



**UNIVERSIDAD NACIONAL AUTÓNOMA
DE MÉXICO**

**FACULTAD DE INGENIERÍA
DIVISIÓN DE INGENIERÍA ELÉCTRICA**

**IMPLEMENTACIÓN DE UN PILOTO AUTOMÁTICO PARA UN UAV
(VEHÍCULO AÉREO NO TRIPULADO), MONITOREADO Y
CONTROLADO POR MÓDULOS DE TECNOLOGÍA ZIGBEE MEDIANTE
UNA ESTACIÓN TERRENA**

QUE PARA OBTENER EL TÍTULO DE

INGENIERO EN ELÉCTRICA Y ELECTRÓNICA

PRESENTA

RIGOBERTO RANGEL VARGAS

DIRECTOR DE TESIS

M.I. JESÚS ALVAREZ CASTILLO



MÉXICO D.F., CIUDAD UNIVERSITARIA JULIO DE 2015

JURADO ASIGNADO

Presidente: **DR. JESUS SAVAJE CARMONA**

Vocal: **M.I. JESUS ALVAREZ CASTILLO**

Secretario: **DR. JUAN MARIO PEÑA CABRERA**

1er Suplente: **DR. NICOLAS CEFERINO KEMPER VALVERDE**

2° Suplente: **DRA. GRACIELA VELASCO HERRERA**

Lugar donde se realizó la tesis:

Ciudad Universitaria, México D.F.

M.I. Jesús Álvarez Castillo

Tutor de tesis

Dra. Graciela Velasco Herrera

Cotutor de tesis

El presente trabajo fue realizado en el laboratorio de Transductores y Actuadores del Departamento de Procesamiento de Señales en el edificio S de Posgrado de Ingeniería de la Universidad Nacional Autónoma de México, bajo la tutoría del M. en I. Jesús Álvarez Castillo y bajo la asesoría del Dr. Mario Alberto Mendoza Bárcenas y la Dra. Graciela Velasco Herrera.

AGRADECIMIENTOS

A la Universidad Nacional Autónoma de México por haberme brindado la oportunidad de una formación profesional en sus aulas.

A la Dra. Graciela Velasco Herrera y al Dr. Mario Alberto Mendoza Bárcenas por su apoyo, paciencia y contribuciones para el desarrollo de este proyecto de tesis.

Al M.I. Jesús Álvarez Castillo por su apoyo y dedicación durante la dirección de este proyecto.

DEDICATORIAS

A mis padres que sin su apoyo y esfuerzos incondicionales no hubiese podido cursar una licenciatura, así como el amor, la guía, la dedicación y paciencia que me han brindado en toda la vida.

A mis hermanos que siempre han formado parte fundamental en el núcleo familiar y en mi vida.

A los amigos que me brindaron su apoyo en el transcurso de la carrera así como de la elaboración de este proyecto.

Índice general

Resumen	8
Capítulo 1. Introducción.....	9
1.1 Justificación	10
1.2 Objetivo principal	11
1.2.1 Objetivos secundarios	11
1.3 Antecedentes.....	11
1.4 Metodología	14
1.5 Narrativa por capítulos	16
Capítulo 2. Conceptos Básicos.....	18
2.1 Sistemas Aéreos no tripulados	18
2.2 Tareas civiles	20
2.3 Aspectos puntuales de los subsistemas.	22
2.4 Características de maniobra UAV:.....	23
2.5 Estación de control de tierra.....	23
2.6 Monitoreo, control y planeamiento de misión.....	24
2.7 Seguridad	25
2.8 Carga útil	25
2.9 Link de datos	26
2.10 Características particulares de operación	26
Capítulo 3. Navegación y vuelo en un UAV.	29
3.1 Ejes del UAV, movimientos de Pitch, Roll, Yaw.....	29
3.2 Sistema de Navegación Inercial (INS)	31
3.3 Instrumentos básicos de vuelo	34
3.3.1 Horizonte artificial.....	34
3.3.2 Indicador de altura (Altímetro) y presión barométrica	35
3.3.3 Indicador de velocidad	36
3.4 Instrumentos de Navegación (Rumbo).....	36
3.4.1 Brújula	36
3.4.2 GPS	37

CAPÍTULO 4. Comunicaciones.....	39
4.1 Estructura de un sistema de comunicación	39
4.2 Comunicaciones inalámbricas	40
4.2.1 Comunicación infrarroja.....	41
4.2.2 Comunicación por Radiofrecuencia.....	41
4.3 Comunicaciones Alámbricas	54
4.3.1 Protocolo UART (Universal Asynchronous Receiver-Transmitter)	55
4.3.2 Interfaz I2C (Inter Integrated Communication)	57
Capítulo 5 Desarrollo del sistema de piloto automático abordo de un UAV	60
5.1 Integración de elementos electrónicos	60
5.2 Sensores	62
5.2.1 Indicador de horizonte artificial.....	64
5.2.2 Sensor de altitud y de temperatura	67
5.2.3 GPS	68
5.3 Tarjeta de telemetría para el control y el monitoreo del UAV	71
5.3.1 Control automático de navegación.....	72
CAPITULO 6. Implementación de la estación terrena GCS.....	75
6.1 Configuración de la red punto a punto con dispositivos ZigBee®	75
6.2 Especificaciones del software de la estación terrena GCS	79
6.3 Desarrollo de software y despliegue de datos en Labview	80
CAPITULO 7. Realización de pruebas y adquisición de datos.....	84
CAPITULO 8. Conclusiones.....	88
BIBLIOGRAFIA.....	91
REFERENCIAS.....	97
GLOSARIO.....	99
Anexos.....	100
Anexo A Matriz DCM	100
Anexo B Rutina de programación de la telemetría	104
Anexo C Índices	119
Índice de Imágenes	119

Índice de Tablas	121
Anexo D Certificaciones asociadas al proyecto de tesis.....	122
Anexo E Interfaz Radio Control-Tiva-KK2.0	124
Anexo F Tarjeta de estabilización KK2.0.....	125

Resumen

El uso de UAV (*Unmanned Aerial Vehicle*), para tareas civiles se encuentra en expansión logrando dominar varios campos de aplicación para facilitar tareas que en algunos casos resultaba muy difícil, peligroso y en ocasiones imposible. Estos mayormente facilitan tareas como el monitoreo agrícola, vigilancia, evaluación del estado físico de infraestructuras tales como puentes, presas y oleoductos así como el la vigilancia en áreas de desastre como los terremotos e inclusive en áreas con riesgos tóxicos o mortales para el humano como el derrame de sustancias peligrosas. Hasta hace algunos años pocas de estas tareas se restringían a cámaras fijas incrementando la cantidad de dispositivos así como en algunos casos el extenso cableado. Las necesidades de obtener información por medios móviles como lo es la fotografía o el video en tiempo real han acaparado presencia en sistemas de esta índole y en algunos casos llegan a ser indispensables.

En este trabajo de tesis se presenta la implementación básica de un piloto automático partiendo de distintos sensores y dispositivos necesarios para cumplir la comunicación a corta distancia y el monitoreo tanto de la ubicación geográfica como del comportamiento de la aeronave para su interpretación en una PC.

Finalmente dicha integración de dispositivos se montan en un UAV de despegue vertical VTOL (Vertical Take Off Landing) de cuatro hélices marca Turnigy SK450 del cual se utiliza una tarjeta de estabilización de vuelo, esto con la finalidad de realizar las pruebas de operación correspondiente al vuelo con el piloto automático elaborado.

Capítulo 1. Introducción

En este momento el uso de pequeñas aeronaves (menores a 2 Kg), para uso civil ha crecido considerablemente, el mercado actual se encuentra con poca participación de empresas lo cual hace atractiva la inversión y el desarrollo de estos sistemas.

Asimismo existen plataformas basadas en microcontroladores en las cuales el código es de libre acceso como el ARDUINO, la característica que los define es que contienen algoritmos únicamente para estabilizar la aeronave y seguir indicaciones de un piloto en tierra. Para lograr fines de aplicación, estas plataformas no son la mejor opción pues son de capacidades muy pequeñas o con aplicaciones muy particulares y en todo caso sirven como modelos de aprendizaje pues no existe un seguimiento y corrección de errores a lo largo del tiempo, como por ejemplo la resolución de un convertidor analógico-digital dentro de dichos sistemas es crucial para minimizar el error en las lecturas de los sensores y típicamente no utilizan el de mayor resolución. Así también la velocidad de procesamiento tiene que acercarse a ser una respuesta prácticamente en tiempo real para lograr las acciones correctivas necesarias lo antes posible y así evitar la colisión o inclusive la caída del UAV.

En la mayoría de los casos la operación de monitoreo y control de un UAV se llevan a cabo por medio de comunicaciones vía radio, esto con distintas tecnologías las cuales se diferencian esencialmente por la cantidad de datos transportados en un determinado tiempo y el radio de acción del dispositivo.

Para lograr realizar mediciones físicas se hace uso de herramientas como lo son acelerómetros, giroscopios y magnetómetros. Estas operan a través de circuitos MEMS (*Micro Electro-Mechanical Systems*), los cuales transforman los movimientos en señales eléctricas proporcionales al movimiento y por medio de un procesamiento en un microcontrolador es posible reproducir la representación aproximada de la actitud en vuelo del UAV.

En este trabajo se presenta la integración de tecnologías recientes en cuanto a obtención y procesamiento de datos por medio de sensores integrados en una tarjeta y en otra un microcontrolador con núcleo Cortex-ARM de 32 bits, permitiendo el despliegue de datos en tiempos relativamente cortos para su posterior procesamiento bajo algún algoritmo, permitiendo la versatilidad de ser escalable y con las ventajas de tener un tamaño pequeño y un peso reducido de acuerdo a las características de los UAS (*Unmanned Aerial System*), siendo de vital importancia como se verá más adelante. Si bien la velocidad de procesamiento del microcontrolador utilizado en el presente trabajo es buena,

existen dispositivos con mayores capacidades logrando aproximar la respuesta a un tiempo real tales como los DSP (*Digital Signal Processor*). Asimismo se implementa un control Proporcional- Derivativo, el cual se realiza con la finalidad de incursionar en el vuelo autónomo, lo anterior no contempla el bloque que estabiliza el vuelo del UAV, para ello se recurre a el uso de la tarjeta KK2.0 (ver anexo F).

1.1 Justificación

Este proyecto surge de la necesidad de realizar vuelos prolongados de un UAV en casos donde la distancia juegue un papel importante y no se pueda tener el modelo a simple vista, algunos casos pueden ser: inundaciones, revisión de fallas en tuberías petroleras, revisión de puentes colgantes, topografía, vigilancia, etc... O inclusive como herramienta secundaria en casos en los que el piloto en tierra dedique bastantes horas a la manipulación, haciendo la labor más ligera.

Para lo cual se parte de la lectura de una serie de instrumentos que dictan el comportamiento y la posición del UAV, los que por medio de un microcontrolador es posible interpretar y procesar para obtener una respuesta de un algoritmo y se pueda proceder a la corrección necesaria hacia el rumbo deseado y así lograr el vuelo autónomo. El objeto de este trabajo no contempla las etapas de estabilización de la aeronave e interfaces con los motores.

Paralelamente se envían los datos hacia una PC por medio de módulos transceivers (transmisor y receptor en un solo módulo) de tecnología ZigBee® para desplegar gráficamente el monitoreo a distancia.

Lo que impulsa al desarrollo de esta plataforma es el uso de tecnologías recientes con la finalidad de formar un sistema encaminado a la complementación y correcciones futuras sirviendo como punto de partida y a la vez manteniendo la adaptabilidad a cada misión partiendo de una cantidad mínima de instrumentos hasta cubrir las necesidades propias de cada una de ellas, eliminando así la dependencia y limitaciones que se puedan encontrar en relación a tarjetas de que se encuentran a la venta y que son elaboradas con fines propios de algún modelo existente.

A partir de lo expuesto con anterioridad es posible expandir el sistema hacia la perfección del vuelo por medio de un piloto automático a bordo del UAV partiendo de la experimentación realizada en este trabajo y así para los casos en los que amerite disminuir el estrés que conlleva realizar misiones manualmente desde tierra durante tiempos prolongados.

1.2 Objetivo principal

Implementar un piloto automático para un UAV, monitoreado y controlado por módulos de tecnología ZigBee mediante una estación en tierra, el cual tiene como alcance mostrar la instrumentación necesaria para poder visualizar el estado de la aeronave en vuelo así como realizar la tarea de vuelo autónomo posteriormente.

1.2.1 Objetivos secundarios

1. Obtención de datos en vuelo como lo son: altura por medio de la presión barométrica, comportamiento de la aeronave en sus tres ejes por medio de giroscopios, acelerómetros y magnetómetros mediante el uso de la tarjeta BOOSTXL-SENSHUB ensamblada por Texas Instruments.
2. Implementar la red de comunicación inalámbrica entre estación terrena y el UAV para verificar el estado y ubicación del modelo en uso por medio de tecnología ZigBee.
3. Implementar una interfaz gráfica en una PC para visualizar el comportamiento y ubicación del UAV.
4. Implementación de un algoritmo de control en la tarjeta de abordo para lograr el vuelo autónomo de acuerdo a la navegación inercial.

1.3 Antecedentes

Los primeros precursores de la aviación no tripulada se remontan al siglo XIX, cuando los norteamericanos Elmer Sperry que junto a su hijo Lawrence y el inventor Peter Cooper Hewitt, establecieron las bases para el éxito de esa tecnología, enfocándose a las áreas de radio control y guiado inercial. Tras haberse producido varios intentos frustrados prácticamente desde el inicio de la aviación en 1903, podría decirse que la fecha que marcaría formalmente el primer vuelo con éxito de una aeronave no tripulada es el 6 de marzo de 1918, esto en Copiague, Long Island con el lanzamiento del torpedo aéreo Curtiss-Sperry. Aunque los intentos fracasaron por cuestiones de avances prematuros no fue

sino hasta la segunda guerra mundial que se comenzó a realizar un empleo efectivo como blancos aéreos para validar la eficacia de los cañones antiaéreos. A raíz de esto y con el paso del tiempo poco a poco se fue perfilando a pasar de ser un blanco a su aplicación principal en la década de los: reconocimiento aéreo. Bajo este criterio se considera que el primer UAS de reconocimiento fue el SD-1, que fue empleado desde 1959 hasta 1966 primero por los estadounidenses y luego por los británicos con el nombre de Observer. Un uso secundario en esa época fue el de servir como señuelos para las defensas antiaéreas soviéticas.

