CAPÍTULO 3. DISEÑO E IMPLEMENTACIÓN DEL SISTEMA DE NAVEGACIÓN

3.10. SISTEMA DE COMUNICACIÓN ENTRE SISTEMA DE NAVEGACIÓN Y ESTACIÓN TERRESTRE

son Rx y Tx. La comunicación se realiza Maestro esclavo y tanto el XBee como el convertidor son esclavos ya que es la estación terrestre la que inicia la comunicación.

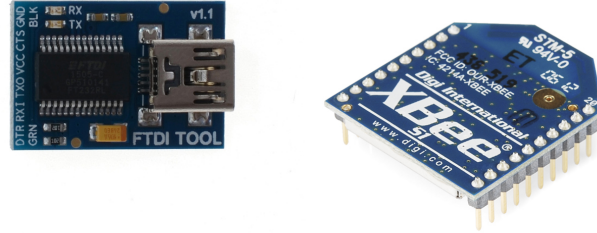


Figura 3.7: Dispositivos utilizados para la comunicación

De acuerdo a Digi [29], los módulos XBee son soluciones integradas que brindan un medio inalámbrico para la interconexión y comunicación entre dispositivos. Estos módulos utilizan el protocolo de red llamado IEEE 802.15.4 (Zigbee) para crear redes *FAST POINT-TO-MULTIPOINT* (punto a multipunto); o para redes *PEER-TO-PEER* (punto a punto). Están diseñados para aplicaciones que requieren de un alto tráfico de datos, baja latencia y una sincronización de comunicación predecible. Por lo que básicamente XBee es propiedad de Digi basado en el protocolo Zigbee.

Existen distintos tipos de series en los Xbee, en la tabla se muestra una comparación entre las diferentes series de XBee comerciales.

XBee	Velocidad	Frecuencia de banda	Antena	Rango
XBee 1mW	250 kbps	2.4 GHz	Whip	100m
XBee 2mW	250 kbps	2.4 GHz	chip,whip, pcb,rpsma	120m
XBee pro 50mW	250 kbps	2.4 GHz	rpsma,wire	1600m
XBee pro 60mW	250 kbps	2.4 GHz	rpsma,wire	1600m
XBee pro 900	156 kbps	900 MHz	rpsma	10 km

Tabla 3.4: Especificaciones técnicas diferentes series de XBee [29]

Las antenas Whip son cables que sobresalen del XBee, las antenas chip son un circuito integrado que actúa como antena, u.FL o PCB son conectores para colocar una antena propia y las RPSMA son conectores mas grandes para antenas con mayor tamaño (ver figura 3.8).

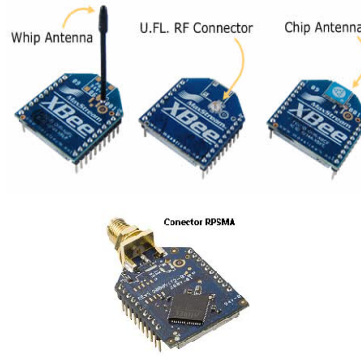


Figura 3.8: Diferentes antenas utilizadas en XBee

3.11. Batería de polímero de litio

Las baterías de polímero de litio, también denominadas baterías de Lipo son más livianas, almacenan más energía y retienen la carga por más tiempo que las baterías de níquel de un tamaño similar, lo que da lugar a una mayor autonomía de batería en un diseño más ligero. Además las baterías LiPo pueden cargarse en cualquier momento sin tener que esperar a que descarguen por completo y sin que se reduzca su vida útil, debido a que no sufren de efecto de memoria de carga.

Las baterías LiPo pueden entregar entre 3.7 y 22.2 volts (dependiendo del numero de celdas por las que este compuesta la batería) con una corriente de descargar de dos a treinta veces la capacidad de la celdas(dadas la tasa de descarga en amperes y la capacidad nominal en amperio/hora), cualidades que les permiten entregar altas dosis de potencia en regímenes de gran consumo. La corriente de descarga suministrada por una batería LiPo puede ser determinada empleando la relación dada por la ecuación 3.46

$$I = M \times C_n \quad (3.46)$$

donde I es la corriente descargada en (A), M es un múltiplo ó fracción de C , C es un valor numérico de la capacidad nominal de la batería en (Ah) y n es el tiempo en horas al cual C está declarado (ver [25] para más detalles en administración de energía).

El peso y tamaño de las baterías LiPo, las hace muy útiles para equipos pequeños que requieran potencia y duración; debido a estas cualidades se opta por usar este tipo de baterías en la presente aplicación. En la figura 3.9 se muestra las baterías LiPo de una celda utilizada para energizar el sistema AHRS y en la Tabla sus respectivas características.

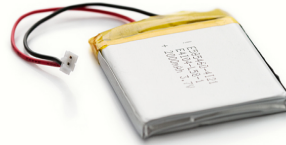


Figura 3.9: Batería de LiPo

Características	Batería LiPo
Capacidad nominal(C)	1000 mAh
Tasa de descarga	2C
Dimensiones	5,8 × 54 × 60

Tabla 3.5: Especificaciones técnicas de la batería de LiPo [26]

Para el presente trabajo de tesis se utilizan dos de estas baterías de Lipo puestas en serie para tener un voltaje de 7.4v el cual se regula a 5v mediante un módulo regulador de voltaje que incluye el circuito integrado Lm2596[27] que se puede observar en la figura3.10



Figura 3.10: Regulador de voltaje

3.12. Desarrollo de interfaz gráfica

La interfaz gráfica esta desarrollada en LabVIEW versión 2013, cuya función es visualizar la información de orientación del sistema AHRS (cuaternión de orientación estimado, vectores de

CAPÍTULO 3. DISEÑO E IMPLEMENTACIÓN DEL SISTEMA DE NAVEGACIÓN

3.12. DESARROLLO DE INTERFAZ GRÁFICA

velocidad, de aceleración y de campo magnético) y enviar un comando de habilitación del envío de datos al sistema AHRS. En la figura 3.11 se muestra el diagrama a bloques de la interfaz y en la tabla se muestran los requisitos mínimos necesarios para correr LabVIEW 2013.



Figura 3.11: Diagrama a bloques de la estación de monitoreo

En la figura 3.12 se muestra la interfaz desarrollada, ésta se compone de un menú en la parte superior, el cual se compone de una serie de botones para activar la visualización de los datos requeridos, cada medida de sensor y la salida del integrador cuenta con su propio visualizador.

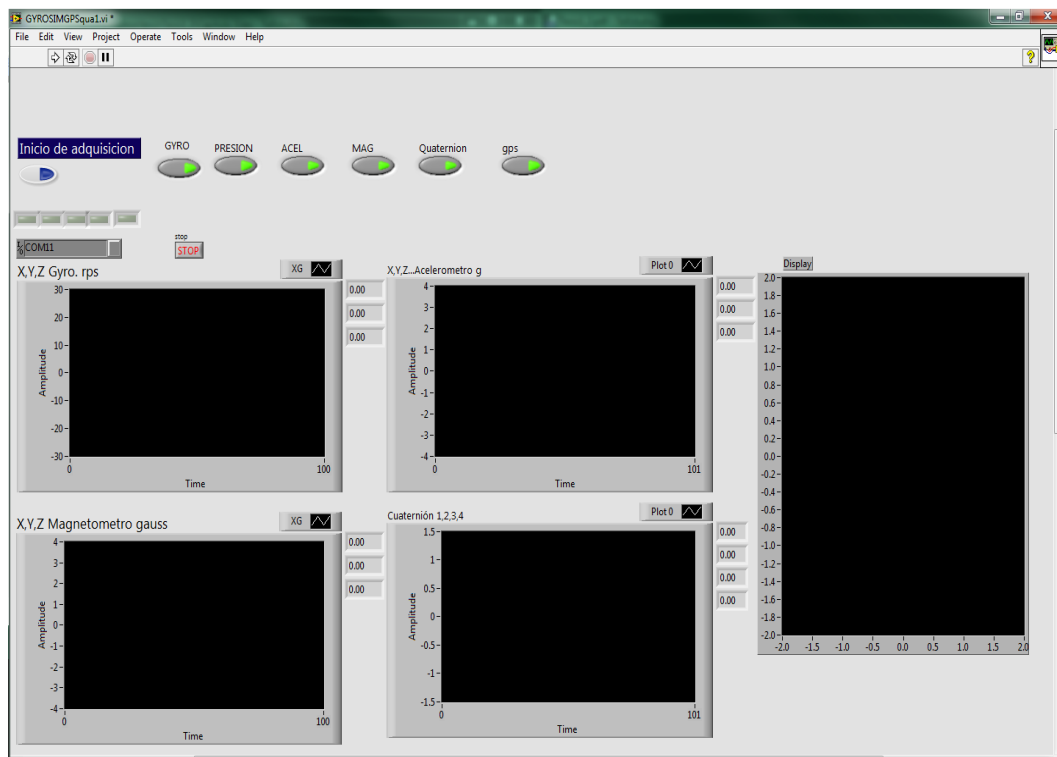


Figura 3.12: Interfaz Gráfica

Características	
Procesador	Pentium 4 o posterior
RAM	1 GB
Resolución de pantalla	1024 x 768 pixeles
SO	Windows xp o posteriores
Espacio en disco duro	5 GB

Tabla 3.6: Requerimientos mínimos para ejecutar LabVIEW

3.12.1. Integración de interfaz

Para observar el comportamiento del sistema se desarrolla un interfaz en LabVIEW cuya tarea es desplegar la información mediante gráficas tanto de los vectores de velocidad angular, aceleración, campo magnético y el del cuaternión estimado que son enviados por medio del protocolo de comunicación RS-232 al módulo XBee o al convertidor serial-usb. La información transmitida esta en varios formatos y se desconcatena dependiendo del tamaño de la variable enviada siendo el software de la interfaz la que los concatene de nuevo. Para recibir la información en la interfaz esta tiene que enviar un comando específico al FPGA que corresponda al dato requerido y así el FPGA empieza el envío del dato pedido por la interfaz, estos comandos enviados son letras en código ASCII, en la tabla 3.7 se muestra que letra en código ASCII corresponde a cada tipo de dato y en que formato se envía.

Para que el FPGA envíe cada dato requerido se tiene que dar un milisegundo de tiempo entre cada solicitud de un dato diferente o del mismo, es decir aproximadamente un baudio, esto para que el FPGA envíe bien los datos y no se sobrescriban con los de otros sensores no requeridos. También se tiene que tomar en cuenta la velocidad de procesamiento de la computadora que se utilice, ya que de esto dependerá la velocidad a la que se muestran los datos y por lo tanto el número de datos requeridos al FPGA por segundo. Una vez que la información llega a la interfaz gráfica es mostrada en los indicadores gráficos correspondientes y al final se envía un dato de confirmación para saber si los datos llegaron correctamente a la interfaz, en caso de que no lleguen completos a la interfaz estos no se muestran y se pasa a la siguiente iteración. En la figura 3.12 se muestra la interfaz gráfica realizada en LabVIEW.

Carácter enviado	Dato correspondiente	Tipo de variable	Orden en que se reciben los datos
a	Velocidad Angular	int16	VelAngX MSB VelAngX LSB VelAngY MSB VelAngY LSB VelAngZ MSB VelAngZ LSB
b	Aceleración	int 16	AceX MSB AceX LSB AceY MSB AceY LSB AceZ MSB AceZ LSB
c	Magnetómetro	int 16	MagX MSB MagX LSB MagY MSB MagY LSB MagZ MSB MagZ LSB
f	Cuaternión	Punto flotante	Cua1(bit 25-32) Cua1(bit 17-24) Cua1(bit 9-16) Cua1(bit 1-8) Cua2(bit 25-32) Cua2(bit 17-24) Cua2(bit 9-16) Cua2(bit 1-8) Cua3(bit 25-32) Cua3(bit 17-24) Cua3(bit 9-16) Cua3(bit 1-8) Cua4(bit 25-32) Cua4(bit 17-24) Cua4(bit 9-16) Cua4(bit 1-8)

Tabla 3.7: Envío y recepción de datos de la interfaz

3.13. Integración del sistema de navegación

En esta sección se describe la integración tanto de software como hardware para la formación del sistema AHRS. El sistema AHRS está formado por la batería de LiPo, la IMU AltIMU-10, el módulo XBee, el convertidor serial- USB y el procesador digital de señales (FPGA Spartan 6 XL-25). En la figura 3.13, se muestra como está integrado cada componente mencionado, así

como el flujo de información y energía entre estos.