Con el recrudecimiento de la guerra fría y el derribo de varios aviones espía U-2 de la CIA en la década de los 60, a raíz de estos acontecimientos Estados Unidos se enfocó en la aceleración del desarrollo de tecnologías UAS, cosechando sistemas como el Ryan que posteriormente se renombró como Firebee y más tarde como Lightning Bug. El primer despliegue operativo de este sistema fue en 1964 sobre China y posteriormente sobre Corea del Norte y Vietnam del norte a partir de 1965. El uso principal de estos sistemas fue de reconocimiento, perturbación electromagnética e inteligencia electrónica.



Figura 1.1 Avión hércules cargando dos drones Firebee. [1]

Los soviéticos por su parte también iniciaron programas a finales de los 50 desarrollando sistemas como el La-17 y las series DBR y VR-3 realizando misiones de reconocimiento sobre el Reino Unido, Francia, España e Israel.



Figura 1.2 Dron Ia-17 de origen soviético. [2]

Después de miles de misiones en Vietnam, Israel se interesó en seguir desarrollando la tecnología Estadounidense introduciendo sus modelos en la guerra de Líbano en 1982. Posteriormente tomó rápidamente la delantera sobre Estados Unidos siendo este último el que pasó a comprar tecnología Israelí, concretamente sistemas Pioneer IAI en colaboración con la empresa estadounidense AAI. EEUU desplegó por primera vez este sistema en el estrecho de Hormuz, y a finales de los 80 decidió retomar los esfuerzos propios de desarrollo en este campo.

Tras la guerra de Vietnam, el siguiente empleo masivo de UAS tuvo lugar durante la primera guerra del golfo en 1991 donde se volaron multitud de UAV's por parte estadounidense, con el Pioneer para misiones de reconocimiento y el Chukar como señuelo para la detección de radares y amenazas antiaéreas, como por parte británica con el Midge y francesa con el Mart MKII.

El éxito de los UAS durante la guerra del golfo, junto con nuevos avances tecnológicos (GPS, cámaras miniaturizadas, navegación por satélite, microprocesadores, etc.), así como la aparición de nuevos tipos de conflicto permitió despertar de forma definitiva a comienzos de los 90 el interés de numerosos ejércitos de la OTAN por los UAS. Los acontecimientos a nivel mundial provocaron la multiplicación de los UAS en los campos de batalla al parecer un nuevo tipo de enemigo, cuya derrota pasaba por la obtención de inteligencia en tiempo real sobre pequeños objetivos, lo que provocó que tanto fuerzas armadas como los organismos de inteligencia vieran en los UAS un sistema idóneo para cubrir estas necesidades.

La historia de los UAS prácticamente está en sus inicios. Actualmente su evolución ha sido inevitablemente escrita siguiendo el guion de los conflictos armados en los siglos XX y XXI, pero la inminente introducción de los UAS en el mundo civil permitirá escribir nuevas páginas menos dramáticas.

Los avances tecnológicos permitirán ver cambios sustanciales en ese sector siendo una tarea pendiente la reducción de accidentes en estos sistemas.

1.4 Metodología

Esta tesis está conformada por la interconexión de varios sistemas electrónicos y un algoritmo de control bastante simple. La parte principal se compone por una tarjeta en la que se encuentran sensores inerciales (acelerómetros, giroscopios y magnetómetros en tres ejes), los cuales permiten conocer la posición y el movimiento del sistema de referencia a bordo de la aeronave respecto a un sistema en tierra y la orientación respecto al norte magnético así como un termómetro y un sensor de presión que muestra la altura barométrica y un GPS. Una vez obtenidas las lecturas anteriores se envían a un microprocesador donde se procesan por medio de un algoritmo llamado filtro complementario, el cual tienen como resultado una serie de datos llamados ángulos de Euler, lo cuales relacionan los ángulos de rotación entre dos sistemas ortogonales de tres ejes, lo anterior permite conocer por medio de instrumentos gráficos típicos de aviación el comportamiento y ubicación del UAV en vuelo. Asimismo en esta etapa y por medio de un algoritmo de un control PD se dispone a procesar los datos inerciales por medio de ecuaciones cinemáticas para lograr el vuelo autónomo.

Para la sección de ubicación global se utiliza un GPS, el cual proporciona datos de gran relevancia como la latitud, longitud, hora local, velocidad respecto al suelo, entre las más importantes para este trabajo. Dicho dispositivo cuenta con una resolución de tres metros, permitiendo una buena aproximación a la ubicación global real.

En la etapa de transmisión y recepción inalámbrica se utilizan dos antenas ZigBee, las cuales operan bajo un protocolo establecido que permite el envío de datos de una manera confiable por medio de una frecuencia libre (2.4Ghz). En el extremo de la PC se designa una de ellas como concentrador conectándose a la PC por medio de una interfaz serial y un puerto virtual para interconectarse al software. En el lado de la aeronave se designa la configuración de punto final.

Por otra parte en las etapas de adquisición de datos, procesamiento y control se usan dos tarjetas electrónicas, una que contiene un microcontrolador encargado de la coordinación entre la recepción y los datos que se procesaran o transmitirán, así como del procesamiento de los mismos para conocer el comportamiento y ubicación de la aeronave no tripulada para poder tomar decisiones de navegación por medio de un algoritmo de control. Este también es encargado de coordinar y enviar los datos a la antena que transmitirá hacia una computadora portátil.

En una segunda tarjeta que es posible acoplar con la tarjeta anteriormente descrita, contiene los sensores necesarios para obtener el comportamiento físico de variables como lo son: altura, movimiento relativo al horizonte terrestre, etc..

Lo anterior tiene como finalidad obtener datos en vuelo para lograr obtener una interpretación grafica de la orientación y el estado del UAV sin la necesidad de tenerlo a la vista

Para el desarrollo, alcance y finalización se aplicará la siguiente metodología:

- Analizar las características que definen a un UAS para poder identificar los alcances.
- Analizar los dispositivos y protocolos para la transmisión y recepción de información.
- Implementar una red de comunicación inalámbrica por medio de tecnología ZigBee.
- Desarrollo de algoritmos en un microcontrolador para la obtención de datos y ordenamiento para su transmisión.
- Implementación de un algoritmo de control en un microcontrolador para la autonomía en vuelo
- Desarrollo de software para la interpretación de datos obtenidos en una estación terrena
- Despliegue de datos en interfaces gráficas para su interpretación

1.5 Narrativa por capítulos

En la presente se desarrolla una serie de capítulos con la finalidad de implementar un piloto automático valiéndose de tecnologías actuales, las cuales se integraran en una tarjeta que obtendrá datos en vuelo y los transmitirá por medio de una red de comunicación de corto alcance hacia una estación terrena la cual tendrá como objetivo desplegar los datos adquiridos en interfaces gráficas para su interpretación.

En el capítulo 2 se desarrolla un panorama hacia las características que definen a los sistemas aéreos no tripulados, tales como lo son su uso en tareas civiles, los bloques más significativos para su operación y modalidades de comunicación y gestión.

El capítulo 3 hace referencia a la navegación inercial, los movimientos que una aeronave puede realizar en vuelo y los instrumentos básicos de vuelo.

El capítulo 4 se desarrolla en dos vertientes de las comunicaciones, la inalámbrica en la cual se exponen las tecnologías de radiofrecuencia bajo distintos estándares definiendo los componentes del protocolo ZigBee, su interconexión y características para lograr la transmisión de datos y por otro lado se muestran las comunicaciones alámbricas utilizadas en el bus de periféricos del microcontrolador a utilizar.

En el capítulo 5 se define y se implementa la estructura y las características de la instrumentación utilizada en la tarjeta de abordó, así como el procesamiento de los datos obtenidos mediante un microcontrolador para obtener una respuesta de guiado del rumbo a través de un algoritmo de control basado en navegación inercial. En una última etapa se transmiten los datos obtenidos por vía inalámbrica hacia la estación terrena.

Para el capítulo 6 se elabora la adquisición de datos vía serial por medio de software en la PC y se les da tratamiento para mostrarlos en un ambiente gráfico.

En el capítulo 7 muestra los resultados obtenidos durante las distintas etapas del proyecto, desde la adquisición de los datos, hasta su despliegue en la PC.

El capítulo 8 reúne las conclusiones de la tesis mostrando los resultados obtenidos durante la experimentación, así como las limitaciones y problemas presentados, comentando las características que deben ser de interés en un trabajo a futuro.

Capítulo 2. Conceptos Básicos

A continuación se definen las características de los Sistemas Aéreos No tripulados, las diferencias entre las distintas categorías y las áreas de aplicación de estos sistemas, así como la descripción de los elementos que requiere cubrir una estación terrena.

2.1 *Sistemas Aéreos no tripulados*

Los Sistemas Aéreos no Tripulados (*UAS*) constituyen el segmento de la industria aeronáutica actual que experimentará uno de los mayores crecimientos en la era 2000. Estos sistemas reúnen como un todo, los avances en ciencia y tecnología de punta: robótica, inteligencia artificial, navegación, guiado y control, comunicaciones, aerodinámica, propulsión, materiales, óptica, etc. [1].

Un sistema aéreo se compone de subsistemas (fig. 2.1):

- Vehículo aéreo
- Carga útil
- Estación de control
- Lanzamiento y recuperación de la aeronave,
- Navegación
- Comunicaciones
- Interfaces con otros sistemas
- Equipo de soporte
- Transporte; considerados los más relevantes del sistema

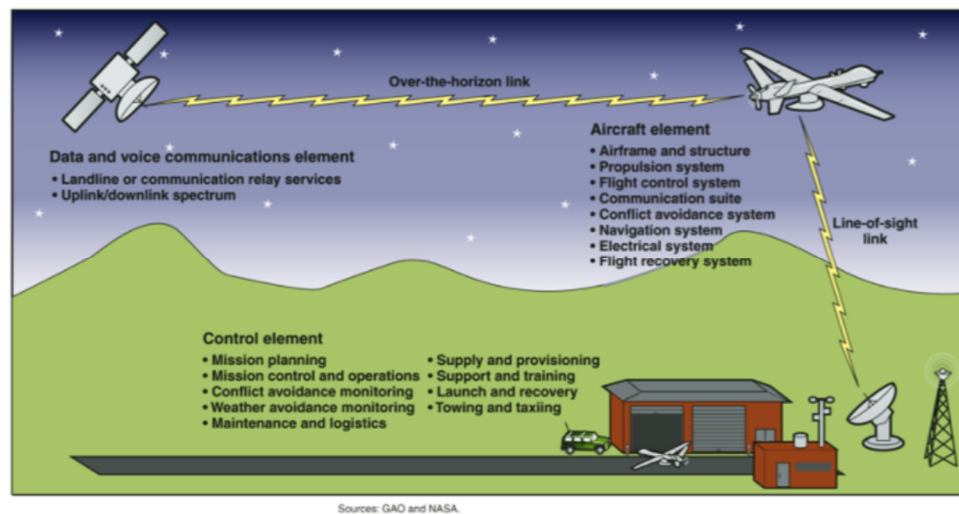


Figura 2.1 Sistema Aéreo no tripulado. [3]

El UAS puede definirse como la composición de uno o más vehículos aéreos no tripulados (UAV) y los subsistemas necesarios para su operación, como las estaciones de control, sistemas de comunicaciones, sistema de lanzamiento, trayectoria y recuperación.

Un UAS contempla todos los niveles del espacio aéreo para el vuelo de UAV's, que en general está en función de sus dimensiones, estos últimos se describen típicamente en términos de peso, resistencia, propósito de uso y la altitud de operación. Los "pequeños" UAV (menores a 25kg) que vuelan por debajo de 400m sobre el nivel del suelo y permanecen en el aire durante algunas horas, generalmente se utilizan para realizar tareas civiles y algunos pueden tener mayor resistencia y operar más allá de la capacidad de la línea de visión.

Tabla 2. 1 Clasificación por características del UAV [1]

Categoría	Peso del UAV [kg]	Altitud normal de operación [km]	Línea de visión [km]	Autonomía de vuelo
Micro	<2	0.06	5	Pocas horas
Mini	2-20	0.9	25	< 2 días
Pequeño	20-50	1.5	50	< 2 días
Táctico	150-600	3	200	< 2 días
MALE	>600	13.5	200	Días/ semanas
HALE	>600	20	Ilimitada	Días/ semanas
Combate	>600	20	Ilimitada	Días/ semanas

2.2 Tareas civiles

Actualmente el acceso a estos sistemas no tripulados es dedicado al estudio y apoyo de tareas civiles, como por ejemplo: seguimiento agrícola, adquisición de datos meteorológicos y atmosféricos, cartografía, geológica de infraestructuras, principalmente cuando se presentan situaciones de riesgo, emergencia y en condiciones climáticas adversas.

Demandan actualmente un mayor grado de eficiencia para cumplir con una misión exitosa; el desarrollo de tecnología que se aplica, en particular, carga útil, estación de control, comunicaciones, interfaces con otros sistemas, equipo de soporte, implica optimizar las etapas en el sistema, así como presentar nuevas propuestas de integración de algoritmos y métodos que se implementen en el UAS.

La industria aeroespacial se ha caracterizado por ser un sector a la vanguardia en las últimas innovaciones tecnológicas y el desarrollo de proyectos han supuesto un importante avance para la evolución de la aviación y el transporte aéreo pero también para su aplicación o contribución en otros campos de la realidad económica y social. Una de las áreas que mayor evolución viene experimentando dentro de la aeronáutica es la de los *Sistemas Aéreos No Tripulados* (UAS) y los UAV's, los cuales también son conocidos como *Aeronaves Controladas Remotamente RPAS (Remotely Piloted Aircraft Systems)*. Así mientras que en aplicaciones militares la mayor parte de los sistemas utilizan UAV de ala fija aeroplano, para aplicaciones civiles, la maniobrabilidad y capacidad de vuelo estacionario de los multi-rotores ha hecho que la combinación de ellos sea destinado a realizar tareas en las áreas de:

- Climatología: monitorización de la contaminación atmosférica.
- Agricultura: vigilancia y monitoreo de cultivos.
- Levantamiento de mapas: topografía.
- Búsqueda y rescate: naufragios.
- Medioambientales: detección de incendios.
- Comunicaciones: conmutador emergente de comunicaciones.
- Monitorización de autopistas.
- Control de fronteras.
- Ubicación de cardúmenes en altamar para la pesca.