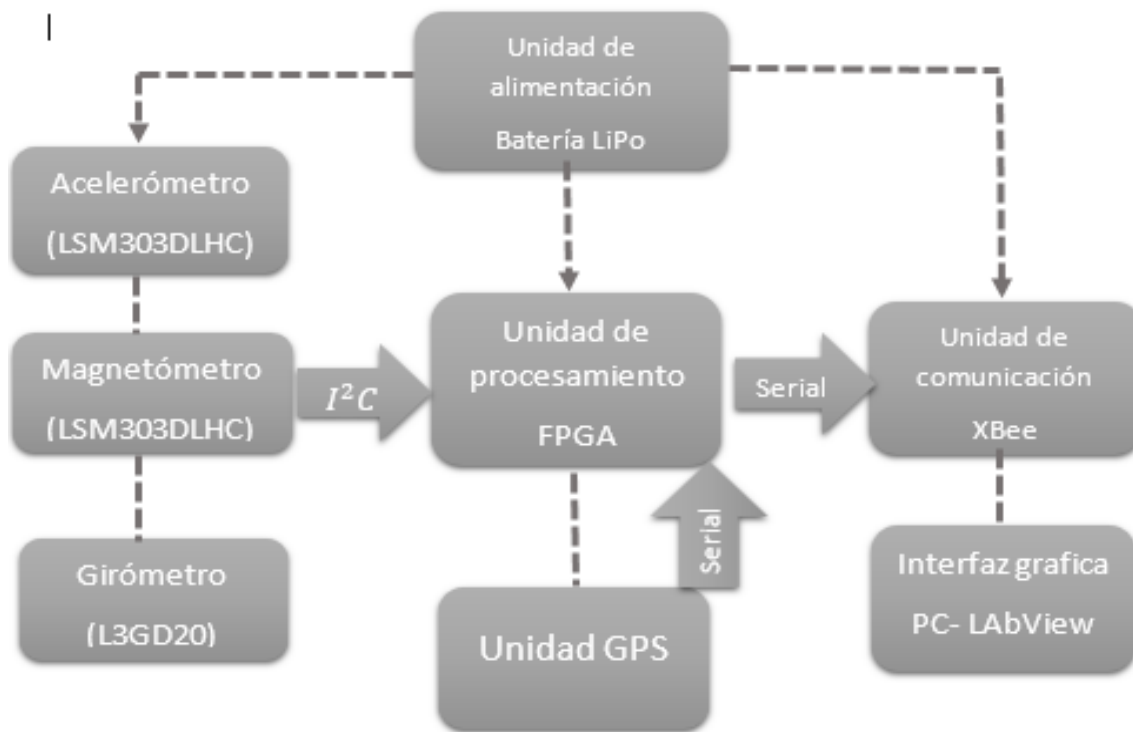


Figura 3.13: Diagrama a bloques del sistema de navegación

En el sistema AHRS el FPGA se encarga de configurar los diferentes sensores y recibir la información recolectada por estos. Una vez que los datos de los sensores son adquiridos, se obtiene el cuaternión de orientación estimado. Para visualizar como se encuentra orientado el sistema AHRS, la estación de monitoreo manda un comando de habilitación de envío de información. Este comando se transmite por RS-232 utilizando el XBee o el convertidor serial-USB. Cuando este comando es recibido por el FPGA se transmite la información requerida por el usuario que puede ser el cuaternión o las medidas de los sensores dependiendo del comando enviado. En la figura 3.14 se muestra el diagrama de flujo de los procesos mencionados anteriormente que realiza el FPGA.

3.13.1. Adquisición de datos de los sensores

En la presente aplicación se emplean sensores de salidas digitales utilizando un mismo bus de datos mediante el protocolo de comunicación I^2C . Debido a esto se implementa una librería para este tipo de comunicación en el programa de VHDL y poder así establecer la comunicación con los sensores. Es necesario configurar cada tipo de sensor mediante la escritura de sus registros principales.

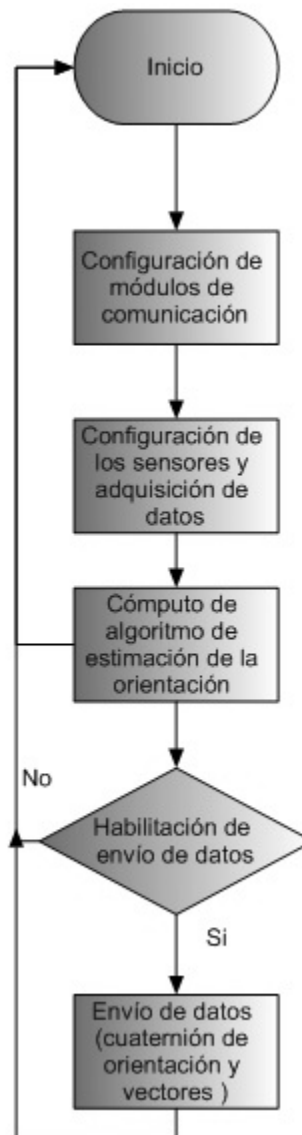


Figura 3.14: Diagrama de flujo del software

Para comunicarse con cada sensor de la IMU es necesario iniciar la comunicación desde el FPGA que se comportara como maestro y todos los sensores como esclavos. Una vez iniciada la comunicación con el sensor correspondiente se tiene que poner en modo de escritura para que se pueda escribir en sus registros y así configurarlos, una vez configurados se pone en modo de lectura para leer los datos del sensor. Los datos correspondientes a la medición son de 16 bits de tamaño en complemento a 2. Para el girómetro, se tiene que escribir en el registro 1_G (con dirección en el banco de registros 20h) el dato (00001111b) para que la configuración sea la siguiente:

- Actualización de datos a 95 Hz.
- Modo normal de consumo de energía
- Todos lo ejes sensibles activados

- Rango de medición de $\pm 250^\circ/s$

En el caso del acelerómetro, se escribe en el registro 1_A (con dirección 20h) el dato (01010111b) para que la configuración sea la siguiente:

- Actualización de datos a 100 Hz.
- Modo normal de consumo de energía
- Todos lo ejes sensibles activados
- Rango de medición de $\pm 4g$

Por último en el magnetómetro se escribe en el registro 1_M (con dirección 00h) el dato (00011000b) para que la configuración sea la siguiente:

- Actualización de datos a 75 Hz.
- Modo normal de consumo de energía
- Todos lo ejes sensibles activados
- Rango de medición de $\pm 4\text{gauss}$

Una vez que se obtiene los datos de los respectivos sensores estos son guardados dentro del FPGA en registros, después de esto se envía a un convertidor a punto flotante construido en System Generator de MatLab que se puede ver en la figura 3.15. Por ultimo, se multiplica por una constante para que después entre en el cálculo del cuaternión de orientación.

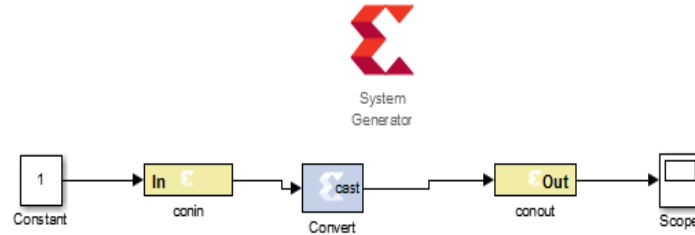


Figura 3.15: Diagrama de flujo del software

3.13.2. Computo del cuaternión de orientación

Una vez que se obtiene todos los datos de los sensores guardados en sus respectivos registros. Primero, se obtiene la Matriz de observación $\bar{H}$ utilizando los vectores de referencia (r_a y r_m) y los vectores de observación (b_a) y (b_m) provenientes de las mediciones de acelerómetros y magnetómetros, quedando de la siguiente forma:

$$\bar{H} = \begin{bmatrix} H_a \\ \vdots \\ H_m \end{bmatrix} = \begin{bmatrix} 0 & -(\vec{b}_a - \vec{r}_a)^T \\ \vec{b}_a - \vec{r}_a & -[(\vec{b}_a + \vec{r}_a)^\times] \\ \vdots & \vdots \\ 0 & -(\vec{b}_m - \vec{r}_m)^T \\ \vec{b}_m - \vec{r}_m & -[(\vec{b}_m + \vec{r}_m)^\times] \end{bmatrix} \quad (3.47)$$

Aquí se aplica el algoritmo resumido en la ecuación (3.26) para encontrar qps , después encontramos el cuaternión estimado $\hat{q}$ y el bias estimado $\hat{v}$ integrando las ecuaciones del ONL mediante el método de Runge-Kutta [19]:

$$O_A = \begin{cases} \dot{\hat{q}} = \frac{1}{2}\Xi(\hat{q})[\vec{\omega}_G - \hat{v} + \text{sign}(q_{e0})K_1\vec{q}_e] \\ \dot{\hat{v}} = -T^{-1}\hat{v} - \text{sign}(q_{e0})K_2\vec{q}_e \end{cases} \quad (3.48)$$

Finalmente se calcula el cuaternión de error q_e multiplicando el conjugado del cuaternión estimado $\hat{q}^{-1}$ con el cuaternión de orientación q_{ps} . Todas las constantes de las ecuaciones anteriormente mencionadas son definidas en la sección 3.1.

También se crean los módulos para hacer operaciones aritméticas con punto flotante ya que VHDL no cuenta con este tipo de operación en las librerías disponibles, todas estos módulos se implementan y se reciclan para el cálculo del cuaternión.

3.13.3. Diseño del PCB

Finalmente, se fabrica un PCB que contenga al FPGA y a la IMU el cual esta diseñado en el software ISIS de Proteus, el diseño se muestra en la figura 3.16

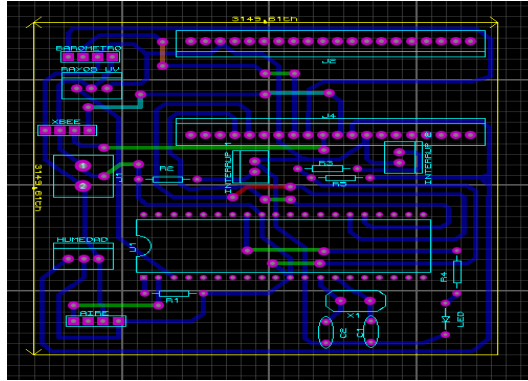


Figura 3.16: Diseño de PCB

En la siguiente figura se puede ver el sistema implementado al igual que los componentes.

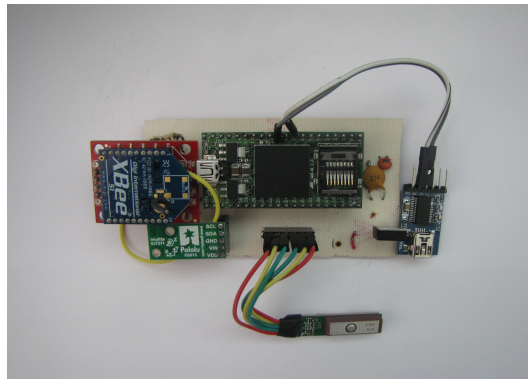


Figura 3.17: PCB

3.14. Resultados experimentales

El observador no lineal descrito en la sección 3.5 se programa en un diagrama a bloques en Simulink de MATLAB para comprobar que este efectivamente calcula el error generado por la medida de la velocidad angular proveniente del girómetro y a su vez para ver la convergencia entre el valor del cuaternión estimado y el valor del cuaternión real. Se simula en MATLAB la estimación de la orientación colocando un ofset conocido en el vector de las velocidades angulares ω el cual tiene como valores $[0,01 \ 0,01 \ -0,05]$ estos valores son a los cuales tiene que llegar la estimación del error producida por el observador. Con estos parámetros bien definidos se puede ver en la figura 3.19 como la estimación del error es coincide de manera muy aproximada a los valores propuestos como ruido.

También se puede ver en la figura 3.20 la convergencia entre el cuaternión real denotado por el color amarillo y el cuaternión estimado por el observador no lineal denotado de color morado. Por lo tanto, se puede decir que el algoritmo para calcular la actitud de un cuerpo rígido funciona de manera correcta.

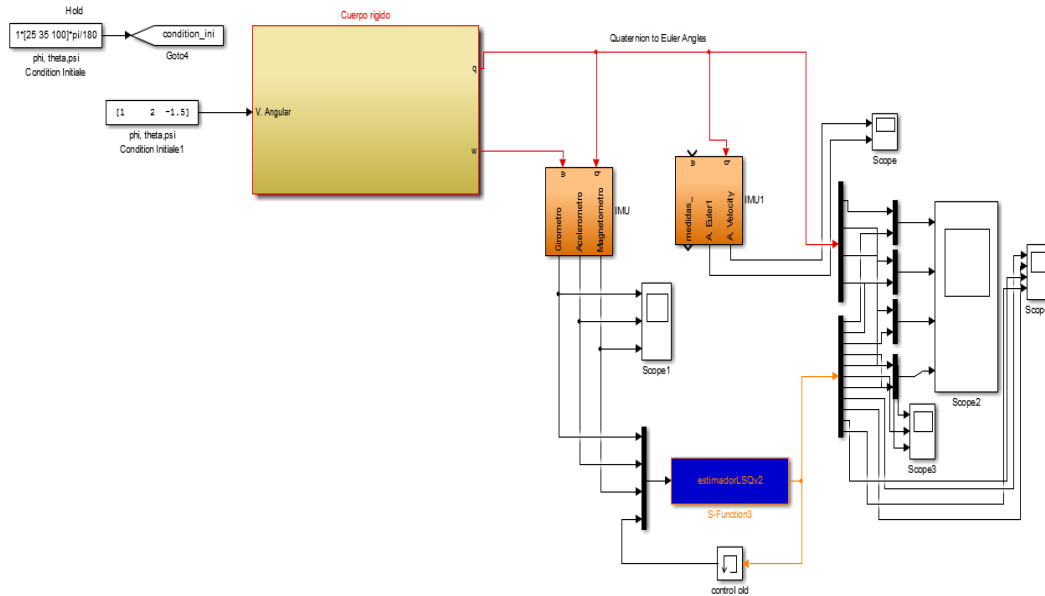


Figura 3.18: Diagrama a bloques en MatLab del observador no lineal

Se implementó el observador no lineal en el FPGA, pero debido a la complejidad del sistema el compilador no logró sintetizar de una manera óptima el software creado en VHDL y por lo tanto, no funcionó de una manera correcta. En la figura 3.21 se puede observar el *warning* que arroja la compilación, el cual hace que el observador no funcione correctamente ya que no se satisface las restricciones de tiempo por lo tanto no se sincronizan las salidas y el resultado no es optimo.