El desarrollo de los últimos prototipos que se utilizan en los UAS y su empleo en misiones civiles, ha permitido la apertura de los sistemas informáticos y el desarrollo de satélites de navegación (GPS / GLONASS). Hoy en día, se presenta una creciente demanda mundial en el uso estos sistemas, dentro de los requisitos que todos los sistemas deben poseer se destacan: relación costo-efectividad, confiabilidad, mantenimiento, utilidad y disponibilidad operacional; además de estos, para tareas se toma en cuenta ciertos requerimientos básicos como:

- a) operaciones diurnas y nocturnas
- b) operatividad en un amplio rango de condiciones ambientales
- c) operación a distintas altitudes
- d) operación más allá de línea de vista
- e) operación en tiempo real

2.3 Aspectos puntuales de los subsistemas.

Dentro de la calificación de UAS se siguen utilizando UAV dirigidos con control remoto, que incorporan una interesante instrumentación necesaria para su estabilización automática e inteligente y para su guiado de manera simple, con instrucciones proporcionadas generalmente por un mando de tipo joystick.

La configuración de un UAV contempla una serie de elementos o sistemas comunes, que van variando según las características concretas y especiales que los definan y lo diferencien entre ellos, que se pueden segmentar como subsistemas:

- Plataforma de vuelo. Se compone del vehículo en sí (normalmente definido como airframe o aircraft) y la carga útil a bordo de éste (cámaras y otros sensores).
- Sistema de control de vuelo o FCS (*Flight Control System*). Se compone de los sensores (IMU, Air data, etc.), sistemas de posicionamiento (GPS, GLONASS), servomecanismos, computador de abordo, etc. integrados en la plataforma de vuelo. El FCS se caracteriza por ser el encargado de georeferenciar la información adquirida por la plataforma en vuelo y contribuir al control y estabilización del UAV.
- Sistema de comunicaciones. Se compone de enlaces de comunicación, vía radio, entre la estación de control de tierra y el vehículo aéreo no tripulado. Se caracteriza por ser el encargado de transferir la información adquirida por la plataforma en vuelo y garantizar la comunicación entre el centro de control y la plataforma de vuelo.
- Estación de Control de Tierra o GCS (*Ground Control Station*). Se compone del conjunto de ordenadores y/o dispositivos que monitorizan y supervisan la plataforma de vuelo y su FCS. Cuenta con un sistema de observación para realizar el análisis (generalmente gráfico) de la información adquirida e integrarla en una aplicación relacionada con el propósito de la misión.

2.4 Características de maniobra UAV:

- *Radio de acción:* Se define como la máxima distancia que el UAV puede alejarse de su punto base con su carga normal de misión y regresar sin una nueva dotación de combustible. Esta distancia es directamente dependiente del nivel de operación de la unidad e idealmente cubre exclusivamente el área de interés.
- *Altitud de operación nominal:* Se puede definir como la altitud donde la carga útil en unión a la cobertura llega a punto deseado de operación.
- *Velocidad máxima y de crucero:* Es dependiente de la potencia del motor y del diseño aerodinámico. El requerimiento de alto rango de cubrimiento entra en conflicto con el requerimiento de velocidad ya que estos usualmente tienen pequeños motores de alta eficiencia y alas de gran envergadura.
- *Velocidad de sostenimiento:* Es usualmente la velocidad óptima para el alcance del rango de cobertura y típicamente la mínima, muy cerca de la velocidad de pérdida.
- *Tasa de ascenso:* Se relaciona con el comportamiento de la relación de velocidad y altitud de la aeronave en vuelo.

2.5 Estación de control de tierra

Esta es encargada de revisar constantemente las variables que afectan la navegación y operación del UAS mientras se encuentre bajo alguna misión. Típicamente es conformada por una consola en tierra la cual muestra gráficamente instrumentos y mapas con las trayectorias recorridas o por recorrer, dependiendo de la construcción de esta es probable que cuente con las modalidades de vuelo autónomo u operado en tierra por una persona.

Los aspectos generales de la GCS (Ground Control System):

-Proveer un sistema de arquitectura abierta la cual pueda soportar futuros requerimientos tales como la expansiones de sistemas la cual debería ser soportada sin reestructurar la GCS completamente.

-Ser escalable, lo cual le permita ser usada en distintas plataformas.

-Ser modular, lo cual le permita a los sistemas cambios con solo el reemplazo del módulo objetivo.

- Ser flexible, esto debido a que los requerimientos del usuario final y las misiones varían, los atributos del sistema puedan ser fácilmente reemplazados.
- Ser capaz de ejecutar mantenimiento por software y presentar en pantalla el estado de resultados.
- Ser ergonómicamente diseñado para el control por parte del operador.
- Capacidad de operación en distintas condiciones ambientales.
- Ser de fácil transporte y desembarque.

2.6 Monitoreo, control y planeamiento de misión

La GCS debe:

- Tener la funcionalidad de permitir al operador generar y procesar los planes de misión de UAV.
- Permitir la reasignación del perfil de misión así como la carga útil durante todas y cualquiera de las fases de misión en las que se encuentre la aeronave.
- Proveer la funcionalidad necesaria para cargar en vuelo un plan de ruta y un plan de carga útil vía link de datos por conexión directa desde ella misma.
- Comprobar automáticamente la viabilidad del plan de ruta trazado antes de ser cargado incluyendo las constantes de altitud, carga útil, link de datos, rango, restricciones aeroespaciales, combustible, efectos del terreno y un plan en caso de pérdida de señal.
- Tener la capacidad de controlar y monitorear la carga útil del UAV, link de datos e interfaces durante la ejecución de la misión.
- Tener la capacidad de controlar y monitorear múltiples UAV.
- Pasar y tomar el control del UAV a otra GCS.
- Implementar un plan de acción de emergencia para controlar el AUV durante fallas en los equipos.
- Monitorear la telemetría y grabar todos los datos para su futuro análisis.

-Recibir, procesar y mostrar todos los datos de salida de la carga útil.

2.7 Seguridad

La GCS debe:

-Tener apropiados sistemas de advertencia en el caso de que los sistemas de UAV entren en un modo de operación inseguro.

-Proveer la información requerida que permita al operador mantener la operación segura de otras aeronaves.

-Ser capaz de reinsertar restaurar la potencia con suficiente tiempo para evitar la pérdida del UAV durante apagones.

-Ser diseñada para proteger los links de comunicación y datos contra maniobras de guerra electrónica o destrucción física.

2.8 Carga útil

Este término se refiere al equipo que es instalado en el UAV con el propósito de desarrollar sus misiones.

La capacidad de carga útil de un UAV es una medida de su peso, talla y capacidades de potencia disponible para realizar funciones por encima y por debajo de su habilidad básica de aterrizar, despegar o volar.

Las cargas útiles de reconocimiento son las más utilizadas por los sistemas de UAV y en muchos casos las más prioritarias. Desde el punto de vista de carga útil para los casos de reconocimiento está dado por sistemas electro-ópticos y radares, la clave de estos sistemas radica en tener la suficiente resolución para "ver" lo suficientemente lejos y con la suficiente cobertura de área además de ser de tamaño reducido, bajo consumo de potencia y accesibles en precio. Adicionalmente en conjunción con otro tipo de sensores, como los sistemas de navegación, la carga útil podría ser utilizada para determinar parámetros de interés de acuerdo al tipo de misión a desarrollar.

2.9 Link de datos

El link de datos es la llave de los subsistemas para cualquier UAV. Este provee comunicación de dos vías ya sea por demanda o continuamente. Un link de subida provee control del patrón de vuelo del vehículo y comanda los sistemas de carga útil. El link de datos de bajada por su parte provee simultáneamente un canal de baja tasa de datos para comandos de confirmación y transmite el estado de información acerca del vehículo así mismo mediante un canal de alta tasa de datos para la carga útil como por ejemplo video y radar.

El link de datos típicamente consiste de dos sistemas: el ADT o terminal de datos en vuelo localizado en la aeronave y el GDT o terminal de datos en tierra localizado generalmente en la GCS.

Dependiendo de la misión y escenarios, los atributos deseables para un link de datos a bordo de un UAV son:

- Disponibilidad de frecuencias: su uso debe ser diseñado para operar en la mayoría de rangos de frecuencia.

- Resistencia a la interferencia no intencional presente en la banda a causa de otros sistemas de RF e incluso por la acción de motores eléctricos de alta frecuencia (tipo brushless).

- Baja capacidad a la interceptación, para usos militares y misiones especiales es altamente deseable para evitar la localización de la GCS, por su largo tiempo en modo estacionario. Esta puede lograr utilizando sistemas de salto en frecuencia, así como de espectro ensanchado en unión a estrategias de bajos ciclos de trabajo y el uso de antenas direccionales.

- Resistencia a tomas de control externo o maniobras de engaño.

2.10 Características particulares de operación

Confiabilidad del sistema.

Este importante parámetro es resultado directo de la madurez del software y del hardware. Fallas en el hardware y *bugs* en el software son comunes durante la etapa de desarrollo del sistema y afectan la confiabilidad del sistema. Se identifica la propulsión y el software como las variables más críticas. Este parámetro debe

ser tomado con cuidado aun en el caso de aeronaves costosas y gran envergadura.

Confiabilidad de misión.

Se define como la probabilidad de que el sistema pueda realizar todas las fases de la misión libre de fallas esto incluye, pre-vuelo, despegue, operación del link de datos entre otros. Muchos de los accidentes ocurren durante la etapa de aterrizaje y los métodos autónomos para la realización de estas maniobras se hacen cada vez más comunes con el fin de reducir el error humano y los problemas relacionados al clima.

Mantenimiento.

En este contexto es la habilidad del sistema de ser retenido o re - almacenado en condición de operación cuando se realiza el mantenimiento por personal capacitado teniendo en cuenta procedimientos pre-escritos por el fabricante.

Movilidad.

Es la capacidad del sistema de transportarse de un lugar a otro mientras retiene su capacidad de cumplir con su misión primaria.

Sostenibilidad

Es la habilidad para mantener el nivel necesario y duración de la actividad operacional para el logro de los objetivos en campo, este se encuentra en función de su capacidad de mantener los niveles de fuerza disponible, materiales y consumibles necesarios para soportar los esquemas de operación.

Seguridad

La medida de este parámetro debe ser tomada llevando al mínimo cualquier evento riesgoso sobre el personal o sobre bienes que pudiesen resultar afectados por un impropio comportamiento del sistema. Los modos de operación y

terminación de emergencia deben ser tomados en cuenta, además de poseer sistemas redundantes de cada bloque crítico de la aeronave.

Intercambiabilidad y modularidad

El conjunto de estos hace referencia a la capacidad del sistema de intercambiar partes averiadas sin ninguna alteración del comportamiento de la aeronave de una manera eficiente y sin implicar la sustitución de módulos adyacentes.

Potencial de crecimiento

Los sistemas de UAV deben facilitar la actualización para acomodar varias cargas útiles de sensores. El potencial de crecimiento podría darse en estas áreas:

- Extensión del rango de carga útil.
- Aumentar la eficiencia capacidad de carga tanto en volumen como en peso.
- Capacidad de la aeronave de resistir a cualquier interferencia.
- Reservar recursos de computador para futuras actualizaciones.

Capítulo 3. Navegación y vuelo en un UAV.

La navegación es una parte vertebral en los UAV, la cual se define como una serie de decisiones o algoritmos basados en mediciones tales como la ubicación geográfica, la velocidad, el tiempo y las distancias. De esta dependen las trayectorias que ha de tomar el modelo para lograr llegar de un punto de salida al punto o puntos deseados de una manera eficiente.

Existen cinco formas básicas de navegación [1];

1. Pilotaje: esencialmente se enfoca en el reconocimiento de puntos de referencia
2. Navegación a estima: se basa en saber el punto de partida más un tipo de información de rumbo y algunos estimados de la velocidad
3. Navegación Celestial: usando el tiempo y los ángulos entre una vertical local y objetos celestiales conocidos.
4. Radio navegación: por medio de radio frecuencia con fuentes de locaciones conocidas incluyendo GPS y GLONASS
5. Navegación Inercial: se basa sabiendo la posición inicial, velocidad, y la actitud y después de esto midiendo las tasas de actitud y aceleraciones. Esta es la única forma de navegación que no se basa en referencias externas.

3.1 Ejes del UAV, movimientos de Pitch, Roll, Yaw.

Antes de poder conocer la orientación y actitud del vehículo es necesario definir los movimientos o grados de libertad que este tiene en el espacio.

La capacidad general de cualquier aeronave en vuelo es la de rotar alrededor de su centro de gravedad, donde tienen origen tres ejes perpendiculares. Estos momentos generalmente son manipulados por superficies de control y en algunos casos particulares de los UAV de despegue vertical, Vertical Take Off Landing (VTOL), se controla la cantidad de revoluciones en los motores.

Eje longitudinal: se encuentra alineado con el eje de referencia x, inicia en el centro de masa y tiene la dirección hacia el frente del modelo. El movimiento alrededor de este eje nos proporciona el ángulo de alabeo o también llamado roll, y su efecto es el movimiento hacia la derecha o la izquierda. (fig. 3.1)

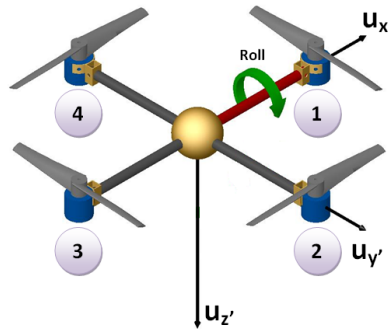


Figura 3. 1 Movimiento de Alabeo o Roll. [4]

Eje lateral: este se encuentra perpendicularmente al eje longitudinal y el movimiento en torno a este eje provoca un movimiento hacia adelante o atrás. El movimiento alrededor del eje es denominado cabeceo o pitch. (fig. 3.2)

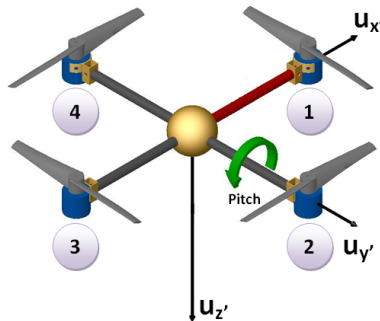


Figura 3. 2 Movimiento de cabeceo o pitch. [5]

Eje Vertical: se encuentra a 90° respecto a los ángulos de cabeceo y de alabeo, conocido como ángulo de guiñada o yaw. El momento en este eje causa un giro del modelo sin afectar el plano X-Y, es decir con causa elevación ni descenso particularmente para el modelo utilizado.

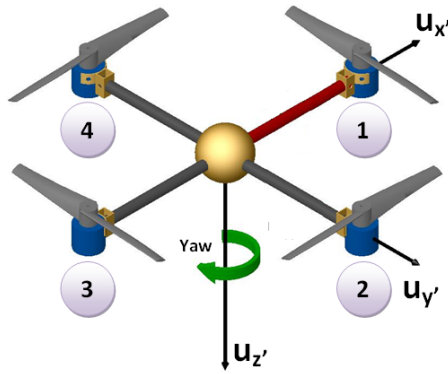


Figura 3.3 Movimiento de guiñada o yaw. [6]

Podemos entonces definir la orientación de la aeronave por la cantidad de rotación de la aeronave a lo largo de estos tres ejes ortogonales [2]. (fig. 3.4)

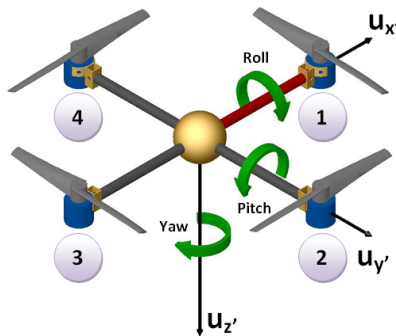


Figura 3.4 Momentos alrededor de los ejes ortogonales de un UAV. [7]

3.2 Sistema de Navegación Inercial (INS)

Esta es una técnica de navegación autónoma que depende enteramente de mediciones inerciales la cual se sustenta en un principio básico de la cinemática:

“Conocidos en un instante inicial la velocidad, la orientación y la posición de un móvil así como los valores instantáneos presentes y futuros de su aceleración lineal y su velocidad angular relativas a un dado sistema de referencia, es posible calcular la posición, la velocidad y la orientación del vehículo en todo instante futuro”.[3]

Este sistema se encuentra compuesto por una Unidad de Medición Inercial (IMU), el cual contiene tres giroscopios y tres acelerómetros ortogonales, los cuales miden la velocidad angular y la aceleración lineal respectivamente y en algunos casos es posible incluir la medición magnética de la tierra por medio de magnetómetros de igual manera ortogonales para obtener la orientación entre dos marcos de referencia, uno estático (en tierra) y el otro móvil (en el UAV). (fig. 3.5)

Mediante el procesamiento de los datos obtenidos a partir de estas unidades es posible realizar un seguimiento de la posición y la orientación actual del marco de referencia en la aeronave respecto a la posición inicial o marco de referencia en tierra conocidos el instante inicial la velocidad, la orientación y la posición así como los valores instantáneos presentes y futuros de su aceleración lineal y su velocidad relativas dado un sistema de referencia es posible calcular la posición, la velocidad y la orientación del vehículo en todo instante futuro de aceleraciones.[4]

Una vez obtenido los datos anteriores a manera de representar matemáticamente dicho comportamiento es por medio de una matriz conocida como DCM (*Direction Cosine Matrix*) o matriz de rotación, la cual nos permite trasladar un vector de un marco de referencia global hacia un marco de referencia local o viceversa, proporcionando la orientación relativa del UAS. (fig. 3.6)

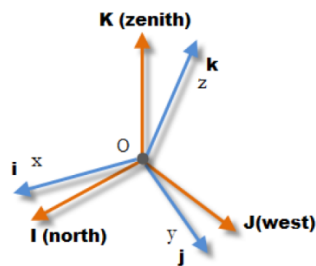


Figura 3.5 Marcos de referencia del INS. [8]

$$\begin{bmatrix} \cos(xx) & \cos(xy) & \cos(xz) \\ \cos(yx) & \cos(yy) & \cos(yz) \\ \cos(zx) & \cos(zy) & \cos(zz) \end{bmatrix}$$

Figura 3.6 Matriz DCM [9]

La figura 3.6 muestra la estructura de la matriz DCM a partir de las mediciones de la IMU. (Ver anexo A).

A partir de la matriz anteriormente vista es posible obtener la representación de los movimientos alrededor de los ejes por medio de ángulos de Euler, los cuales indican el ángulo que se ha rotado en torno a uno de los tres ejes de referencia (I, J, K de acuerdo a la figura 3.5), cada uno de estos ángulos corresponde a los movimientos de roll (ϕ), pitch (θ), yaw (ψ).

La conversión de la representación matricial a los ángulos de Euler viene dada por:

$$\mathbf{R} = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix} \quad \begin{aligned} \phi &= \text{Atan2} \left(-r_{31}, \sqrt{r_{11}^2 + r_{21}^2} \right) \\ \psi &= -\text{Atan2} \left(\frac{r_{21}}{\cos(\phi)}, \frac{r_{11}}{\cos(\phi)} \right) \\ \theta &= \text{Atan2} \left(\frac{r_{32}}{\cos(\phi)}, \frac{r_{33}}{\cos(\phi)} \right) \end{aligned}$$

Figura 3.7 Conversión de Matriz DCM a ángulos de Euler [10]

3.3 Instrumentos básicos de vuelo

Se denominan instrumentos de vuelo a los que se encuentran dedicados a medir variables tales como la altura, velocidad, presión atmosférica, rumbo magnético y movimiento relativo al horizonte, los cuales permiten conocer el comportamiento de un UAV para tomar las decisiones pertinentes. Posterior a la adquisición de dichos datos en un ambiente digitalizado estos se procesan en la computadora de abordo (en el caso de contar con una), para su pilotaje autónomo y por otro lado son enviados a la estación terrena para el monitoreo de la aeronave o en su defecto el pilotaje desde tierra.

3.3.1 Horizonte artificial

Este instrumento tiene como objetivo la estimación de la posición de los ejes longitudinal y transversal de la aeronave respecto al horizonte terrestre. Este permite conocer de forma rápida los ángulos de cabeceo y alabeo. Su función principal es proporcionar una referencia inmediata de la posición respecto al campo de gravedad sin entrar a detalle en una revisión numérica exhaustiva.



Figura 3. 8 Interpretación grafica del horizonte artificial. [11]

Este se basa en mediciones de giroscopios por medio de la fuerza centrífuga proporcionada por una masa en rotación (ver Fig. 3.9), pero con la integración de las nuevas tecnologías electrónicas es posible sustituir los modelos que actúan

en base a estos movimientos por circuitos a base de MEMS logrando incluir acelerómetros y por medio de cálculos matemáticos es posible obtener mediciones más precisas en una área mucho más pequeña.



Figura 3. 9 Componentes de un horizonte artificial mecánico. [12]

3.3.2 Indicador de altura (Altímetro) y presión barométrica

Este instrumento obtiene la altura por medio del cambio de presión atmosférica típicamente por medio de un barómetro el cual realiza lecturas dentro de una capsula barométrica y el exterior, proporcionando una diferencia de alturas entre el punto en el que se encuentre localizado y otro de referencia. Aproximadamente la presión atmosférica disminuye en el orden de 1hPa por cada 8.2 metros.



Figura 3. 10 Caratula de un altímetro barométrico. [13]

3.3.3 Indicador de velocidad

Este indicador se basa en el empleo de un tubo de pitot el cual determina la rapidez del UAV respecto al aire que lo rodea. Este está conformado por dos tubos uno dentro del otro, separados herméticamente, donde el tubo interno enfrenta al viento recogiendo la presión de impacto interpretándose como una presión total, esta es asociada a la velocidad del aire mientras tanto el tubo externo con orificios tangenciales proporciona la lectura de una presión estática independiente del aire en movimiento. La diferencia de estas presiones es llamada presión dinámica, mediante la cual es posible obtener la velocidad en vuelo.

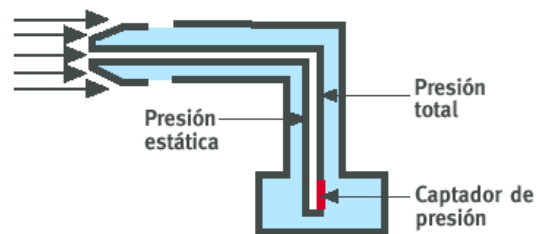


Figura 3. 11 Configuración de un tubo de pitot. [14]

3.4 Instrumentos de Navegación (Rumbo)

El conjunto de estos tiene como finalidad proporcionar la información necesaria sobre la posición actual y el rumbo al que se dirige el UAV para así realizar las correcciones necesarias por medio de aceleraciones en distintas proporciones de sus motores, tomando las decisiones ya sea a bordo o por medio de la estación terrena y por tanto lograr completar exitosamente la misión. Esta sección se explican las características de los instrumentos de navegación utilizados en este proyecto.

3.4.1 Brújula

La brújula, también llamada compás magnético, es un instrumento que al orientarse con las líneas de fuerza del campo magnético de la tierra, proporciona una indicación permanente del rumbo del UAV respecto al Norte magnético

terrestre. Este instrumento es una referencia básica para mantener la dirección de vuelo. Este tipo de instrumento está sujeto a errores provocados por la curvatura del campo magnético terrestre y de igual manera las líneas de campo magnético no son exactamente una línea recta por lo que se deben tener precauciones para corregir este problema. Para distancias cortas hablando de un par de kilómetros este efecto no es de gran consideración solo habrá que realizar los ajustes necesarios de la ubicación del norte.



Figura 3. 12 Compás magnético. [15]

3.4.2 GPS

El GPS (Global Position System), es una herramienta que actualmente tiene como uso principal la navegación por medio de estaciones de referencias a bordo de satélites en órbita alrededor de la tierra. NAVSTAR (GPS) es un sistema de radionavegación mundial que es cubierto por 24 o más satélites desarrollado por el Departamento de Defensa de USA, el cual está distribuido en seis orbitas circulares.



Figura 3. 13 Constelación de satélites. [16]

Asimismo hay una segunda configuración llamada Global Orbiting Navigation Satellite System (GLONASS), puesto en órbita por la Unión Soviética y ahora mantenido por la Republica de Rusia.

Esta última también usa 24 satélites pero estos están distribuidos aproximadamente en tres planos orbitales. Cada satélite orbita la tierra dos veces en un día y trabajan en la banda de ultra alta frecuencia (UHF) en dos rangos L1 (1575.42MHz) y L2 (1227.60MHz).

Una última configuración que es la única en operación de origen civil es el GALILEO, desarrollado por la comunidad europea y puesta en marcha en 2014, considerado de segunda generación puesto que abordo cada satélite lleva consigo dos relojes atómicos de muy alta calidad.

La constelación de satélites constituye un sistema de referenciación absoluta que permite a un receptor alcanzado por las señales satelitales medir su distancia respecto de cada satélite y determinar, por triangulación, su propia posición en coordenadas geográficas preestablecidas. [5]

Uno de los beneficios que caracterizan a este tipo de herramienta es que no le afectan fenómenos atmosféricos como la nieve, la lluvia, los huracanes, etc. Además una característica particular de estos sistemas es que se requiere de la visibilidad de cierta cantidad de satélites para obtener una lectura correcta. (Tabla 3.1)

La aplicación del posicionamiento por medio de satélites está ligado a los campos militares y civiles tales como la aviación, orientación de naves espaciales, actividades marítimas, uso en tierra.

Tabla 3. 1 Características orbitales de las principales constelaciones satelitales [2]

GNSS	Planos orbitales	Satélites/orbita	Inclinación	Periodos	Satélites visibles	Altura
GPS	6 a 60°	4+1	55°	12hs	≥ 4	20200Km
GALILEO	3 a 120°	10+1	56°	14hs	≥ 6	23600Km
GLONASS	3 a 120°	7+1	64,8°	11,25hs	≥ 5	19100Km

CAPÍTULO 4. Comunicaciones

En este capítulo se abordan los distintos tipos de protocolos de redes alámbricas e inalámbricas que son de interés en un UAV de baja altura y bajo alcance, tales como son los aplicables a la transmisión de datos por medio del aire, así como los propios en la comunicación con los periféricos utilizados en la FMS (*Flight Management System*), mostrando sus ventajas, alcances y limitaciones.

4.1 Estructura de un sistema de comunicación

Se puede definir la comunicación como el proceso por medio del cual la información se transfiere de un lugar fuente a otro destino. [1]

El esquema más general de un sistema de comunicaciones es el siguiente (fig. 4.1).

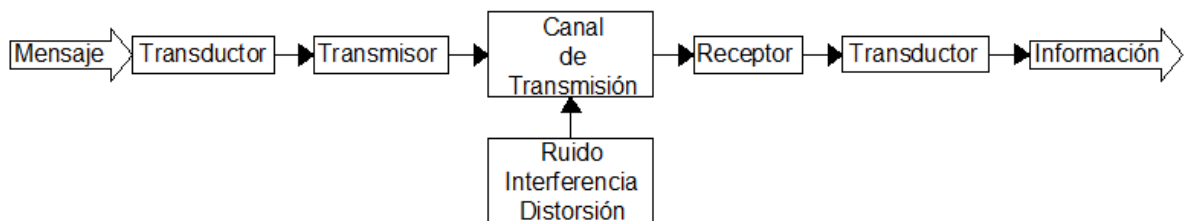


Figura 4.1 Sistema básico de Comunicaciones. [17]

El mensaje que se requiere enviar es convertido a una señal eléctrica por medio de un transductor, en la etapa de transmisión el mensaje se modula con una señal llamada portadora para poder propagarse en el canal de una manera adecuada.

Después de recibir la señal en el lado receptor, esta se demodula para recuperar el mensaje y en el transductor se acondiciona la señal para el uso de la información transmitida.

Existen varios tipos de canales por lo que se clasifican como guiados (cable, fibra óptica) y no guiados (medio radioeléctrico).

4.2 Comunicaciones inalámbricas

Es un hecho que lograr comunicar la GCS y el UAS se requiere realizar una transmisión de datos de bajada para efectos de este proyecto. La distancia juega un papel importante en la decisión del tipo de tecnología inalámbrica seleccionada para su uso en las diferentes tareas que involucran UAS. Particularmente el enfoque es hacia la transferencia de datos que permitan conocer el estado de la aeronave en vuelo.

Para delimitar las tecnologías aplicables a la demanda de comunicaciones de un proyecto específico es necesario conocer el rango de frecuencias y sus aplicaciones, así como los usos restringidos previamente establecidos.

En la Tabla 4.1 se indican los nombres de las bandas que pertenecen al rango de radiofrecuencias en función de su frecuencia y longitud de onda y se señala la banda en uso en este proyecto.