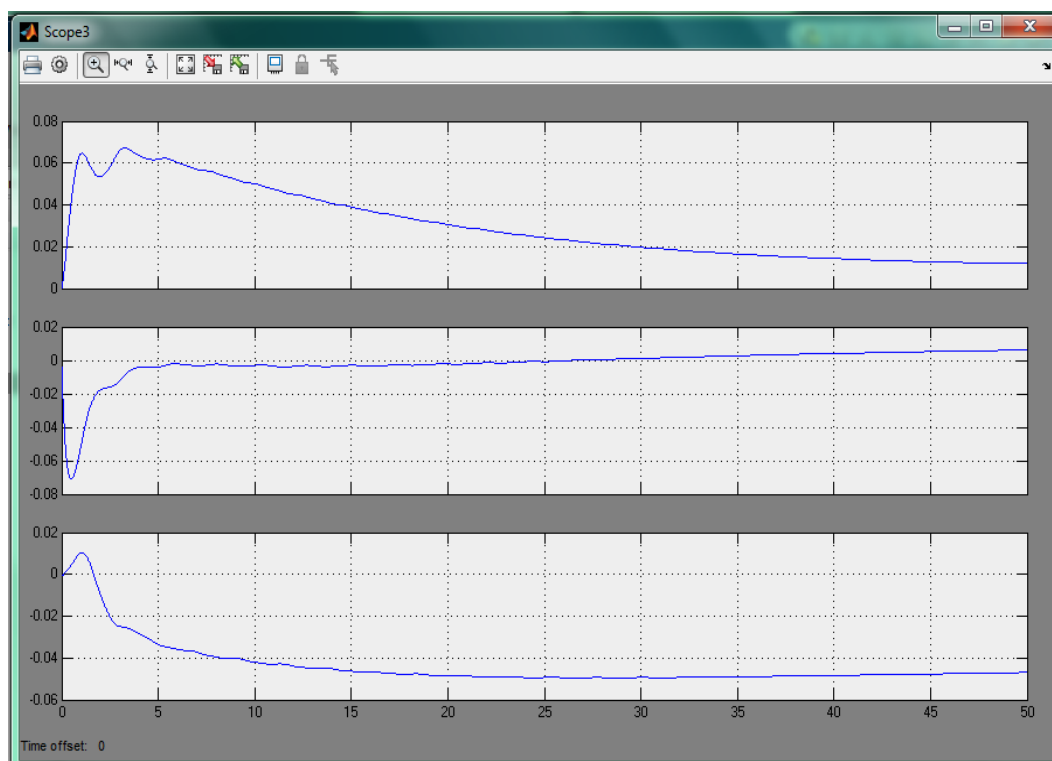


Figura 3.19: Gráfica del error estimado

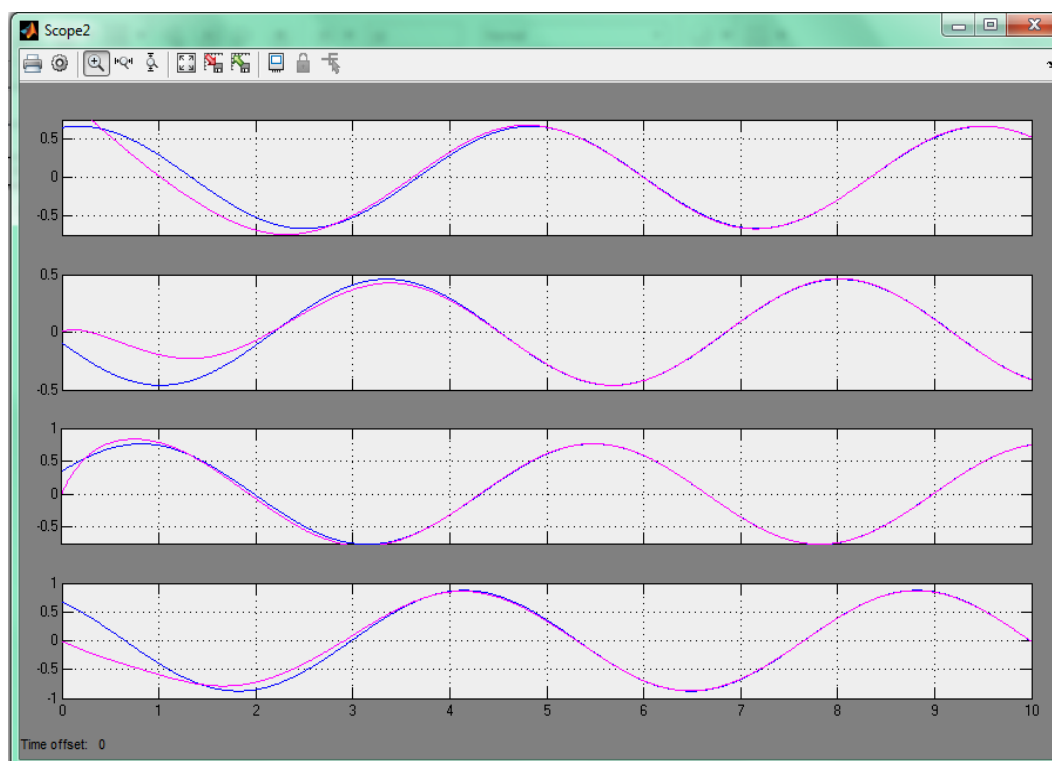


Figura 3.20: Gráfica comparativa entre el cuaternión estimado de color morado y el cuaternión real de color azul

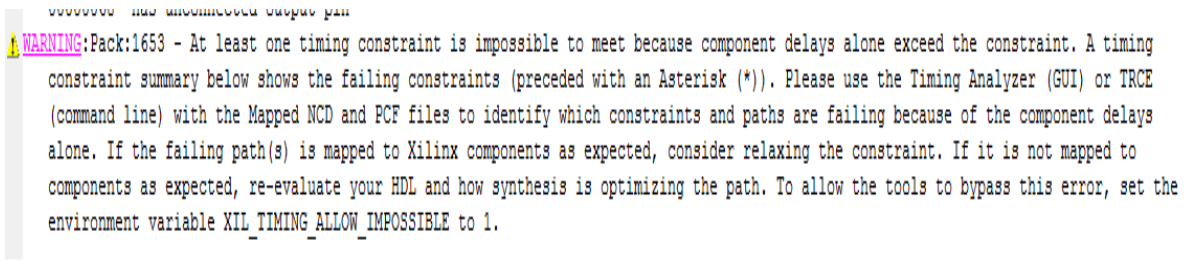


Figura 3.21: Advertencia en el compilador

Una vez considerando esto se procedió a probar el integrador RK4 colocando en la entrada velocidades angulares constantes de valor de $(-0,25)$ en cada eje y se graficaron en MatLab comparándolas con un integrador de matlab que utiliza el mismo método de integración. En la figura 3.22 se puede ver la comparación de las señales procedentes del integrador de MatLab y del integrador del FPGA. La gráfica siguiente muestra el comportamiento del integrador programado en el FPGA y el integrador de MatLab, ya que el cuaternión es un ente que representa el movimiento de un cuerpo rígido en un mapa en 3D con magnitud de 1, no tiene unidades. Se grafica con respecto a pasos de integración o iteraciones.

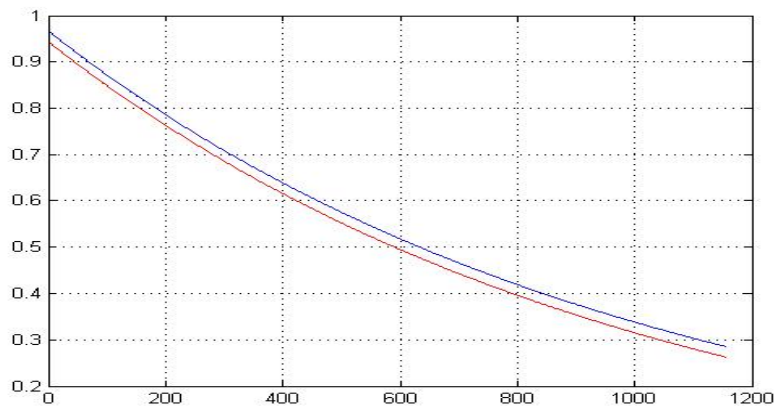


Figura 3.22: Comparación q_0 teórico y practico

La comparación se hace con el cuaternión q_0 y se puede observar claramente un desvío de los datos entre el cálculo del FPGA en color azul y el arrojado por Matlab de color rojo, esto se debe a que se utilizan variables de punto flotante las cuales aunque pueden representar un número de una manera aproximada, no siempre es exacta por lo tanto existe un pequeño error presente en el cálculo. En la figura 3.23 se puede observar la comparación del vector q_2 que en este caso sería el mismo que los demás vectores ya que la velocidad angular es la misma en todos los ejes, también se puede observar un desvío entre el valor teórico y el valor que arrojó el FPGA.

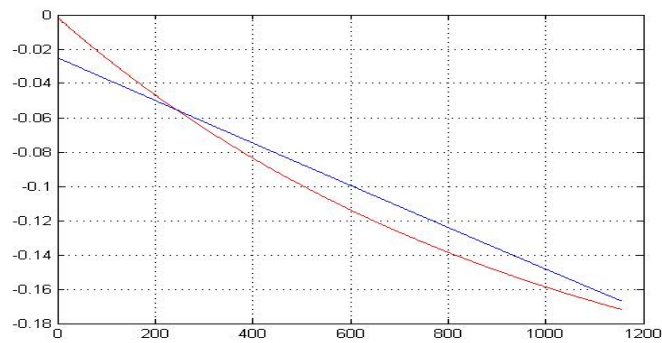


Figura 3.23: Comparación q_2 teórico y práctico

En el caso del GPS se probó mediante la interfaz de LabView colocando los datos en un buscador embebido dentro del software y así dar una imagen del lugar en donde nos encontramos tomando las coordenadas dadas por el GPS y procesadas por el FPGA.

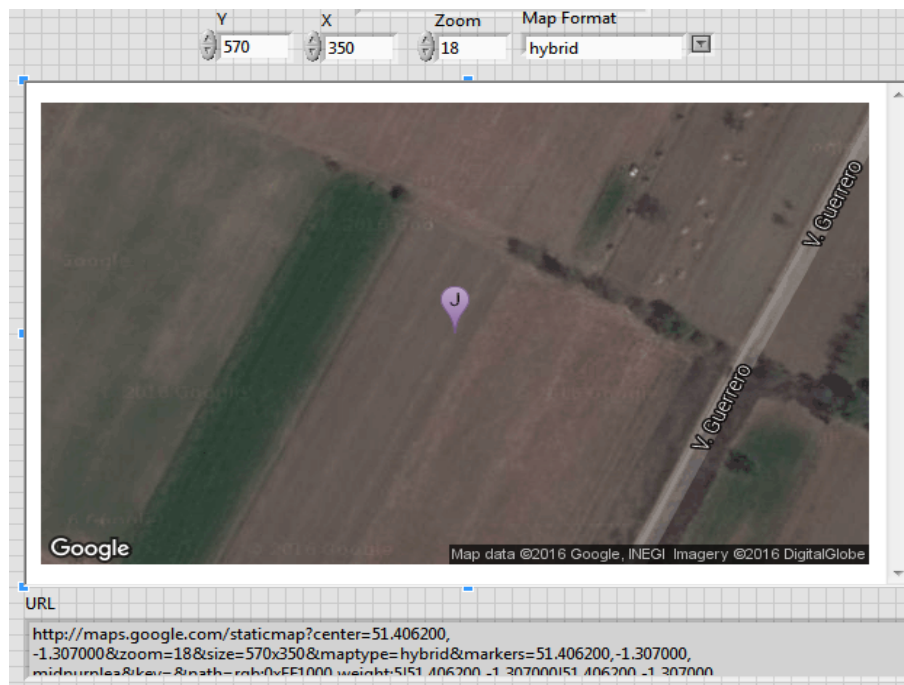


Figura 3.24: Imagen de ubicación dada por el GPS

En la figura 3.24 se observa la visualización de los datos proporcionados por el GPS, cabe señalar que se necesita una conexión de Internet para obtener la imagen de satélite esto es porque se utiliza la aplicación de Google Maps como un servicio embebido dentro de LabView.