Tabla 4. 1 Denominación de las bandas de RF. [3]

Nombre	Banda	Frecuencia	Longitud de onda	Uso
		< 3 Hz	>100,000 km	
Extra baja frecuencia (Extremely low frequency)	ELF	3-30 Hz	100,000-10,000 km	No se usa en radiofrecuencia
Super baja frecuencia (Super low frequency)	SLF	30-300 Hz	10,000-1,000 km	Comunicaciones submarinas
Ultra baja frecuencia (Ultra low frequency)	ULF	300-3000 Hz	1,000-100 km	comunicaciones militares secretas
Muy baja frecuencia (Very low frequency)	VLF	3-30 kHz	100-10 km	Comunicaciones militares y gubernamentales
Baja frecuencia (Low frequency)	LF	30-300 kHz	10-1 km	Comunicaciones aéreas y marítimas
Frecuencia media (Medium frequency)	MF	300-3000 kHz	1000-100m	Radiodifusión
Alta frecuencia (High frequency)	HF	3-30 MHz	100-10 m	seguridad, defensa, onda corta, radioaficionados
Muy alta frecuencia (Very high frequency)	VHF	30-300 MHz	10-1 m	Televisión, radio FM, aviación, satélites, servicio marítimo
Ultra alta frecuencia (Ultra high frequency)	UHF	300-3000 MHz	1m- 100mm	Televisión, radiotransmisiones uso personal, telefonía móvil, militar
Super alta frecuencia (Super high frequency)	SHF	3-30 GHz	100-10mm	Televisión vía satélite, radioenlaces, radar
Extra alta frecuencia (Extremely high frequency)	EHF	30-300 GHz	10-1 mm	Radioastronomía, radar alta resolución
		> 300 GHz	<1mm	

Existen diferentes tipos de comunicaciones inalámbricas que pueden ser de utilidad en función de la aplicación que se requiera, siendo la distancia y la

cantidad de datos posibles a transmitir fundamentales en la decisión del tipo de tecnología seleccionada, entre las que destacan:

- Infrarrojo
- Radiofrecuencia
- Microondas

4.2.1 Comunicación infrarroja.

La comunicación infrarroja es una tecnología de transmisión inalámbrica por medio de ondas de calor a corta distancia (hasta 1 m), esto debido a la longitud de onda pequeña (850-900 nm) lo cual hace que no pueda propagarse de la misma forma en que lo hacen las señales de radio.

Esta típicamente tiene una velocidad promedio de transmisión de datos hasta de 115 Kbps. Permitiendo una comunicación unidireccional entre dos nodos (emisor-receptor) en línea recta (Fig. 4.2).

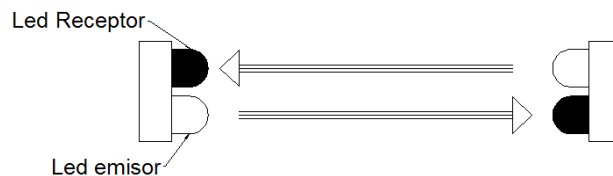


Figura 4.2 Comunicación Infrarroja. [18]

4.2.2 Comunicación por Radiofrecuencia.

Es la porción del espectro electromagnético en la cual se pueden generar ondas electromagnéticas a por medio de circuitos eléctricos y electrónicos y emitirlas a la atmósfera por medio de una antena. La ubicación de este tipo de ondas en el espectro electromagnético es en el rango de frecuencias que van desde los 3 Hz hasta los 300Ghz. Las comunicaciones inalámbricas por RF se pueden dividir en las que cumplen un protocolo estándar (libres) y las que no (propietarias).

La distancia para lograr un enlace entre transmisor-receptor por medio de estas ondas depende de la homogeneidad del canal, condiciones atmosféricas, potencia y tipo de antena y longitud de onda en la que se esté trabajando.

La potencia de salida del circuito transmisor y la sensibilidad del circuito receptor dependen de la calidad y del tipo de antena (externa, cerámica o de circuito impreso) y de sus características.

El canal de transmisión por el cual se radian las señales influye en la calidad de señal recibida y en la distancia máxima que podrá operar una antena pues este normalmente atenúa la señal transportada.

4.2.2.1 Redes inalámbricas.

Las redes inalámbricas se basan en un enlace que utiliza ondas electromagnéticas en lugar de cableado estándar. La variedad de tecnologías se pueden identificar fácilmente por el protocolo de comunicación.

La estructura mínima de una red inalámbrica es entre dos puntos, en la cual puede existir intercambio de información en uno o en ambos sentidos bajo características únicas definidas por el sistema y sus protocolos en uso.

Estas permiten que los dispositivos remotos se conecten sin dificultad, ya sea que se encuentren a unos metros de distancia o a varios kilómetros.

En la actualidad los tipos de redes son variados y se diferencian por la distancia que pueden cubrir, la cantidad de usuarios que pueden integrarse a cada una de ellas. Este tipo de comunicaciones se encuentran bajo estándares emitidos por IEEE (*Institute of Electrical and Electronics Engineers*).

Las figuras 4.3 y 4.4 muestran los diferentes tipos de redes inalámbricas y protocolos establecidos para cada una de ellas:

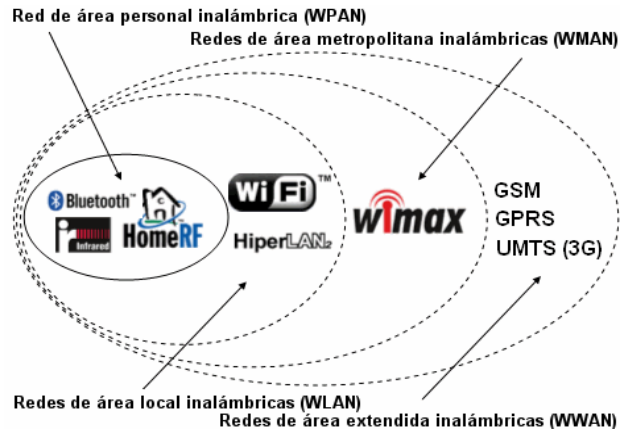


Figura 4. 3 Categorías de redes inalámbricas. [19]

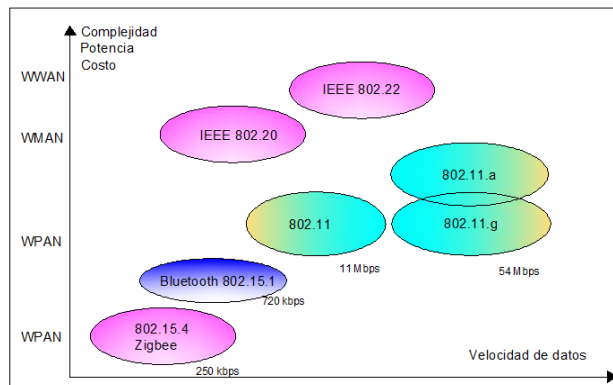


Figura 4.4 Comparativo de velocidades en estándares de redes inalámbricas. [20]

Wireless Personal Area Network (WPAN).

Estas regularmente incluyen redes inalámbricas de corto alcance que normalmente cubren distancias de decenas de metros, facilitan la intercomunicación de dispositivos personales como pueden ser teléfonos móviles y electrodomésticos, PDA's, impresoras.

La tecnología principal WPAN es **Bluetooth®** también conocida como *IEEE 802.15.1*. Ofrece una velocidad máxima de 1 Mbps con un alcance máximo de unos treinta metros.

La Tecnología Bluetooth es de pequeña escala, bajo costo y se caracteriza por usar enlaces de radio de corto alcance entre móviles y otros dispositivos, como

teléfonos celulares, puntos de accesos de red (Access points) y computadoras (fig. 4.5).

El transmisor está integrado en un pequeño microchip de 9x9 milímetros y opera en una frecuencia de banda global (2,4 GHz, utilizada en muchos países para usos médicos y científicos) que asegura la compatibilidad universal. Los dispositivos que incorporan Bluetooth se reconocen y se hablan de la misma forma que lo hace una computadora con su impresora. El canal permanece abierto y no requiere la intervención directa y constante del usuario cada vez que se quiere enviar algo [2].

Las características notables de bluetooth son:

- Permitir la comunicación sencilla entre dispositivos fijos y móviles.
- Evitar la dependencia de cables que permitan la comunicación.
- Permitir la creación de pequeñas redes de forma inalámbrica.
- Proveer un camino fácil para el intercambio de información.

El transmisor permite enviar voz y datos a una velocidad máxima de 700 Kbps y consume un 97% menos que un teléfono móvil. Además, es inteligente: cuando el tráfico de datos disminuye el transmisor adopta el modo bajo de consumo de energía.

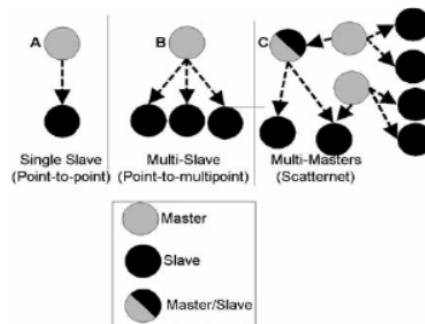


Figura 4.5 Clasificación de Clases de Conexiones en Bluetooth. [21]

Wireless Local Area Network (WLAN).

Esta es una red que cubre un área equivalente a la red local de una empresa, con un alcance aproximado de cien metros. Permite que las terminales que se encuentran dentro del área de cobertura puedan conectarse entre sí.

Uno de los varios tipos de tecnologías es: Wifi (ó IEEE 802.11) con el respaldo de WECA (*Wireless Ethernet Compatibility Alliance*) ofrece una velocidad máxima de 54 Mbps en una distancia de varios cientos de metros.

Wireless Metropolitan Area Network (WMAN).

También son conocidas como bucle local inalámbrico (*WLL, Wireless Local Loop*). Las WMAN se basan en el estándar *IEEE 802.16*. Los bucles locales inalámbricos ofrecen una velocidad total efectiva de 1 a 10 Mbps, con un alcance de 4 a 10 kilómetros.

La mejor red inalámbrica de área metropolitana es WiMAX, que puede alcanzar una velocidad aproximada de 70 Mbps en un radio de varios kilómetros.

Wireless Wide Area Network (WWAN).

Estas redes inalámbricas de área extensa tienen el alcance más amplio de todas las redes inalámbricas. Por esta razón, todos los teléfonos móviles están conectados a una red inalámbrica de área extensa.

Las tecnologías principales son:

- GSM (*Global System for Mobile Communication*)
- GPRS (*General Packet Radio Service*)
- UMTS (*Universal Mobile Telecommunication System*)

Wireless Sensor Networks

Una red de sensores inalámbricos es una infraestructura comprendida entre el sensado (medición), computo de datos y transmisión vía radio. Lo cual permite a un administrador monitorear el comportamiento de distintos sensores y así evaluar las acciones que en su caso se requieran. La distancia actualmente cubre

hasta 45km en una línea de vista sin obstáculos. La tecnología ZigBee se encuentra dentro de esta clasificación de reciente creación.

Estos se basan en el uso de dispositivos con un consumo bajo de energía hablando y una tasa de transmisión baja que se encuentra alrededor sobre los Kbps.

4.2.2.2 Estándares utilizados en las redes inalámbricas.

Para realizar una transmisión con una alta tasa de éxito es necesario utilizar ciertas reglas, los estándares emitidos por el IEEE realizan esta función con la finalidad de estandarizar las comunicaciones de acuerdo a la tecnología en uso y así minimizar errores.

Estándar IEEE 802.15.1 Bluetooth

Trabaja a una velocidad de transmisión de datos de 1 Mbps. Se puede ver que Bluetooth y ZigBee tienen similares corrientes en transmisión, pero ZigBee tiene un recurso significativamente mejor, más baja corriente en modo espera. Esto es debido a que los dispositivos en redes Bluetooth deben dar información a la red frecuentemente para mantener la sincronización, así que no pueden pasarse fácilmente a modo inactivo.

Estándar Wi-Fi (IEEE 802.11)

Posee las características de una red de área local inalámbrica (WLAN). Wi-Fi (fidelidad inalámbrica por sus siglas en inglés) es el nombre de la certificación otorgada por la Wi-Fi Alliance, que garantiza la compatibilidad entre dispositivos que utilicen esta tecnología.

Con Wi-Fi se pueden crear redes de alta velocidad siempre y cuando el equipo que se vaya a conectar no esté muy alejado del punto de acceso, esta tecnología admite computadoras portátiles, equipos de escritorio, asistentes digitales personales (PDA) o cualquier otro tipo de dispositivo con propiedades de conexión de 11 Mbps o superior dentro de un radio de varias docenas de metros en ambientes cerrados (de 20 a 50 metros en general) o dentro de un radio de aproximadamente cien metros al aire libre.

El estándar 802.11 permite un ancho de banda de 1 a 2 Mbps, éste se ha modificado para optimizar el ancho de banda (802.11a, 802.11b y 802.11g) o para especificar componentes de mejor manera con el fin de garantizar mayor seguridad o compatibilidad.

Los estándares 802.11a, 802.11b y 802.11g, llamados estándares físicos, son modificaciones del estándar 802.11 y operan de modos diferentes, lo que les permite alcanzar distintas velocidades en la transferencia de datos según sus rangos (Tabla 4.2).

Tabla 4. 2 Características de estándares físicos. [4]

Estándar	Frecuencia	Velocidad	Rango
WiFi (802.11a) a	5 GHz	54 Mbit/s	10 m
WiFi (802.11b) b	2.4 GHz	11 Mbit/s	100 m
WiFi (802.11b) g	2.4 GHz	54 Mbit/s	100 m

Estándar IEEE 802.15.4

Este es un estándar define el nivel físico y el control de acceso al medio de redes inalámbricas de área personal con tasas bajas de transmisión de datos (low-rate wireless personal área network, LR-WPAN)

- Velocidades de datos de 250 kbps, 40 kbps, 20 kbps
- Modos de direccionamiento IEEE de 16 y 32 bits.
- Soporte para dispositivos de latencia críticos como palancas joystick
- Creación automática de la red por el coordinador
- Administración de energía para asegurar un bajo consumo energético
- Acceso al canal por medio de CSMA/CA (Carrier Sense Multiple Access/Collision Avoidance) evitando colisiones introduciendo una demora aleatoria antes de intentar hacer uso del canal
- Cada paquete debe ser reconocido mediante un el envío de un ACK al transmisor [3].

- 16 canales en la banda ISM de 2.4Ghz, 10 canales en el 915 MHz y un canal en la banda de 868MHz.

Sobre este se define la especificación de ZigBee, la cual ha sido específicamente creada para redes de control y de sensores cuyo propósito es ofrecer una solución completa para este tipo de redes, construyendo los niveles superiores de la pila de protocolos que el estándar no cubre.

El propósito del estándar es definir los niveles de red básicos para dar servicio a un tipo específico de red inalámbrica de área personal (WPAN) centrada en la habilitación de comunicación entre dispositivos en todas partes, con bajo costo y velocidad (en contraste con esfuerzos más orientados directamente a los usuarios medios, como Wi- Fi). Se enfatiza el bajo costo de comunicación con nodos cercanos y el tipo de red, para favorecer aún más el bajo consumo.

4.2.2.3 Protocolo inalámbrico ZigBee®.

En esta sección se dan a conocer los aspectos fundamentales del protocolo inalámbrico ZigBee®, para entender la manera de envío de datos y proceder correctamente al acoplamiento de los sistemas circundantes a esta tecnología. Esta tecnología posee la característica de alcanzar distancias desde 30 metros con muy baja energía y hasta casos en los que se utilicen distancias de poco más de 15 km obviando el incremento de energía requerida con la distancia y la cantidad de datos transmitidos por segundo.

Zigbee es una tecnología creada por ZigBee Alliance bajo el estándar IEEE 802.15.4 de redes inalámbricas de área personal WPAN, con objetivos orientados a redes inalámbricas seguras con baja tasa de envío de datos y maximización del uso energético, creada principalmente las redes de control y de sensores [4].

Posee una arquitectura basada en el modelo OSI (figura 4.6), en la cual se define en las dos capas más bajas: la Capa Física y la Subcapa de Control de Acceso al Medio de la Capa de Enlace de Datos, la cual se encarga de aislar los detalles de las tecnologías físicas a la capa de control de acceso al medio. Estas capas son utilizadas por ZigBee para crear un marco de trabajo para las aplicaciones.

El método de codificación de canal usado en capa física (PHY), es el de espectro ensanchado por secuencia directa (Digital Spread Spectrum Signaling, DSSS), por lo que puede trabajar en uno de dos bandas de frecuencias: 868 MHz, 915

MHz y en el caso de la banda de 2.4 GHz usa señalización ortogonal (4 bits por símbolo), con velocidades de hasta 20 Kbps, 40 Kbps y 250 Kbps respectivamente.

Para la capa de enlace el acceso al canal (MAC), se efectúa por medio de CSMA/CA (Carrier Sense Multiple Access/Collision Avoidance) evitando colisiones introduciendo una demora aleatoria antes de intentar hacer uso del canal. Para los modos de direccionamiento se utiliza el IEEE de 16 bits y cada dispositivo tiene una dirección MAC única de 64 bits.

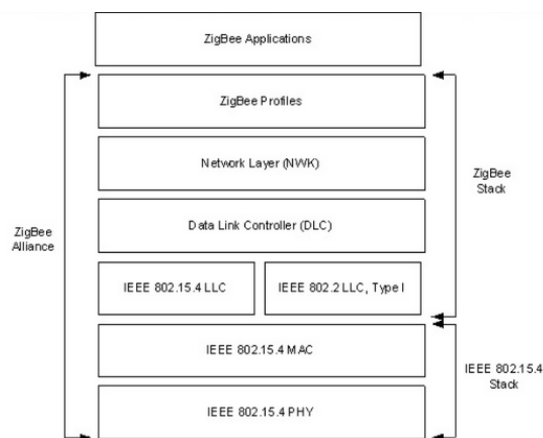


Figura 4.6 Pila del protocolo IEEE 802.15.4 y ZigBee. [22]

Este protocolo tiene como misión permitir una comunicación inalámbrica confiable, con bajo consumo de energía y bajas tasas de transmisión para aplicaciones de monitoreo y control. El alcance normal de un dispositivo con antena dipolo en visión directa suele ser aproximadamente de 100m y en interiores de unos 30m para elementos de baja potencia, y de hasta 15.5Km con otro tipo de antena.

La velocidad de transmisión de datos de una red ZigBee® varía desde los 10Kbps hasta los 256kbps.

Conexión punto-punto.

La conexión punto-punto corresponde a una arquitectura de red en la que cada canal de datos se utiliza para comunicar dos nodos y que en su caso cualquier dispositivo puede tomar la característica de esclavo-maestro, realizando una transacción de datos half-dúplex (ver Figura 4.7), la cual permite el flujo

bidireccional de información solo en un sentido por cada unidad de tiempo, siendo esta configuración la utilizada en este proyecto.



Figura 4.7 Operación en modo transparente (half-dúplex). [23]

ZigBee® está conformado para crear y coordinar redes de hasta 65535 elementos, en este caso solo se requiere formar una red con un nodo. Para crear una conexión punto-punto se usarán dos denominaciones de firmware:

1. Coordinador ZigBee® (ZigBee® Coordinator, ZC). Se encargará de coordinar al único nodo requerido. Típicamente sus funciones en una red de más de un nodo son las de encargarse de controlar la red y gestionar los caminos que deben seguir los dispositivos para conectarse entre ellos.
2. Dispositivo final (ZigBee® End Device, ZED). Posee la funcionalidad necesaria para comunicarse con su nodo padre, pero no puede transmitir información destinada a otros dispositivos en caso de redes de más de un nodo. De esta forma, este tipo de nodo puede estar inactivo el tiempo en que no se requiera transmisión-recepción de datos, aumentando la vida media de las baterías de alimentación, haciéndolo apto para el uso en un UAV.

Dependiendo de las cualidades requeridas se tienen dos modos de operación:

a) Dispositivo de funcionalidad completa (FFD):

También conocidos como nodo activo. Es capaz de recibir mensajes en formato 802.15.4 gracias a la memoria adicional y a la capacidad de procesar, característica que lo hace funcional para recibir datos desde el UAV.

b) Dispositivo de funcionalidad reducida (RFD):

También conocido como nodo pasivo. Tiene capacidad y funcionalidad limitadas con el objetivo de conseguir un bajo costo y una gran simplicidad. Básicamente, son los sensores/actuadores de la red.

Un nodo ZigBee, tanto activo como pasivo, reduce su consumo gracias a que puede permanecer inactivo la mayor parte del tiempo, incluso muchos días seguidos. Cuando se requiere su uso, es capaz de activarse en un periodo de tiempo muy corto, para volverse a desactivar cuando deje de ser requerido. Un nodo cualquiera puede ser activado en aproximadamente 15 ms.

Canales de operación.

El número de canales de transmisión también depende de la banda utilizada, existen 16 canales para 2.4 GHz en la banda ISM (*Industrial, Scientific, Medical*), 10 para las de 915 MHz y uno si la comunicación se realiza en la banda de 868 MHz. En la Figura 4.8 se pueden observar los canales de comunicación ZigBee para las distintas frecuencias de operación. Se muestra, además, la separación que existe entre las frecuencias centrales de cada canal.

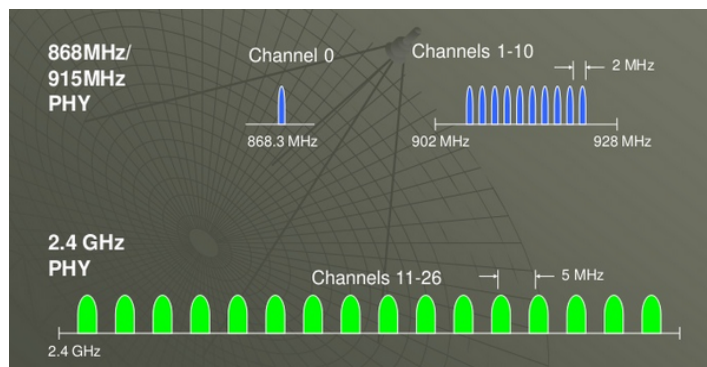


Figura 4.8 Canales de operación para ZigBee. [24]

Técnicas de Modulación.

La modulación que ofrece ZigBee® en cada una de sus bandas es:

- 2.4 GHz con tasa máxima de transferencia de 250 Kbps, para este caso, modula en O-QPSK (Modulación con desplazamiento de fase en cuadratura con desplazamiento temporal), la cual consiste en realizar una transición de fase en cada intervalo de señalización de bits, por portadora en cuadratura en la que se utilizan dos bits para formar cada símbolo. En

las figuras 4.9 y 4.10 se muestra la similitud con la modulación QPSK, la única diferencia es que existe un corrimiento de fase.

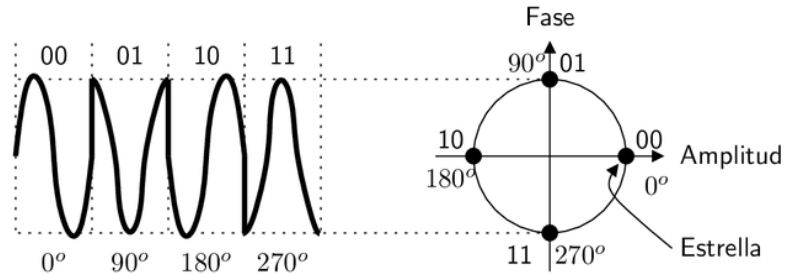


Figura 4.9 Modulación QPSK. [25]

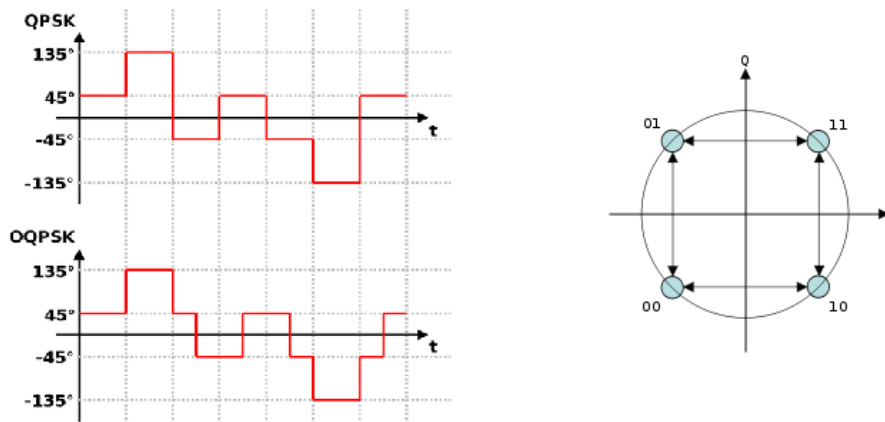


Figura 4.10 Diferencias entre QPSK y O-QPSK y constelación para O-QPSK. [26]

- 868-928 MHz para tasa de datos entre 20 y 40 Kbps, para este otro, modula en BPSK (Modulación con desplazamiento de fase binaria), en esta se tienen como resultados posibles dos fases de salida para la portadora con una sola frecuencia. Una fase de salida representa un 1 lógico y la otra un 0 lógico. Conforme la señal digital de entrada cambia de estado, la fase de la portadora de salida se desplaza entre dos ángulos que están 180° fuera de fase (fig. 4.11).

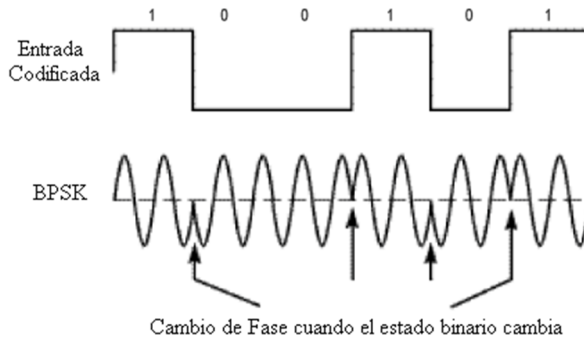


Figura 4.11 Modulación BPSK. [27]

Comparativa de tecnologías inalámbricas.

Los estándares inalámbricos ZigBee y Wi-Fi, son los que contienen características deseables para realizar la tarea de monitoreo, para los de requerimientos de distancia utilizados micro-UAS, en el caso de Bluetooth debido a que la distancia es limitada este se descarta como opción, para el Wi-Fi también se tiene la restricción en monitoreo tierra-aire debido al consumo energético, dado que es fundamental para este trabajo.

En la Tabla 4.3 se realiza comparación de las características que ofrecen los estándares inalámbricos: Wi-Fi, Bluetooth y ZigBee.

Tabla 4. 3 Estándares inalámbricos. [5]

	802.11g Wi-Fi WLAN	802.15.1 Bluetooth WPAN	802.15.4 ZigBee WPAN
Rango	100m	10-100m	>100m
Ancho de Banda	2-11 Mb/s	1 Mb/s	≤0.25 Mb/s
Consumo de energía	Medio	Bajo	Muy Bajo
Costo/complejidad	Elevado	Medio	Muy Bajo

Para elegir la tecnología inalámbrica más adecuada a los requerimientos del proyecto se comparan de acuerdo a las siguientes características: Ancho de Banda, distancia, topología de red y disponibilidad de potencia. Se analiza para los tres casos:

Bluetooth: Ofrece una buena tasa de transmisión de datos, y una distancia máxima de 100 metros en su versión de Clase 1, absorbiendo una mayor potencia lo que hace inoperable en medios donde el consumo energético es limitado.

Wi-Fi: Si la distancia a línea de vista necesaria es mayor a 100 metros es necesario recurrir a repetidores y aunque las distancias sean menores de 100 m, las fuentes de interferencia RF incluyendo árboles o edificios pueden reducir la distancia alcanzable.

Zigbee: En el caso del ancho de banda este se reduce al 10% en cuanto al Wi-Fi pero una ventaja al realizar esto es que se incrementa la distancia, obteniendo una máxima de 1000m de línea de vista y por otro lado el consumo de energía hace más viable el uso de esta tecnología para los fines de comunicación para el monitoreo, pues en un UAV la cantidad de energía consumida disminuye el radio de acción y por ende el bajo consumo energético es apremiado con distancia. Unas de las características notables de estos dispositivos es que tienen la capacidad de entrar en modo sleep consumiendo menos de 10uA y en modo de operación es menor a los 50mA.

La operación de estos en modo transparente simplifica bastante la transmisión recepción de tramas, aunque si se requiere monitorear más de un UAV se tiene que gestionar de una manera distinta. Una de las ventajas notables es que encripta los datos en la capa de red bajo el algoritmo AES-128 (FIPS 197) eliminando la posibilidad del robo de datos o en este caso la confusión que pudiera causar una fuente externa.

4.3 Comunicaciones Alámbricas

La comunicación entre los sensores y el microcontrolador típicamente se realiza por medio de dos interfaces UART y I2C. Estas proporcionan las reglas, parámetros de operación y los canales de comunicación para lograr una transmisión óptima en los datos transferidos e inclusive la detección de errores en su caso.