Capítulo 4

Implementación de sistema de monitoreo

En este capítulo se menciona algunas de las tendencias de los sistemas de monitoreo ambiental así como sus aplicaciones para tener una mejor idea del diseño que se le da al sistema construido, también se describe la integración de cada sensor que se utiliza en el sistema de monitoreo, también la integración del módulo bluetooth y del Microcontrolador Pic 18F4550 utilizado para los sensores analógicos. Un diagrama a bloques del sistema implementado se muestra en la figura 4.2 y la descripción de cada uno de los componentes, así como su integración en el sistema. Finalmente, se mostrarán los resultados obtenidos y se compararan con un sistema ya existente dentro de la misma universidad.

Los sistemas de monitoreo ambiental son utilizados en gran parte por meteorólogos, ya que estudian el estado del tiempo y los fenómenos creados a partir de esto mediante modelos. En la meteorología se aplican para acopiar constantemente un máximo de datos sobre el estado del tiempo y la atmósfera, analizarlos, interpretarlos y obtener así deducciones sobre el futuro del tiempo.

Con el surgimiento de nuevas tecnologías en el desarrollo de sensores, haciendo de la mayoría de estos sistemas digitales embebidos se puede lograr un gran alcance en cuanto a las variables medioambientales medidas en un solo dispositivo lo que da el paso a desarrollos de sistemas de monitoreo más pequeños y económicos como el caso de las estaciones portátiles desarrolladas por *Kestrelmeters* [2], estas estaciones tienen una gran cantidad de parámetros medidos, sin embargo no están contruidos para ser montados sobre vehículos autónomos pero si da una clara idea de la tendencia de los nuevos sistemas de monitoreo al hacer estos más portables.

Teniendo en cuenta este tipo de sistemas portátiles se decide hacer este sistema de monitoreo ambiental de una forma que pueda ser montado sobre vehículos no tripulados y a continuación se muestra esta integración y construcción del sistema.



Figura 4.1: Estación de monitoreo ambiental portátil de la empresa Kestrelmeters

4.1. Integración del sistema de monitoreo ambiental

En esta sección se describe la integración entre los componentes de software y hardware, es decir, la integración entre los distintos dispositivos utilizados en la implementación del sistema de monitoreo ambiental.

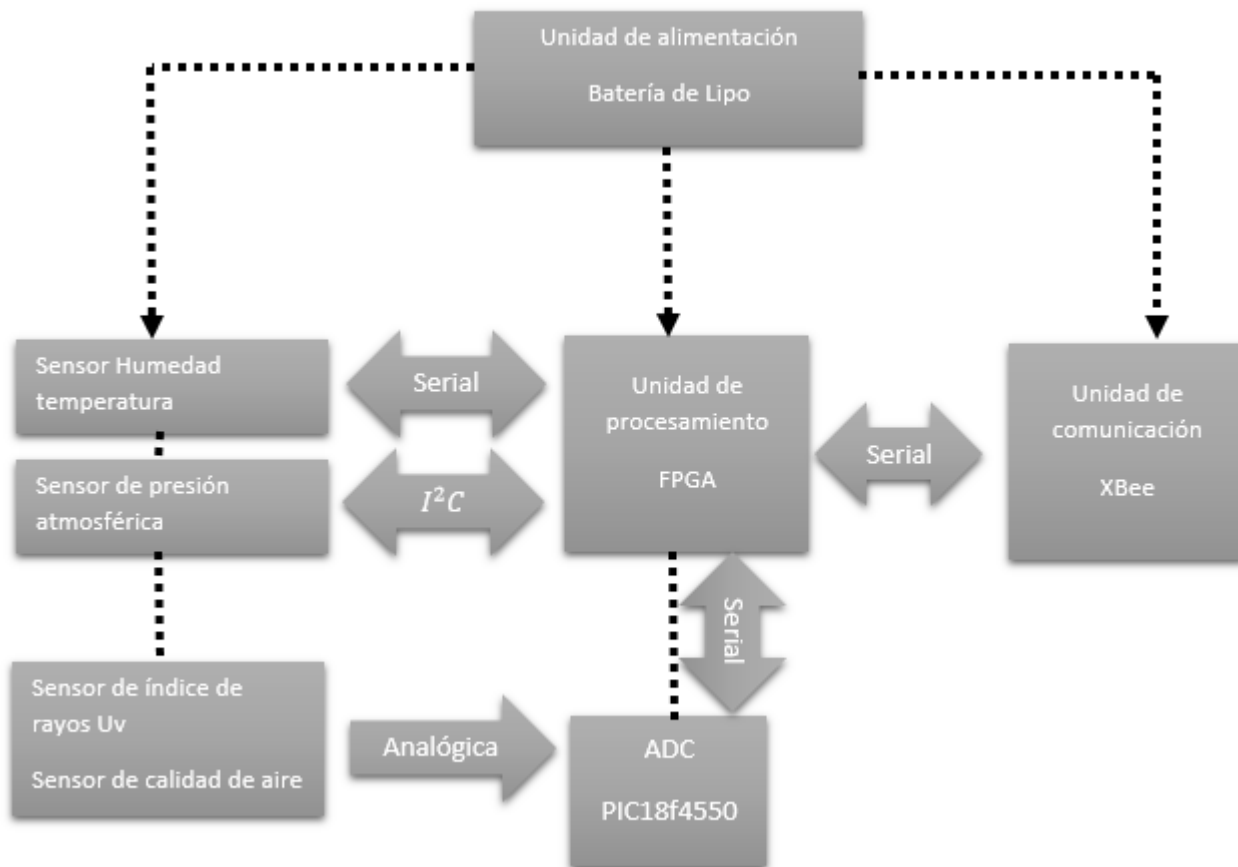


Figura 4.2: Integración del sistema de monitoreo

El sistema de monitoreo ambiental esta formado por la batería de LiPo, sensor de humedad-temperatura (DHT11), indice de rayos Uv, sensor de presión atmosférica (BMP180), sensor de calidad del aire, el módulo bluetooth, convertidor ADC (pic18f4550) y el controlador digital de señales (FPGA). En la figura 4.2 se muestra dicha integración, así como la dirección del flujo de la información. En el sistema de monitoreo ambiental el FPGA se encarga de recibir toda la información de los sensores. Una vez que los datos de los sensores son adquiridos y guardados en registros dentro del FPGA, la estación en tierra manda un comando de habilitación para la transmisión de datos, este comando se transmite de forma inalámbrica utilizando el bluetooth. Cuando el comando es recibido por el sistema de monitoreo este responde con el envío de datos de los sensores empleando el protocolo RS-232. En la figura 4.3 se muestra el diagrama de flujo de los procesos mencionados que realiza el FPGA.

4.1.1. Adquisición de los datos de los sensores

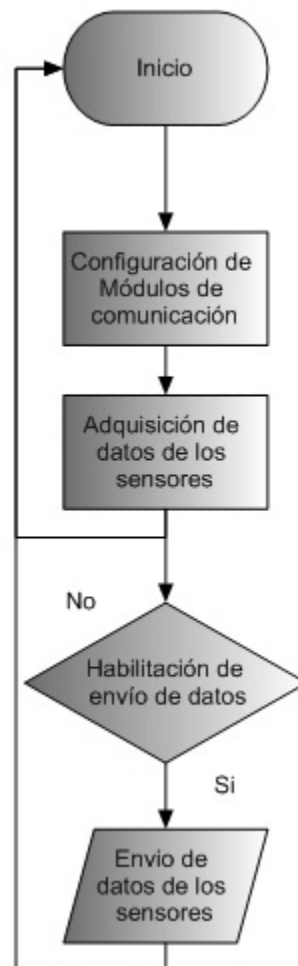


Figura 4.3: Diagrama de flujo del sistema de monitoreo

En el monitoreo ambiental se emplean dos tipos de sensores: sensores analógicos (rayos Uv y calidad de aire) y sensores digitales (Humedad, temperatura y presión atmosférica).

CAPÍTULO 4. IMPLEMENTACIÓN DE SISTEMA DE MONITOREO

4.1. INTEGRACIÓN DEL SISTEMA DE MONITOREO AMBIENTAL

Debido a esto se emplean diferentes librerías en VHDL y un PIC 18f4550 para la obtención de los datos provenientes de los sensores. Para los sensores analógicos se utiliza un convertidor analógico-digital incorporado en el PIC 18F4550 configurado a 8 bits y una frecuencia de muestreo del pic a 100 Hz. Para la conversión a palabra digital de los sensores analógicos se utilizan dos canales del PIC 18f4550 uno por cada sensor. El PIC se comporta como esclavo ya que el FPGA envía un comando específico al PIC para que este recopile la información y responda al FPGA transmitiendo los datos de los sensores por medio de RS-232. Finalmente, el FPGA guarda estos datos en registros internos. En el caso del sensor de Humedad-temperatura una vez que se logra establecer la comunicación con este, los datos son enviados de manera digital y se guardan en registros del FPGA. Para el caso del sensor de presión se establece una comunicación mediante el protocolo de comunicación I^2C y el dato proporcionado por el sensor es de 24 bits que al igual que los demás datos se guardan en registros del FPGA.

4.1.2. Comunicación con la estación de monitoreo

Como se mencionó en la sección 4.7 la interfaz se encarga de enviar un comando específico para que el FPGA transmita los datos almacenados en sus registros, entonces la función principal del FPGA es recibir los datos enviados por los sensores y cuando la estación en tierra lo requiera enviar estos datos por medio del protocolo RS-232 al módulo Bluetooth, el cual se encarga de comunicarse con la estación en tierra.

Numero de Byte	Información representada
1	Calidad del Aire int8
2	Indice de rayos Uv 8
3	Humedad int 8
4	Temperatura int 8
5	check sum int 8
6	Presión atmosférica MSB
7	Presión atmosférica LSB

Tabla 4.1: Transmisión de datos del sistema de monitoreo ambiental

La información transmitida concerniente a cada dato censado esta representada en diferentes formatos, en el caso de los sensores de humedad, temperatura, calidad del aire y rayos Uv el dato es representado por un entero sin signo de 8 bits y en el caso del barómetro la palabra es un entero sin signo de 16 bits conformado por LSB y MSB, además de que se envía el check sum del sensor de humedad y temperatura para saber si la información se recopiló correctamente que es de 8 bits sin signo por lo que el FPGA envía un total de 8 bytes, cuya estructura se puede ver en la tabla. La información solo se transmite cuando se envía la letra *a* en código

ASCII al FPGA.

4.1.3. Diseño del PCB y ensamble

El diseño esta realizado en el software Proteus y el área del PCB queda de una dimensión de $10 \times 10\text{cm}$ y se puede observar en la imagen. 4.4.

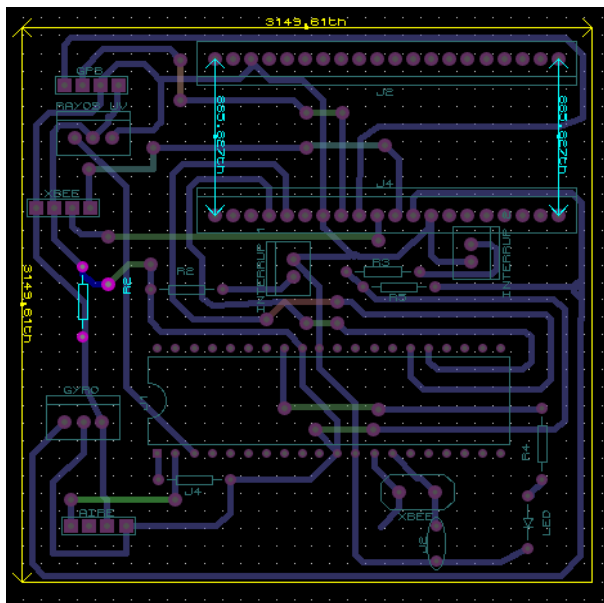


Figura 4.4: Diseño de PCB para sistema de monitoreo ambiental.

Finalmente, el PCB esta montado en un contenedor diseñado para ser montado sobre un vehículo no tripulado el cual se puede observar en la figura 4.6 y 4.5.