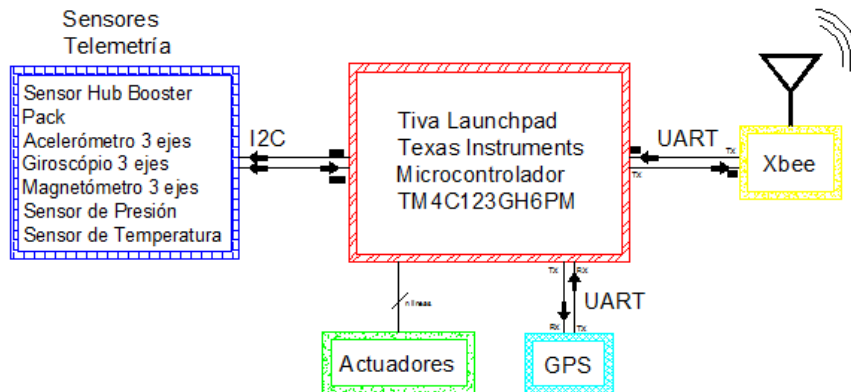


Figura 4. 12 Diagrama de bloques de la tarjeta FMS. [28]

4.3.1 Protocolo UART (Universal Asynchronous Receiver-Transmitter)

Para la operación de este no se requiere que el transmisor envíe una señal de reloj, en lugar de ello el emisor y el receptor deberán estar de acuerdo a los parámetros temporales y los bits especiales que se añadan a cada palabra. Cada extremo físicamente utiliza un par de líneas llamadas Tx y Rx, esta configuración solo tiene la capacidad de transmitir o recibir datos por lo que se realiza una comunicación tipo half-dúplex.

La cantidad total de bits transmitidos por segundo es llamada velocidad de transmisión y a la división de esta por el tamaño de palabra enviada se define como ancho de banda en bytes por segundo.

La trama que es la unidad más pequeña de transmisión serial, contiene un bit de inicio (cero lógico), que cambia el estado de la línea y que activa la lectura del dato enviado en el lado receptor, posteriormente se envía una palabra (comúnmente 8 bits y esto depende directamente del microcontrolador que se use) y al final un bit de paro (uno lógico) que indica se finalizó la transmisión y el receptor queda en espera de la siguiente palabra bajo las mismas condiciones. Los parámetros a configurar son la velocidad de transmisión en bps, el tamaño de la palabra, bit de paro y el bit de paridad (fig. 4.13).

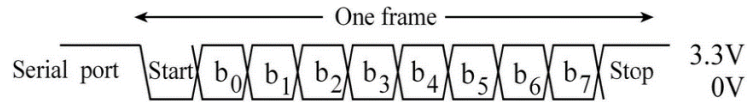


Figura 4.13 Trama de datos seriales con 8 bits, 1 bit inicio, 1 bit de paro y sin bit de paridad. [29]

La comunicación serial del microcontrolador TM4C123GH6PM de Texas Instruments se logra por medio de ocho puertos configurables, los cuales nos permiten modificar velocidades de comunicación, tamaño de buffer, tamaño de trama, etc.

Para realizar la transmisión-recepción de datos se ocupa de un registro que guarda los datos para transmitirlos hacia un buffer FIFO (First input First output) esperando la indicación de una bandera que indica cuando está en uso y cuando puede recibir datos nuevos, que particularmente para este caso son TXEF y TXF. Asimismo en el lado receptor se encuentra otra bandera llamada R la cual se activa cuando se hayan recibido la cantidad de bits programados y sin errores (figuras 4.14 y 4.15).

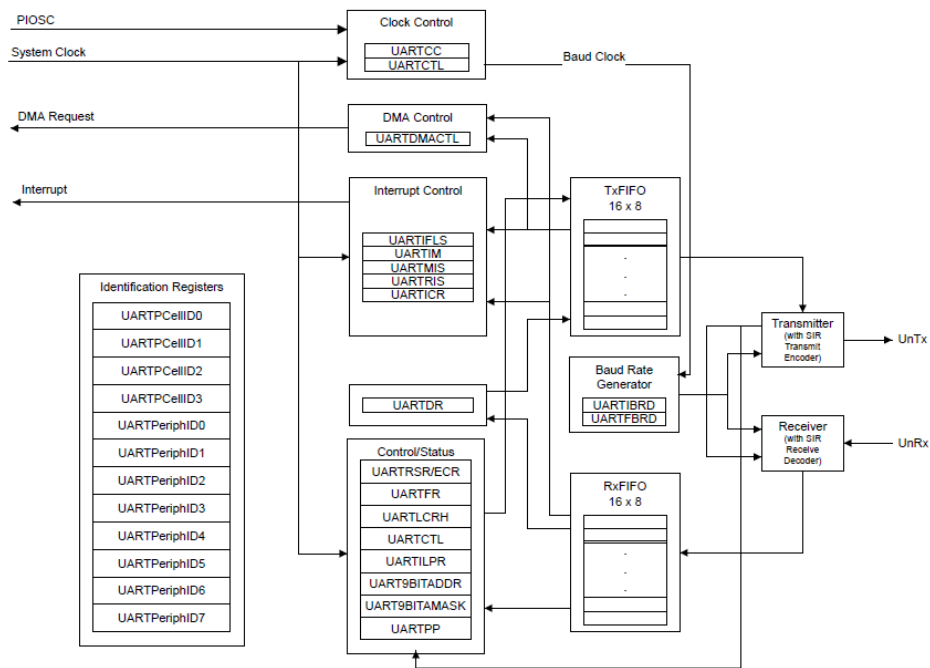


Figura 4.14 Diagrama de bloques del módulo UART del TM4C123GH6PM. [30]

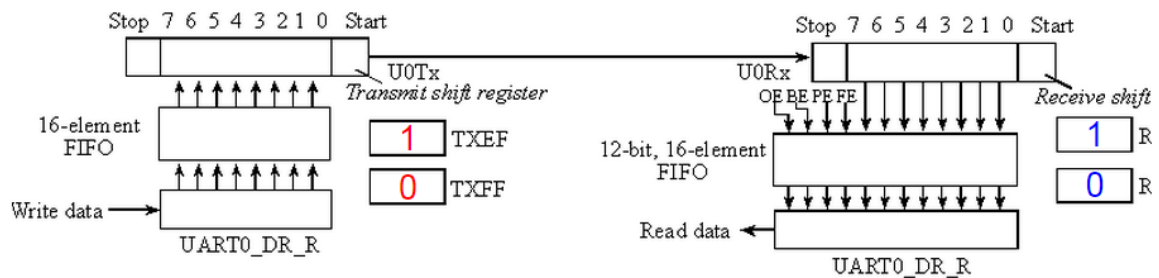


Figura 4.15 Transmisión serial asíncrona UART. [31]

4.3.2 Interfaz I2C (Inter Integrated Communication)

Este protocolo es utilizado para la conexión de periféricos en el microcontrolador a cortas distancias proporcionando una comunicación half dúplex. Proporciona una transferencia de datos bidireccional por medio de dos cables SDA (Serial Data Line) y SCL (Serial Clock Line), lo que permite establecer una interface con otros dispositivos tales como memorias RAM y ROM, LCDs y una variedad de sensores.

Los dispositivos en el bus I2C se pueden designar como maestro o esclavo, en algunos casos dependiendo del microcontrolador que se utilice se puede utilizar un maestro para obtener información de varios esclavos. Las velocidades en las que el protocolo opera el TM4C123GH6PM se muestran en la tabla 4.4. Al seleccionar la velocidad se debe tener en cuenta que al ser mayor se exige un consumo mayor de energía.

Tabla 4. 4 Velocidades de transmisión I2C en el TM4C123GH6PM. [6]

Tipo	Velocidad
Standard	100 Kbps
Fast-mode	400 Kbps
Fast-mode plus	1 Mbps
High-speed mode	3.33 Mbps

Para entender este protocolo se definen cuatro modos de operación para los dispositivos conectados a la red:

- Maestro transmisor Emisor:
- Maestro receptor:
- Esclavo transmisor
- Esclavo receptor

La figura 4.16 muestra el diagrama de bloques de un sistema de comunicación basado en la interface I²C, la red puede contener múltiples maestros y múltiples esclavos. Las líneas SCL y SDA son ambas bidireccionales.

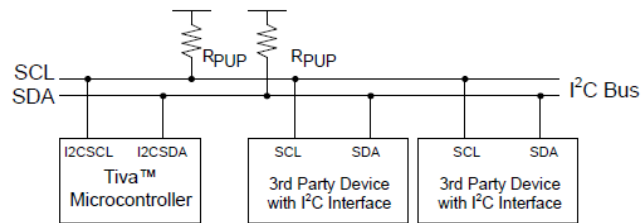


Figura 4.16 Diagrama de bloques de una red de comunicación I²C en el TM4C123GH6PM. [32]

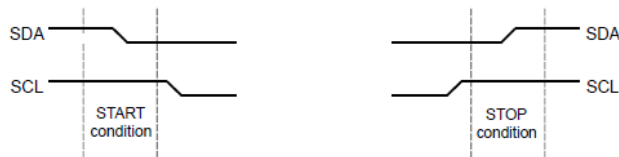


Figura 4.17 Condiciones de inicio y paro de comunicación I²C. [33]

El frame para esta comunicación se compone de la siguiente manera:

1. El bus está en modo de espera cuando SCL y SCA se encuentran en un nivel lógico alto.

2. Después de que se realiza la petición de la condición de inicio mostrada en la figura 4.16 el maestro envía la dirección de 7 bits de un esclavo en particular conectado al bus para que este responda.
3. Un bit (R/S) de un registro específico, se encarga de indicar el estado si se encuentra transmitiendo o recibiendo datos.
4. Un bit llamado ACK es enviado por el esclavo solicitado para responder que está listo para enviar los datos.
5. Se envía un dato de un byte del esclavo al maestro
6. Dependiendo de la cantidad de bytes que el esclavo requiera enviar se genera un ciclo entre los puntos 4 y 5 hasta que el maestro termine la transmisión con la condición de paro (fig. 4.18).

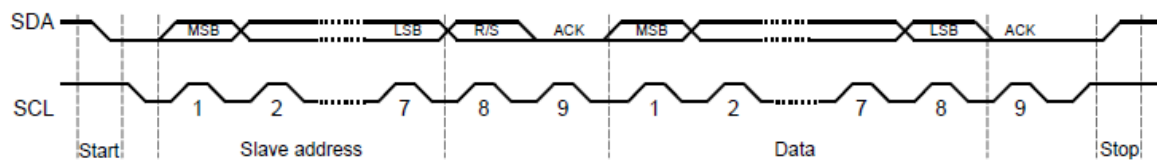


Figura 4.18 Trama de datos I2C. [34]

Capítulo 5 Desarrollo del sistema de piloto automático abordo de un UAV

Actualmente se requiere resolver problemas de monitoreo y sensado a distancia, con las tecnologías actuales es posible hacerlo mediante el uso conjunto de UAV's y sistemas de abordo tanto de comunicaciones como de control y telemetría integrados al modelo.

Esto se logra por medio de instrumentación electrónica para maximizar el espacio y el peso. La implementación de equipamiento a menudo excede la aeronave y el éxito de la misión depende del buen funcionamiento de todos los sistemas de control y sensores.

Actualmente se encuentran en el mercado tarjetas de desarrollo tales como la ArduPilotMega o inclusive Arduino, en las cuales el usuario tiene acceso a segmentos de código abierto sin tener una posibilidad de adaptabilidad hacia una misión particular dado que el código no es propio y es difícil en algunos casos comprenderlo o inclusive saber si es operacional.

Este tipo de tarjetas son más orientadas hacia las actividades de recreación y en algunos casos de comprensión pues suelen requerir de bloques tales como una estación terrena que difícilmente puede ser escalable a necesidades particulares y en algunos casos cuenta con pocas opciones de modelos lo cual limita la operabilidad.

En este capítulo se muestran los diferentes instrumentos electrónicos utilizados para la obtención de los datos del sistema de navegación del UAV: altura, actitud, velocidad y posición que se envían inalámbricamente con un protocolo creado por Zigbee Alliance a la estación terrena. Así como la integración de ellos para lograr implementar un piloto automático.

5.1 Integración de elementos electrónicos

Para completar la configuración electrónica es necesario definir la arquitectura de la computadora de vuelo, así como el sistema de comunicaciones entre tierra y el UAV. Esto permitirá el uso eficiente de espacio y peso.

En la Fig. 5.1 se muestra el diagrama a bloques de la estructura propuesta para integrar la circuitería y el procesamiento de los datos que se enviarán a la estación terrena.

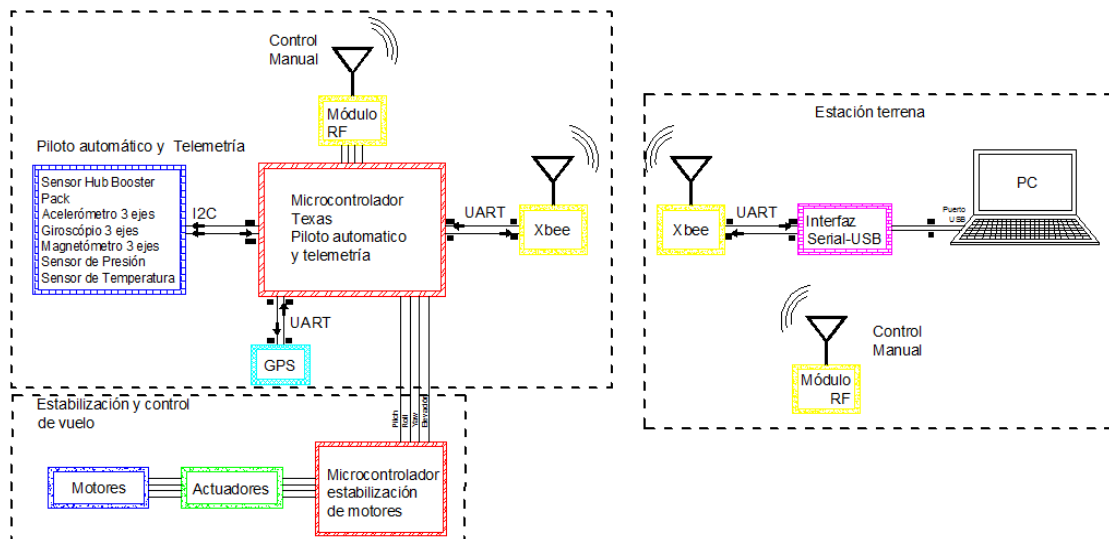


Figura 5.1 Diagrama de bloques del sistema de abordaje propuesto. [35]

Para la implementación de telemetría se ha optado por utilizar un launchpad y una tarjeta que contiene varios sensores fabricadas por Texas Instruments, las cuales se encuentran diseñadas para acoplarse entre ellas optimizando así el espacio utilizado.