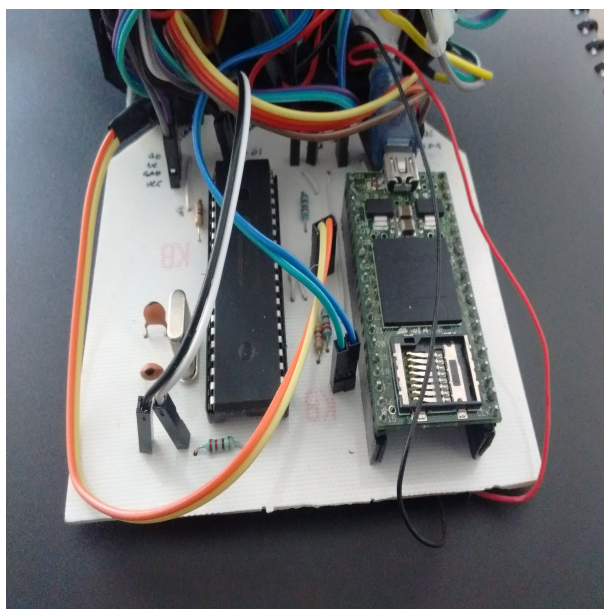


Figura 4.5: Sistema de monitoreo ambiental



Figura 4.6: Sistema de monitoreo ambiental

4.2. Instrumentación de sensor de humedad y temperatura

El sensor que se utiliza para medir estas dos variables es el DHT11 y está integrado en un solo PCB que se puede observar en la imagen 4.7, su alimentación es de 3.3 V a 5 V, este integrado incluye un componente de medición de humedad de tipo resistivo y un componente de medición de temperatura NTC, los cuales se conectan a un microprocesador de 8 bits para enviar la salida digital.

Cada elemento está estrictamente calibrado en laboratorio, los coeficientes de calibración se almacenan como programas en la memoria del microprocesador del sensor. Este tiene un protocolo de comunicación propio el cual es de tipo serial y utiliza solo un bus de datos para enviar las dos mediciones de los sensores, además tiene un bajo consumo de energía.

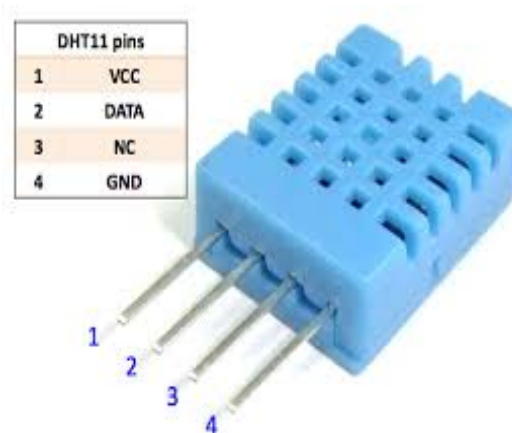


Figura 4.7: Sensor Humedad y temperatura.

La salida se da por el pin 2 y esta señal tiene que ir conectada a voltaje positivo de alimentación con una resistencia de $4.7\text{ K}\Omega$ y a su vez alambrado a un pin del FPGA el cual se encargara de recopilar la información proporcionado por el sensor.

Para la comunicación con el sensor se tiene que seguir el siguiente protocolo:

1. **Solicitud:** Para hacer que el DHT11 envíe las lecturas de los sensores, el FPGA tiene que enviar una señal de solicitud la cual es mantener el bus de datos en estado bajo durante 18ms y después mantener en alto la señal durante $40\mu\text{s}$, para esto el pin del dispositivo receptor de datos en nuestro caso el FPGA se tiene que poner en modo de salida.
2. **Respuesta** Una vez realizada la solicitud el DHT11 envía una señal de respuesta indicando que recibió la solicitud de información. En respuesta el DHT11 mantiene la señal en bajo durante $54\mu\text{s}$ y después en alto durante $80\mu\text{s}$, para esto el pin del FPGA se tiene que poner en modo entrada y detecta esta señal de respuesta.

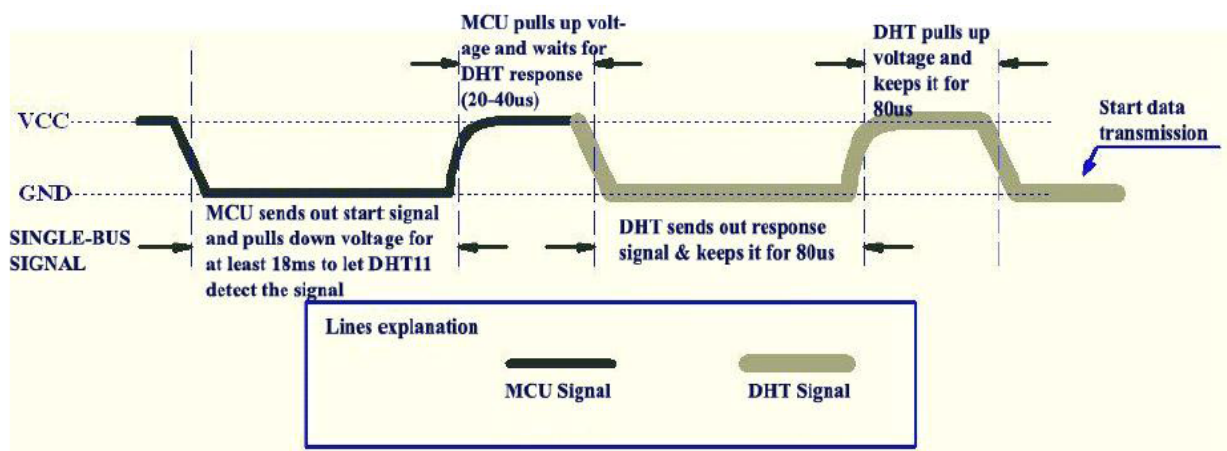


Figura 4.8: Inicio de comunicación con DHT11.

3. **Lectura de datos:** después de la respuesta del DHT11 este envía los datos de los sensores en un paquete de 5 segmentos de 8 bits cada uno. Cuando el DHT11 mantiene la señal en un estado alto durante $24\mu\text{s}$ y un uno es cuando la señal se mantiene en alto durante $70\mu\text{s}$ cada bit es separado por un estado en bajo que dura $54\mu\text{s}$ (ver figura 4.9 y 4.10), al igual que en el paso anterior el pin del FPGA se mantiene en modo entrada para recibir estos datos.

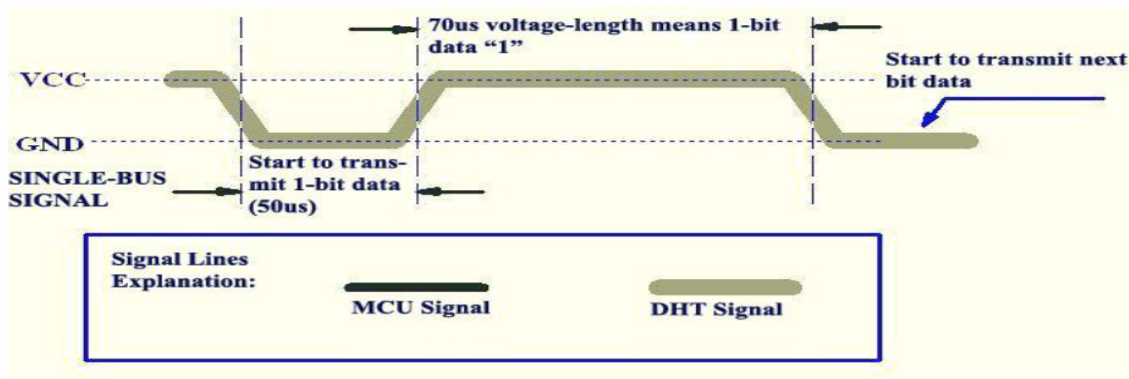


Figura 4.9: Bit a uno enviado por el DHT11

Cada segmento esta en complemento a dos y se recibe de la siguiente manera: primero, se reciben los datos de la humedad entero y decimal, este dato esta dado en porcentaje de humedad, después se reciben los datos de la temperatura igualmente llegan primero la parte entera y luego la parte decimal. Por último, llegan el check sum el cual es la suma de todos los datos recibidos y sirve para verificar que la información llego correctamente.

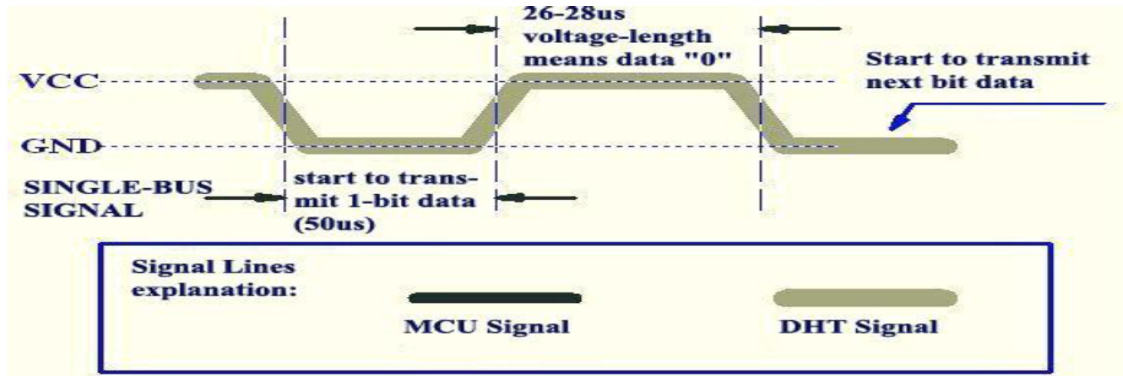


Figura 4.10: Bit a cero enviado por el DHT11

4.3. Instrumentación de sensor de calidad del aire

Este sensor es adecuado para la detección de gases como NH_3 (amoníaco), NO_x (óxidos de nitrógeno), alcohol, CO_2 y humo, estos gases se miden en partes por millón de la muestra detectada y envía una señal analógica a su salida proporcional a esta medición pero debido a que este sensor no distingue entre tipos de gases si no que aumenta su salida al detectar la presencia de cualquiera de estos gases se puede decir que cuando aumenta su salida significa que la calidad del aire es mala por lo tanto, se puede hacer una escala de la calidad del aire dependiendo de su salida, el sensor muestra en la figura 4.11.



Figura 4.11: Sensor de Calidad de Aire

El sensor se alimenta a 5 V y debido a que la salida de este sensor es analógica de 0 a 5v y el FPGA no cuenta con este tipo de entradas se utiliza un PIC 18F4550 con entradas analógicas el cual convierte esta medición a una palabra digital de 8 bits y esta es enviada al FPGA por comunicación serial.

4.4. Instrumentación de sensor de rayos UV

Este sensor se alimenta a 5 V y también cuenta con una salida analógica de 0 a 5v para lo cual se utiliza una de las entradas analógicas del Pic 18F4550 el cual vuelve digital esa lectura y la envía al FPGA por comunicación serial. Una vez que el FPGA recibe esta señal se multiplica por una constante para convertirlo a nivel de radiación ultravioleta. La constante esta dada por la siguiente relación: como la palabra digital formada por el pic es de 8 bits entero entonces podemos representar del numero 0 al 255 decimal y la radiación capaz de medir de este sensor es de 0 a 11 en indice de radiación Uv y el sensor se comporta linealmente en ese rango de medición (ver figura 4.12) entonces solamente se divide 11 entre 255 para tener la constante de conversión a indice de rayos Uv.

Como se puede observar en la figura 4.12 el comportamiento del sensor dentro del rango de medición que corresponde a la longitud de onda de la luz ultravioleta es lineal. El sensor se puede observar en la figura 4.13

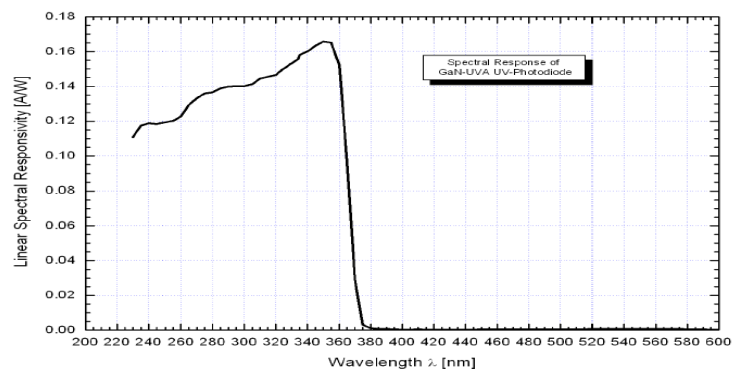


Figura 4.12: Comportamiento de sensor de rayos Uv.

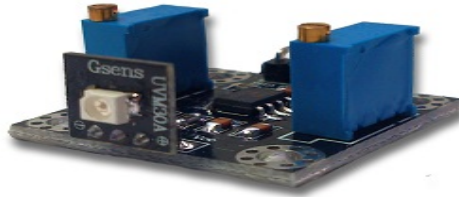


Figura 4.13: Sensor de rayos Uv.

4.5. Instrumentación de sensor de presión atmosférica

El sensor utilizado es el BMP180 de la empresa Bosch que se observa en la figura 4.14 el cual tiene un rango de medición de 300 a 1100hPa y se alimenta a 5 V. Tiene una salida digital de 24 bits que se transmite por el protocolo I^2C



Figura 4.14: Sensor de presión atmosférica.

El BMP180 tiene una configuración determinada que actualiza la información cada 76.5ms. El PCB que incluye a este sensor contiene las resistencias necesarias para establecer la comunicación I^2C por lo tanto solo se alambra directamente con el FPGA que tiene en su programador el módulo específico para este sensor.

4.6. Descripción del módulo Bluetooth utilizado

El envío de datos provenientes del sistema de monitoreo a la estación terrestre se hace a través de una conexión inalámbrica por medio de un módulo bluetooth. El módulo que se uso para hacer esta conexión es el *Bluetooth Mate Gold* mostrado en la figura 4.15. Este módulo presenta un bajo consumo de potencia y soporta diversos protocolos de comunicación como: UART y USB. Este módulo bluetooth es fácil de usar, ya que todas las capas del protocolo bluetooth son transparentes para el usuario, quedando a éste las señales Tx y Rx . Además, se

puede usar este módulo como maestro o esclavo. En el presente trabajo se empleó a tal módulo bluetooth en configuración esclavo, ya que es la estación terrestre quién solicita a éste establecer la comunicación.

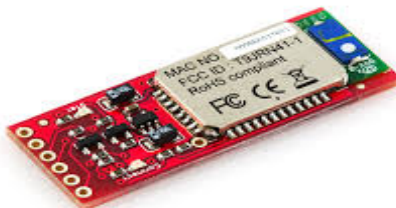


Figura 4.15: Modulo Bluetooth.

El bluetooth se alimenta a 5 V y tiene un alcance de 100 metros de transmisión de datos, para mas detalles sobre las características ver[22].

4.7. Batería

La batería que se utiliza para el sistema de monitoreo es una batería de LiPo de 3 celdas en serie a 11.1 V que se puede ver en la figura 4.16 por lo tanto, se tiene que utilizar un regulador de voltaje a 5 V ya que todos los sensores y el FPGA operan a este nivel de voltaje.



Figura 4.16: Batería de LiPo de sistema de monitoreo.

Las características de la batería de LiPo se muestran en la tabla.

Características	Batería LiPo
Capacidad nominal(C)	1300mAh
Tasa de descarga	20C
Dimensiones	75 × 35 × 24 mm

Tabla 4.2: Especificaciones técnicas de la batería de LiPo [26] para el sistema de monitoreo

4.8. Desarrollo de i nterfaz gráfica

La interfaz gráfica esta desarrollada en LabVIEW versión 2013 que se muestra en la figura 4.17, cuya función es visualizar la información de los sensores del monitoreo ambiental (Humedad, temperatura, indice de rayos Uv, presión atmosférica y calidad del aire) y enviar un comando que inicie la transmisión de datos.

La interfaz también se encarga de clasificar la información recibida y tiene un tiempo de muestreo de 1Hz, por lo que esta envía el comando de requerimiento de señal cada segundo.

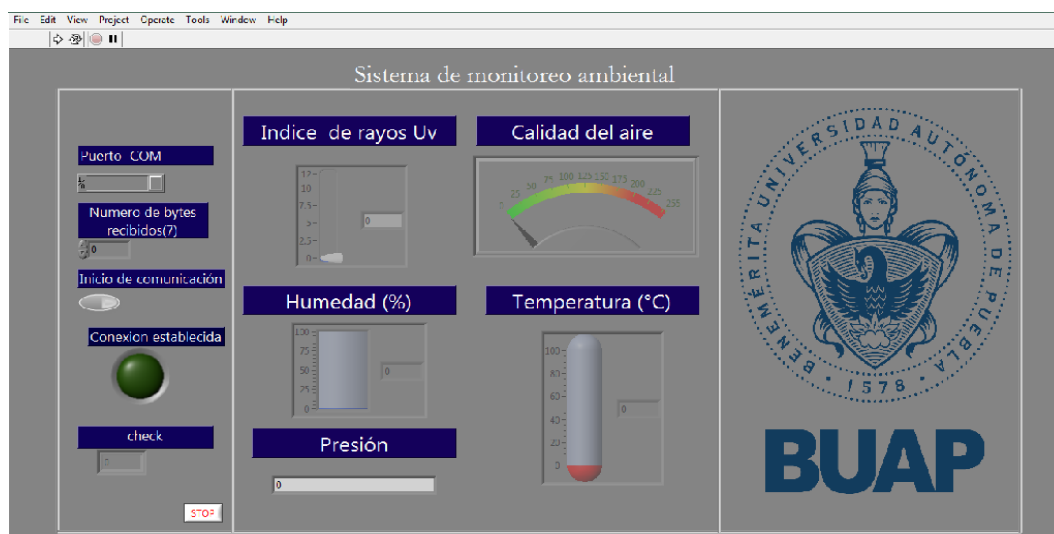


Figura 4.17: Interfaz gráfica del sistema de monitoreo

4.9. Resultados experimentales

En el presente capítulo se muestran los resultados obtenidos experimentalmente del sistema de monitoreo ambiental. Las pruebas realizadas fueron realizadas sobre un vehículo aéreo no tripulado Hexarotor que se ve en la figura 4.18 para conocer el rango de transmisión de datos y comprobar el buen funcionamiento del sistema.

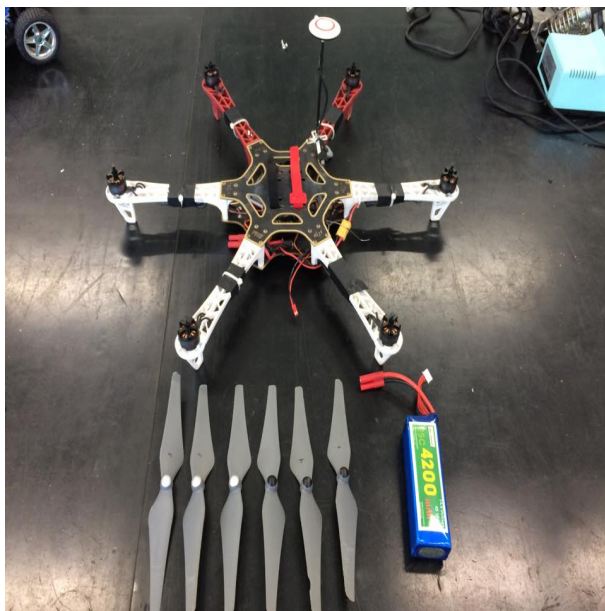


Figura 4.18: Hexarotor

El sistema de monitoreo fue montado sobre el Hexarotor y se hicieron varias pruebas de vuelo y monitoreo simultáneamente, en la figura 4.19 se puede ver el montaje del sistema de monitoreo sobre el Hexarotor y una computadora ejecutando el software de la interfaz gráfica. Una vez realizado un montaje seguro se iniciaba el vuelo y se monitoreaba desde la computadora si todos los datos era recibidos correctamente, en la figura 4.19 podemos observar al Hexarotor volando con el dispositivo de monitoreo ambiental.



Figura 4.19: Prueba de funcionamiento

También se hizo una comparación de los datos obtenidos con una pagina de Internet [29] de

CAPÍTULO 4. IMPLEMENTACIÓN DE SISTEMA DE MONITOREO

4.9. RESULTADOS EXPERIMENTALES

monitoreo ambiental en Puebla.

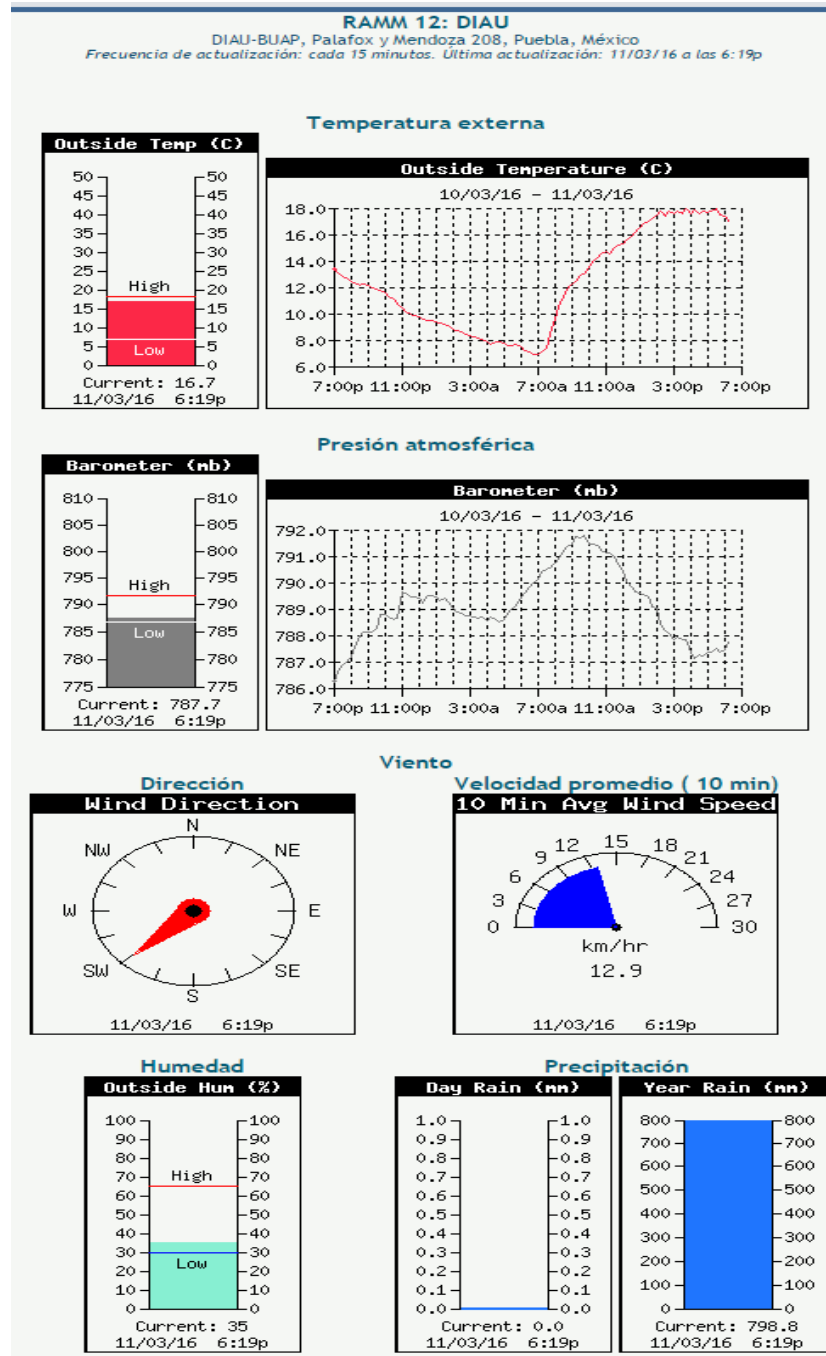


Figura 4.20: Datos recolectados de Internet

Como se puede observar en la figura 4.21 y figura 4.20 los datos que han sido recopilados por la estación de monitoreo son muy parecidos a los sondeados por la pagina de Internet consultada, pero esta pequeña desviación de los datos es debido a que en la pagina de Internet se monitorea desde el centro de Puebla y la estación de monitoreo estaba encontrada en Ciudad Universitaria Puebla, por lo tanto es normal la diferencia entre estos datos.

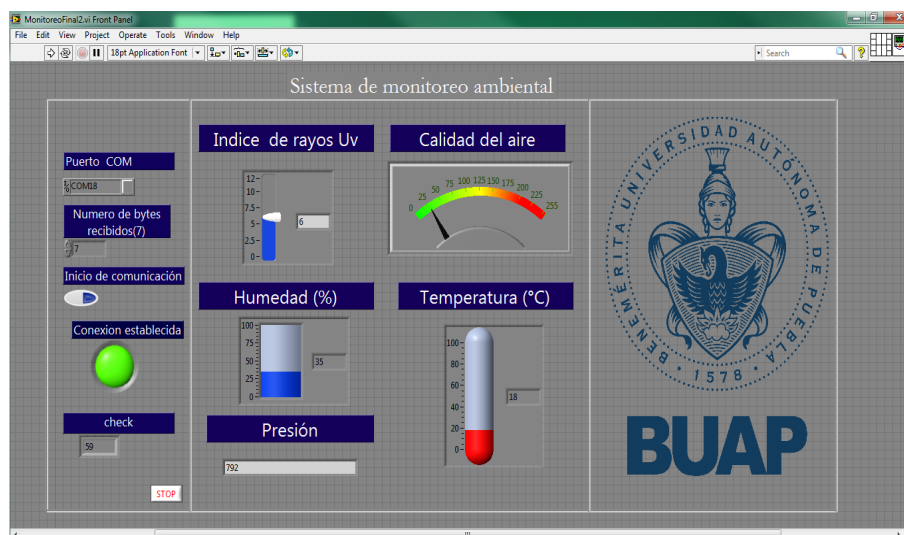


Figura 4.21: Datos recolectados por la estación meteorológica

Conclusiones

- Se implementó el algoritmo para el sistema de navegación con la integración del RK4, también se implementó el algoritmo para el sistema de monitoreo ambiental que recopila todos los datos enviados por los sensores, cada algoritmo fue realizado en VHDL y System Generator integrado en un FPGA Xula2 independiente y ambos pueden ser utilizados en vehículos autónomos.
- Se realizó la programación del FPGA en VHDL para habilitar el modulo AltIMU-10 con comunicación I^2C el cual permitió implementar el sistema de navegación y se caracterizó cada sensor mediante su hoja de datos.
- El GPS funciona correctamente, se habilitó la comunicación con el FPGA utilizando un protocolo serial, solo en el caso de que el cielo este nublado este presentará algunos problemas cuando se requiera recibir la señal del satélite, la visualización de datos del mismo se hace mediante la página web de google maps y se actualiza la información cada segundo por lo que se requiere de una conexión de Internet para que esta funcione, cabe señalar que podría descargarse una versión de los mapas para que no se requiera de conexión a Internet pero este código no es abierto por lo que se tiene que pagar una licencia para sistemas embebidos.
- Se realizo la programación del FPGA en VHDL para habilitar los módulos XBee y así tener una comunicación inalámbrica mediante protocolo Zigbee.
- Se opto por diseñar dos interfaces gráficas en LabView, una para el sistema de monitoreo y la otra para el sistema de navegación, cada una cuenta con un menú para visualizar los datos que el usuario requiera.
- Se construyó un prototipo electrónico para integrar todos los dispositivos electrónicos en un solo PCB, tanto para el sistema de monitoreo como para el sistema de navegación.
- EL contenedor de los PCB se hizo mediante impresión 3D, y se hizo el montaje sobre un VANT para probar su seguridad y funciono de manera correcta ademas de que no afecto la comunicación inalámbrica con la PC.
- Los sistemas AHRS en realidad son estimadores de la orientación (actitud y rumbo). Como no puede ser medida directamente, debe ser estimada usando diferentes tipos de sensores. Existen diferentes sensores inerciales utilizados para este tipos de sistemas pero en este caso solo se usaron acelerómetros, girómetros y magnetómetros.

- El algoritmo del sistema AHRS que integra las señales para hacer la estimación de la orientación es un observador no lineal. La orientación es parametrizada usando cuaterniones unitarios al igual que los vectores de referencia. Usar cuaterniones tiene diversas ventajas entre ellas está el que conforman una representación no singular (comparada con los ángulos de Euler), además de ser más eficiente desde el punto de vista computacional.
- El observador no lineal para la estimación de la orientación fue implementado en un FPGA XL-25 montado sobre una tarjeta de desarrollo llamada Xula 2 que trabaja a 12MHz. Al trabajar con punto flotante hace que el algoritmo tenga una buena precisión al momento de hacer las operaciones, pero debido a que el algoritmo en conjunto con el GPS y el cálculo de Runge Kutta se volvió muy complejo se presentaron errores de sincronización en el alambrado del FPGA lo que provocó que el sistema no funcionara de forma correcta.
- El algoritmo de Runge Kutta funcionó de manera correcta por lo que el presente trabajo de tesis también se puede utilizar para el cálculo de la posición de un cuerpo rígido teniendo sensores sumamente precisos.
- El sistema de monitoreo ambiental funciona de una manera correcta, aunque en lugares donde existe ruido electromagnético se disminuye el rango de transmisión de datos hasta 40 metros en lugar de 100 metros.
- Todos los dispositivos empleados tienen un bajo consumo de corriente y el voltaje de alimentación está entre los 3.3v y 5v, por lo que la batería de LiPo hace que los sistemas sean totalmente autónomos.
- Cabe señalar que el sistema de monitoreo ambiental puede ser más robusto, agregando más o mejores sensores de los que hoy existen ya que el FPGA al ser de una estructura abierta puede agregarse más módulos sin problema alguno, también este se podría integrar con el concepto de Internet de las cosas ya que estos datos pueden ser subidos a la red haciendo un trabajo de investigación en esa misma línea.

Bibliografía

- [1] **José María Martín-Moreno.** <http://www.revistaambienta.es/WebAmbienta/marm/Dinamicas/secciones/articulos/Consumoenergia.htm>, Septiembre 2015.
- [2] **Pce Instruments.** <https://www.pce-instruments.com/espanol/instrumento-medida/medidor/estacion-meteorologica-kat710621.htm>, Septiembre de 2015.
- [3] **J.F Guerrero Castellanos, G.A. Munoz-Hernandez, C. Gracios-Marin, B. Salmeron-Quiroz.** *Design and implementation of low-cost Attitude Heading References Systems (AHRS) for Micro Aerial Vehicles (MAVs).* In "Multisensor Attitude Estimation"(Eds. H. Fourati and D. C. Belkhia), CRC Press, 2015
- [4] **Vectornav.** <http://www.vectornav.com/ahrs>, Octubre, 2015.
- [5] **R.E. Kalman.** , *A New Approach to Linear Filtering and Prediction Problemes*, Trans. ASME,J. Basic Eng., Ser. D, Vol. 83, Marzo 1961, pp. 95-108
- [6] **Javier Urrutia.** *Cartografía, orientación y GPS*, Ed. ETOR-OSTOA S.L,2005.
- [7] **W. Malcolm.** *History of meteorological office*, Ed. Cambridge University Press, Noviembre, 2011
- [8] **Un poco de historia INS.** <http://www.nmine.com/uav/avionica.htm>, Octubre, 2015.
- [9] **S. Salcudean.** *A globally convergent velocity obsever for rigid body motion.*,IEEE Transactions on Automatic Control, 36(12):1493-1497, 1991.
- [10] **B. Vik and T.I. Fossen..** *An nonlinear observer for gps and ins integration*,In 40th IEEE Conference on Decision and Control, 2001.
- [11] **J. Thienel and R.M. Sanner..** *A coupled nonlinear spacecraft attitude controller and observer with an unknown constant gyro bias and gyro noise.*,IEEE Transactions on Automatic Control, 48(11):2011-2014, 2003.
- [12] **R. Mahony, T. Hamel, and J.M. Pflimlin..** *Complementary filter design on the special orthogonal group $so(3)$.*,In 44th IEEE Conference on Decision and Control and the European Control Conference, CDC05 and ECC05, Barcelona, Spain, 2005.
- [13] **R. Mahony and T. Hamel..** *Advances in Unmanned Aerial Vehicles, volume 33 of Intelligent Systems, Control, and Automation: Science and Engineering.*,Springer Netherlands, 2007.

- [14] **J. Vasconcelos, S. Silvestre, and P. Oliveira.** *A nonlinear observer for rigid body attitude estimation using vector observations.*, In Proceedings of IFAC World Congress, pages 8599-8604, Seoul, Korea, 2008.
- [15] **MOOG CROSSBOW.** <http://www.xbow.com/index.html>, noviembre, 2015
- [16] **Xsens.** <http://www.xbow.com/index.html>, noviembre, 2015
- [17] **Mti.** <http://www.xsens.com/en/general/mti>, noviembre, 2015
- [18] **3DM-GX3-25.** <http://www.microstrain.com/3dm-gx3-25.aspx>, noviembre, 2015
- [19] **P. Aggarwal, Z. Syed, A. Noureldin, and N. El-Sheimy.** *MEMS-Based Integrated Navigation.*, Firts Edition, Artech House, USA, 2010.
- [20] **M. J. Caruso, T. Bratland, C. H. Smith, and R. Schneider.** *A New Perspective on Magnetic Field Sensing. Sensors Magazine*, Diciembre 1998, pp. 1-19.
- [21] **M. Díaz Michelena.** *Estudio de dispositivos magnetométricos aplicación al proyecto NANOSAT .*, Tesis doctoral, UTM-ETSIT, Madrid, España, 2004.
- [22] **Roving Networks, Class 1 Bluetooth Module DS-RN41-V3.1** 8/4, 2009
- [23] **Xilinx** <http://www.xilinx.com/products/silicon-devices/fpga/spartan-6.html> productTable, Enero 2016
- [24] **xess** <http://www.xess.com/shop/product/xula-200/>, Enero 2016
- [25] **Power Management in Portable Applications: Charging Lithium-Ion/Lithium-Polymer Batteries** *Microchip Application Note AN947, Microchip Technology Inc*, 2004
- [26] **sparkfun** <https://www.sparkfun.com/products/8483>, Enero 2016
- [27] **Texas Instruments** <http://www.ti.com/lit/ds/symlink/lm2596.pdf>, Enero 2016
- [28] **Ramm** <http://urban.diau.buap.mx/estaciones/ramm/ramm.html>, Octubre 2016
- [29] **Digi** <http://www.digi.com/lp/xbee>, Octubre 2016
<http://www.digi.com/lp/xbee>
- [30] **Thor I. Fossen .** *Guidance and control of ocean vehicles* , University of Trondheim Norway, Appendix B, John Wiley and Sons , 1994
- [31] **Natterer, F .** *The Mathematics of Computerized Tomography* , Publisher:Wiley, Stuttgart , 1986
- [32] **sparkfun** <https://www.pololu.com/product/1269>, Enero 2016

Apéndice A

Cuaterniones

Los cuaterniones fueron descubiertos por William Rowan Hamilton en 1843. Un cuaternión está compuesto por cuatro parámetros $q = (q_0, q_1, q_2, q_3) \in \mathbb{R}^4$. El cuaternión q se puede expresar como un número complejo con la componente real a , y tres componentes imaginarios, q_1, q_2 y q_3 , como sigue:

$$q = q_0 + \mathbf{i}q_1 + \mathbf{j}q_2 + \mathbf{k}q_3 \quad (\text{A.1})$$

Esto es una extensión de la forma habitual de los números complejos con una componente real y una componente imaginaria, $x = a + \mathbf{i}b$, con la que el lector es más probable esté familiarizado.

En ocasiones q_0 se denomina la parte escalar de q y se define $\vec{q} = [q_1 \ q_2 \ q_3]^T$ como la parte vectorial de q .

A.1. Álgebra de cuaterniones

Para poder entender los cuaterniones es importante conocer su álgebra, es decir, como se opera con cuaterniones. Sean dos cuaterniones dados por:

$$a = a_0 + \mathbf{i}a_1 + \mathbf{j}a_2 + \mathbf{k}a_3 \quad (\text{A.2})$$

$$b = b_0 + \mathbf{i}b_1 + \mathbf{j}b_2 + \mathbf{k}b_3 \quad (\text{A.3})$$

A.1.1. Suma de cuaterniones

La suma de cuaterniones se realiza análogamente a como se hace con los números complejos, es decir, término a término

$$a + b = (a_0 + b_0) + \mathbf{i}(a_1 + b_1) + \mathbf{j}(a_2 + b_2) + \mathbf{k}(a_3 + b_3) \quad (\text{A.4})$$

A.1.2. Resta de cuaterniones

La resta de cuaterniones se realiza de forma análoga a la suma

$$a - b = (a_0 - b_0) + \mathbf{i}(a_1 - b_1) + \mathbf{j}(a_2 - b_2) + \mathbf{k}(a_3 - b_3) \quad (\text{A.5})$$

A.1.3. Multiplicación de cuaterniones

Para multiplicar cuaterniones hemos de tener en cuenta el álgebra que hay detrás de las 3 unidades imaginarias, usando la fórmula de Hamilton.

$$\mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = \mathbf{ijk} = -1 \quad (\text{A.6})$$

De donde se pueden obtener el resto de identidades para poder operar la multiplicación entre cuaterniones, que son:

$$\mathbf{i} * \mathbf{i} = \mathbf{j} * \mathbf{j} = \mathbf{k} * \mathbf{k} = -1 \quad (\text{A.7})$$

$$\mathbf{i} * \mathbf{j} = \mathbf{k} \quad (\text{A.8})$$

$$\mathbf{j} * \mathbf{i} = -\mathbf{k} \quad (\text{A.9})$$

$$\mathbf{j} * \mathbf{k} = \mathbf{i} \quad (\text{A.10})$$

$$\mathbf{k} * \mathbf{j} = -\mathbf{i} \quad (\text{A.11})$$

$$\mathbf{k} * \mathbf{i} = \mathbf{j} \quad (\text{A.12})$$

$$\mathbf{i} * \mathbf{k} = -\mathbf{j} \quad (\text{A.13})$$

La multiplicación se realiza componente a componente, y está dado en su forma completa por

$$\begin{aligned} a \otimes b &= (a_0 + \mathbf{i}a_1 + \mathbf{j}a_2 + \mathbf{k}a_3)(b_0 + \mathbf{i}b_1 + \mathbf{j}b_2 + \mathbf{k}b_3) \\ &= (a_0b_0 - a_1b_1 - a_2b_2 - a_3b_3) + \mathbf{i}(a_0b_1 + a_1b_0 + a_2b_3 - a_3b_2) \\ &\quad + \mathbf{j}(a_0b_2 - a_1b_3 + a_2b_0 + a_3b_1) + \mathbf{k}(a_0b_3 + a_1b_2 - a_2b_1 + a_3b_0) \end{aligned} \quad (\text{A.14})$$

donde $\otimes$ representa la multiplicación entre cuaterniones. Una forma ligeramente más reducida puede ser:

$$a \otimes b = a_0b_0 - \vec{a} \cdot \vec{b} + a_0\vec{b} + b_0\vec{a} + \vec{a} \times \vec{b} \quad (\text{A.15})$$

Alternativamente, el producto de cuaterniones puede expresarse en forma de matriz como:

$$a \otimes b = \begin{bmatrix} a_0 & -\vec{a} \\ \vec{a} & a_0 I_3 + [\vec{a}^\times] \end{bmatrix} \begin{bmatrix} b_0 \\ \vec{b} \end{bmatrix} \quad (\text{A.16})$$

donde I_3 es la matriz identidad y $[\vec{\xi}^\times]$ es el tensor antisimétrico asociado con el vector axial $\vec{\xi}$ y está dado por

$$[\vec{\xi}^\times] = \begin{bmatrix} \xi_1 \\ \xi_2 \\ \xi_3 \end{bmatrix}^\times = \begin{bmatrix} 0 & -\xi_3 & \xi_2 \\ \xi_3 & 0 & -\xi_1 \\ -\xi_2 & \xi_1 & 0 \end{bmatrix} \quad (\text{A.17})$$

entonces, la multiplicación de cuaterniones queda de la siguiente forma:

$$\begin{aligned} a \otimes b &= \begin{bmatrix} a_0 & -a_1 & -a_2 & -a_3 \\ a_1 & a_0 & -a_3 & a_2 \\ a_2 & a_3 & a_0 & -a_1 \\ a_3 & -a_2 & a_1 & a_0 \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{bmatrix} \\ &= \begin{bmatrix} b_0 & -b_1 & -b_2 & -b_3 \\ b_1 & b_0 & b_3 & -b_2 \\ b_2 & -b_3 & b_0 & b_1 \\ b_3 & b_2 & -b_1 & b_0 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} \end{aligned} \quad (\text{A.18})$$

Es importante señalar que la multiplicación de cuaterniones no es conmutativa: $a \otimes b \neq b \otimes a$, pero es asociativa $a \otimes (b \otimes c) = (a \otimes b) \otimes c$.

A.1.4. Conjugación de cuaterniones

El conjugado de un cuaternión $q = q_0 + \mathbf{i}q_1 + \mathbf{j}q_2 + \mathbf{k}q_3$ está dado por $\bar{q} = q_0 - \mathbf{i}q_1 - \mathbf{j}q_2 - \mathbf{k}q_3$.

El valor absoluto de un cuaternión q está dado por

$$\|q\| = \sqrt{q \otimes \bar{q}} = \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2} \quad (\text{A.19})$$

A.1.5. División de cuaterniones

Usando la forma del inverso, es posible escribir el cociente de dos cuaterniones como

$$\frac{a}{b} = \frac{a \otimes \bar{b}}{\|b\|^2} \quad (\text{A.20})$$

El inverso multiplicativo de un cuaternión q , distinto de cero, está dado por

$$q^{-1} = \frac{\bar{q}}{q \otimes \bar{q}} = \frac{\bar{q}}{\|q\|^2} \quad (\text{A.21})$$

A.1.6. Exponenciación de cuaterniones

La exponenciación de cuaterniones, al igual que sucede con los complejos, está relacionada con funciones trigonométricas. Dado un cuaternión escrito en forma canónica $q = q_0 + \mathbf{i}q_1 + \mathbf{j}q_2 + \mathbf{k}q_3$ se exponenciación resulta ser

$$e^q = e^{q_0 + \mathbf{i}q_1 + \mathbf{j}q_2 + \mathbf{k}q_3} = e^{q_0} \left(\cos \sqrt{q_1^2 + q_2^2 + q_3^2} + \frac{\sin \sqrt{q_1^2 + q_2^2 + q_3^2}}{\sqrt{q_1^2 + q_2^2 + q_3^2}} (\mathbf{i}q_1 + \mathbf{j}q_2 + \mathbf{k}q_3) \right) \quad (\text{A.22})$$

A.1.7. Cuaterniones a ángulos de Euler

La matriz de rotación $R(q)$ que transforma el sistema de coordenadas cuerpo al sistema inercial, correspondiente al cuaternión q se calcula por

$$\begin{aligned} R(q) &= (q_0^2 - \vec{q}^T \vec{q}) I_3 + 2(\vec{q} \vec{q}^T - q_0 [\vec{q}^\times]) \\ &= \begin{bmatrix} 2(q_0^2 + 2q_1^2) - 1 & 2(q_1q_2 + q_0q_3) & 2(q_1q_3 - q_0q_2) \\ 2(q_2q_1 - q_0q_3) & 2(q_0^2 + 2q_2^2) - 1 & 2(q_2q_3 + q_0q_1) \\ 2(q_3q_1 + q_0q_2) & 2(q_3q_2 - q_0q_1) & 2(q_0^2 + 2q_3^2) - 1 \end{bmatrix} \end{aligned} \quad (\text{A.23})$$

Para conocer la orientación en forma de ángulos de Euler: roll (ϕ), pitch (θ) y yaw (ψ), se necesita conocer la relación que existe entre los cuaterniones y los ángulos de Euler. Para ello, se impone que las matrices de rotación en las dos representaciones sean iguales. Sea $R_E(\phi, \theta, \psi)$ la matriz de rotación en ángulos de Euler, y $R(q)$ la matriz conocida de rotación de cuaterniones, donde R_{ij} es el elemento de la fila i y columna j de esta matriz. Se tiene así que

$$R_E(\phi, \theta, \psi) \equiv R(q) \quad (\text{A.24})$$

$$\begin{bmatrix} c_\theta c_\psi & c_\psi s_\theta s_\phi - s_\psi c_\theta & c_\psi s_\theta c_\phi + s_\psi s_\phi \\ c_\theta s_\psi & s_\psi s_\theta s_\phi + c_\psi c_\theta & s_\psi s_\theta c_\phi - c_\psi s_\phi \\ -s_\theta & c_\theta s_\phi & c_\theta c_\phi \end{bmatrix} = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix} \quad (\text{A.25})$$

donde $c_x = \cos(x)$ y $s_x = \sin(x)$. Una posible solución para encontrar los ángulos de Euler a través de la matriz $R(q)$ es

$$\begin{aligned}\theta &= -\operatorname{asen}(R_{31}) \\ \phi &= \operatorname{asen}\left(\frac{R_{32}}{\cos(\theta)}\right) \\ \psi &= \operatorname{asen}\left(\frac{R_{21}}{\cos(\theta)}\right)\end{aligned}\tag{A.26}$$

Apéndice B

Protocolos de comunicación

Los protocolos de comunicación que se utilizan para los distintos sensores en el presente trabajo de tesis son $RS-232$ (serial) y I^2C . Cada uno de estos protocolos se implemento como un modulo en FPGA utilizando librerías en VHDL. A continuación se muestra el principio básico de estos protocolos.

B.0.8. Protocolo RS-232

La comunicación serial consiste en el envío de un bit de información de manera secuencial, ésto es, un bit a la vez y a una frecuencia conocida. La comunicación serial sigue los estandares definidos en 1969 por el RS-232 (Recommended Standar 232) que establece los niveles de voltaje y velocidad de transmisión de datos. La comunicación serial utiliza tres lineas de trasmisión: (1)Tierra, (2)Transmisor, (3) Receptor.

La transmisión de datos puede realizarse de manera síncrona o asíncrona, es decir si se depende o no de una señal de reloj para sincronizar transmisión, en el caso de este trabajo de tesis solo se utiliza en modo asíncrono.

La transmisión de datos en modo asíncrono se hace de la siguiente manera: en primer lugar se manda un bit de inicio (0) que sirve al receptor para la sincronización, posteriormente se envía el mensaje (un byte) y un bit de alto para indicar que la transmisión finalizo tal como se ve en la figura B.1, para enviar un nuevo byte se repite el mismo proceso, si se requiere se puede enviar un bit de paridad para comprobar si el dato fue recibido correctamente, el valor de este bit depende del dato enviado y del tipo de paridad utilizada.

- Paridad par: su valor es cero si en numero de ceros enviados es par
- Paridad impar: su valor es cero si el numero de ceros enviado es impar.

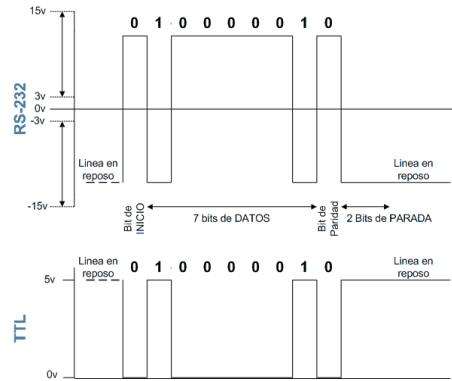


Figura B.1: Protocolo de comunicación serial asíncrono

El voltaje nominal para la comunicación serial por el protocolo RS-232 es de 12 V pero con el uso del convertidor MAX232 se puede adaptar este voltaje a TTL y así poder utilizar este tipo de comunicación en el FPGA.

La comunicación serial además de ser asíncrona para el presente trabajo de tesis tiene siguiente configuración.

- **Velocidad de transmisión (*baud rate*):** Indica el número de bits transmitido por segundo, y se mide en baudios (*bauds*). En este caso la velocidad sera de 9600 baudios es decir 9600 datos por segundo.
- **Bits de datos:** se refiere a la cantidad de bits para la transmisión el cual sera de ocho.
- **Bits de parada:** Se utilizar para indicar el fin de la transmisión en este caso solo sera de un bit de parada.
- **Paridad:** En este caso no utilizaremos bit de paridad.

B.0.9. Protocolo I^2C

El protocolo I^2C es un tipo de comunicación es serie sincrona: su nombre significa *Inter-Integrated Circuit* (Circuitos inter-integrados). Este estándar facilita la comunicación con varios dispositivos en un solo bus de datos auna velocidad estándar de 100 Kbits por segundo, que es la velocidad a la cual todos los dispositivos asociados a esta tesis están configurados, aunque se puede configurar a una velocidad de hasta 400 kbits por segundo.

El Protocolo I^2C utiliza tres señales para lograr establecer el intercambio de datos, a continuación se describen estas tres señales.

- **SCL (*System Clock*)** es el bus por donde se transmiten las señales de reloj que sincroniza los distintos dispositivos conectados al bus de datos.
- **SDA (*System Data*)** es el bus por donde se trasmiten los datos entre los dispositivos.
- **GND (*Ground*)** común entre todos los dispositivos conectados al mismo bus de datos

Las líneas SDA y SCL son bidireccionales y son las señales que transportan la información entre los dispositivos conectados al mismo bus. Cada dispositivo conectado al bus es reconocido por su dirección y puede operar como transmisor o receptor de datos así también pueden funcionar como maestro o esclavo.

El dispositivo maestro es el que inicia la transferencia en el bus de datos y genera la señal de reloj, los dispositivos esclavos solo son dispositivos a los que se accede el maestro mediante su dirección. Las líneas de datos son bidireccionales y deben estar conectadas a voltaje positivo de alimentación a través de resistencia pull-up tal como se muestra en la figura B.2.

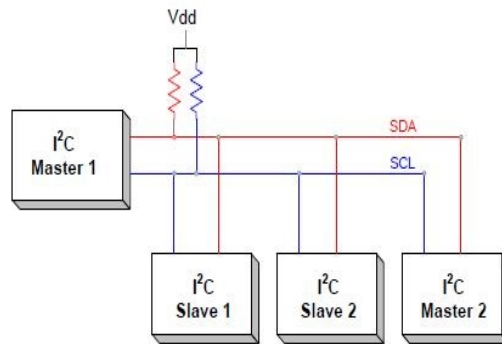


Figura B.2: Conexión I^2C típica

Antes de que se establezca el intercambio de datos entre el maestro y los esclavos, el maestro debe iniciar la comunicación (condición de inicio), esta condición se cumple cuando la línea SDA cae a cero y SCL permanece en alto. A partir de este momento comienza la transferencia de datos, una vez finalizada se debe cumplir la condición de stop, la línea SDA pasa a nivel alto mientras SCL permanece en estado alto. La figura nos muestra como se establece el intercambio de datos mediante este protocolo.

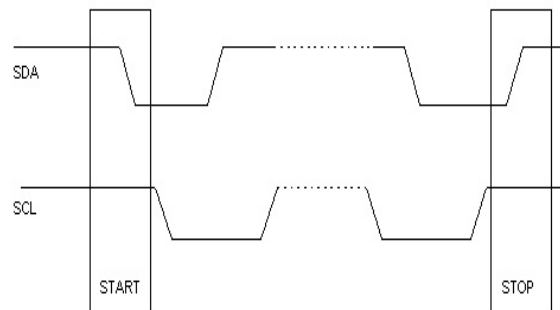


Figura B.3: Transmisión de datos I^2C