La interconexión con otros dispositivos externos es una característica que define a la estructura de montaje del microcontrolador en la tarjeta, puesto que se requieren lecturas de datos proporcionados por elementos externos a esta y que particularmente realizan una comunicación bajo las características de comunicación por medio de un bus I2C o UART.

Para la etapa de estabilización y control de vuelo se utilizó la tarjeta KK2.0 proporcionada por el modelo SK450 de la marca Turnigy. (Ver anexo F)

El microcontrolador seleccionado es el TM4C123GH6PMI de Texas Instruments, el cual cuenta con las siguientes características:

- Procesador ARM Cortex-M4F de 32 bits
- Rendimiento 100DMIPS operando a 80Mhz
- 256KB en memoria Flash
- 32KB RAM
- 2-KB EEPROM
- Periféricos: I2C y UART
- Modo de hibernación en bajo consumo
- 43 Pines de puertos E/S
- Arquitectura Harvard

Se eligió este dispositivo por las prestaciones que ofrece en cuanto a la velocidad de procesamiento, la resolución en cuanto a la cantidad de bits que maneja y la arquitectura ARM pues esta está orientada hacia la máxima eficiencia energética, pues es de gran relevancia obtener datos lo más rápido posible y por la versatilidad de ser un launchpad (fig. 5.2), lo que permite poder complementarse con tarjetas de sensores maximizando así el espacio ocupado y cumpliendo con las características de modularidad necesarias.

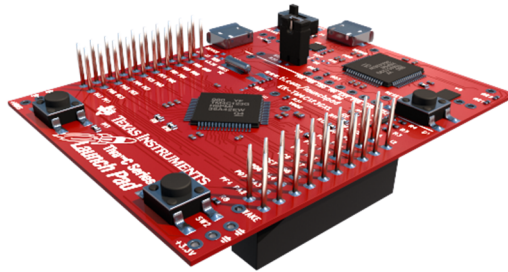


Figura 5.2 Launchpad TIVA de Texas Instruments. [36]

5.2 Sensores

Un sensor es un dispositivo capaz de detectar una magnitud física y transformarla a una señal eléctrica en una proporción lineal y en algunos casos cuenta con un convertor Analógico-Digital para su procesamiento posterior.

Para la medición de las distintas variables se utilizó la tarjeta Sensor Hub Booster pack del fabricante Texas Instruments® la cual cuenta con los siguientes sensores:

- Sensor de movimiento de 9-ejes Marca InvenSense® MPU9150:
 - Acelerómetro en tres ejes
 - Giroscopio en tres ejes
 - Magnetómetro en tres ejes
- Sensor de presión Bosch® BMP180
- Sensor de humedad SHT21
- Sensor de luz ISL29023

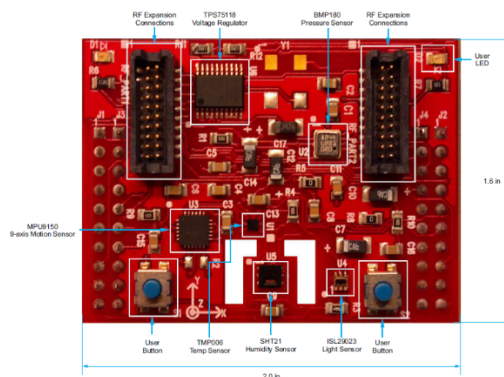


Figura 5.3 Tarjeta IMU Boosterpack TI. [37]

Esta integra en un mismo bus la comunicación I2C de todos los sensores minimizando la estructura de la circuitería (fig. 5.4).

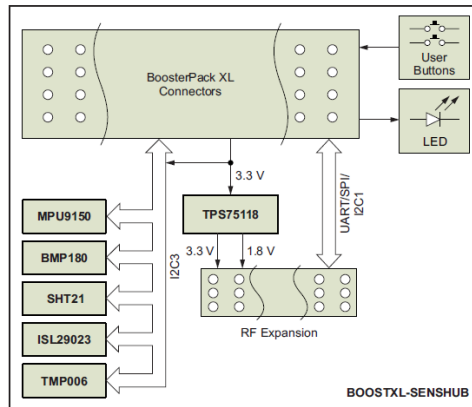


Figura 5. 4 Diagrama de bloques tarjeta Booster Pack. [38]

5.2.1 Indicador de horizonte artificial

El circuito utilizado para realizar la medición de la actitud del UAV respecto a el horizonte es el MPU9150 de la marca InvenSense®, por medio de este se obtienen datos de acelerómetros, giroscopios y magnetómetros en tres ejes.

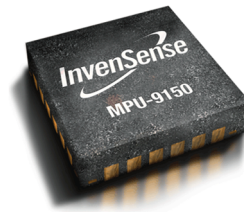


Figura 5.5 Chip MPU-9150. [39]

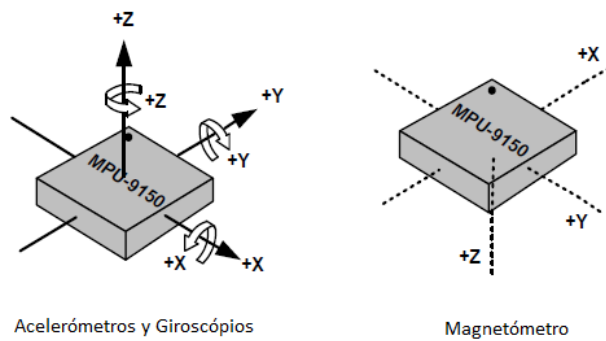


Figura 5.6 Orientación de ejes en el dispositivo. Invensense. [40]

El MPU 9150 es el primer dispositivo que integra el seguimiento de movimiento de 9 ejes basado en MEMS que combina un giroscopio, un acelerómetro y un magnetómetro de 3 ejes cada uno junto con un procesador digital de movimiento de tercera generación (DMP Digital Motion Processor). Este cuenta con tres convertidores ADC, estos interpretan el valor del movimiento en tres ejes a valores digitales, los cuales cuentan con distintas resoluciones programables. En la tabla 5.1 se muestran los valores de medición para la conversión analógico-digital de cada convertidor.

La velocidad de comunicación con los registros del bus I2C es de 400Khz además de contar con filtros digitales programables para los acelerómetros y giroscopios y un buffer FIFO de 1024 bytes.

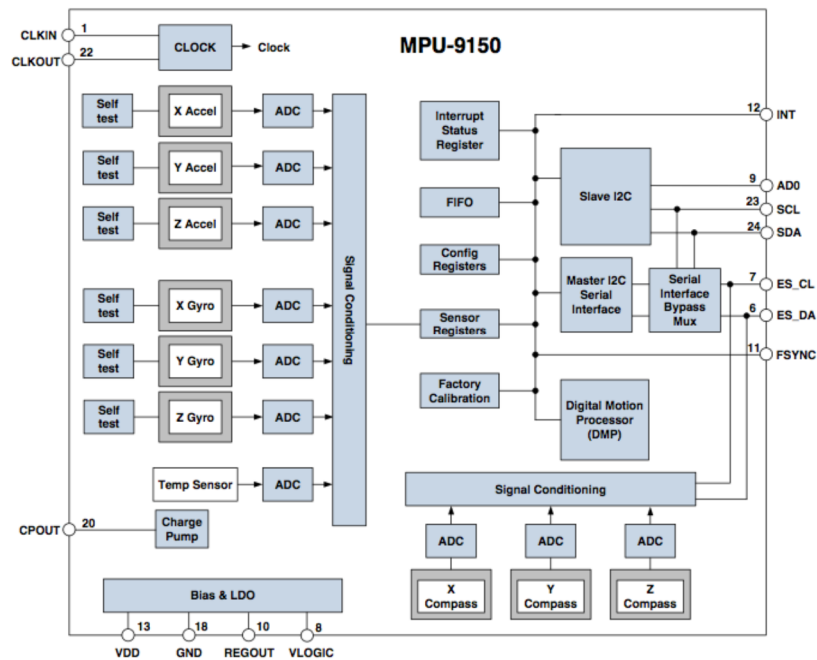


Figura 5. 7 Diagrama de bloques del MPU 9150. [41]

Tabla 5. 1 Resolución de mediciones del MPU 9150. [7]

Dispositivo	ADC	Límites de medición
Acelerómetro	16 bits	$\pm 2, \pm 4, \pm 8, \pm 16g$
Giroscopio	16 bits	$\pm 250, \pm 500, \pm 1000, \pm 2000$ °/s
Magnetómetro	13 bits	$\pm 1200\mu T$

Una vez muestreadas estas mediciones se integran por medio de un algoritmo llamado filtro complementario (ver anexo B), dando como resultado una matriz de rotación de tres dimensiones (DCM). De lo anterior se deriva una medición más precisa y con una tolerancia mayor a las perturbaciones externas disminuyendo así las variaciones no deseadas en las mediciones.

5.2.2 Sensor de altitud y de temperatura

La altitud se puede obtener a partir de los cambios de presión debido a las propiedades físicas de la atmósfera. Estas diferencias de presión pueden ser medidas fácilmente y con una buena resolución por medio del circuito BMP180, el cual se encuentra compensado en presión y temperatura para una lectura más precisa.



Figura 5.8 Sensor de temperatura y presión BOSCH® BMP180. [42]

Este es conformado por un sensor piezo-resistivo que modifica su resistividad en presencia de cambios de presión, un convertidor analógico digital, una unidad de control con una memoria E2PROM y una interfaz I2C. Los valores de Presión y temperatura se obtienen de 16 a 19 bits y 16 bits respectivamente.

El cálculo de la altura se obtiene por medio de la siguiente expresión proporcionada por el fabricante:

$$h = 44330 * \left[1 - \left(\frac{P}{P_0} \right)^{\frac{1}{5.255}} \right] \quad (1)$$

Donde

h= Altura barométrica

Po=Presión atmosférica en el punto de despegue

P=Presión atmosférica en cualquier punto de vuelo

El microcontrolador envía la secuencia de inicio para la medición de presión y de temperatura. Para hacer una lectura en grados centígrados para el caso de la temperatura y hectoPascales para la presión es necesario recurrir a datos de calibración guardados en la memoria. Después de un tiempo de conversión la muestra se transmite al microcontrolador por medio de una solicitud a la interfaz I2C (fig. 5.9).

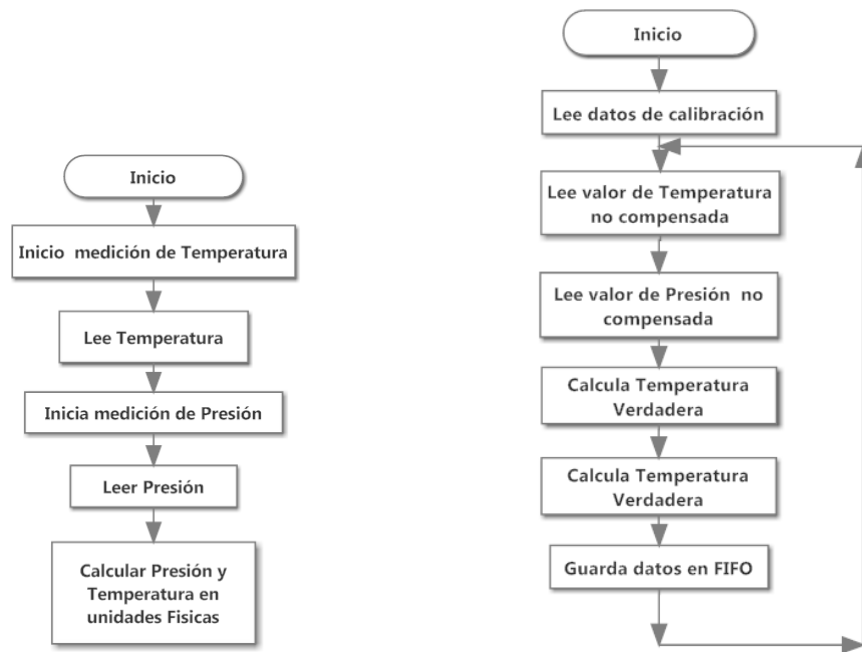


Figura 5. 9 Diagramas de flujo de obtención de datos y mediciones compensadas. [43]

5.2.3 GPS

Para la obtención de datos en esta etapa se utilizó el circuito GPS GTPA010, el cual tiene una buena precisión en cuanto a la posición y velocidad.

Algunas de las características más notables son las siguientes;

- Alta sensibilidad, arriba de -165dBm
- Bajo consumo de potencia 48mA para adquisición y 37mA para el rastreo.
- Interfaz de comunicación UART
- Voltaje de operación 3.3V
- Exactitud de posición 3m



Figura 5. 10 Circuito GPS. [44]

Este envía varias cadenas de datos por medio de un puerto UART, el cual permite la configuración de diferentes velocidades de transmisión que son programables las cuales van desde los 1200bps hasta los 230,400bps y un muestreo de datos de hasta 10 Hz.

La configuración de los datos está basada en el estándar NMEA0183 (National Marine Electronics Association). El cual define requerimientos en las señales eléctricas, protocolos de transmisión de datos y tiempo y define la impresión de datos en código ASCII. [1]

Este estándar contiene varios enunciados con información para diferentes usos. Enseguida se describe el enunciado de interés y tiene la siguiente estructura:

```
$GPRMC,064951.000,A,2307.1256,N,12016.4438,E,0.03,165.48,260406,,,A*55
```

La interpretación de los datos se define por medio de la tabla 5.2.

Tabla 5. 2 Descripción del enunciado GPRMC. [8]

Nombre	Ejemplo	Unidades	Descripción
ID del mensaje	\$GPRMC		Encabezado del protocolo RMC
Hora UTC	64951.000		hhmmss.sss
Estado	A		A=dato valido o V=dato no valido
Latitud	2307.1256		ddmm.mmmm
Indicador N/S	N		N=norte o S=Sur
Longitud	12016.4438		dddmm.mmmm
Indicador E/W	E		E=este o W=oeste
Velocidad sobre tierra	0.03	Nudos	
Curso sobre tierra	165.48	Grados	VERDADERO
Fecha	260406		Ddmmaa
Variación Magnética		Grados	Se requiere de personalización con el fabricante
Modo	A		A=modo autónomo D= modo diferencial E=modo de estimación
Checksum	*65		
<CR><LF>			Terminación del mensaje

La lectura de datos se realiza por medio del siguiente diagrama de flujos. (fig. 5.11)

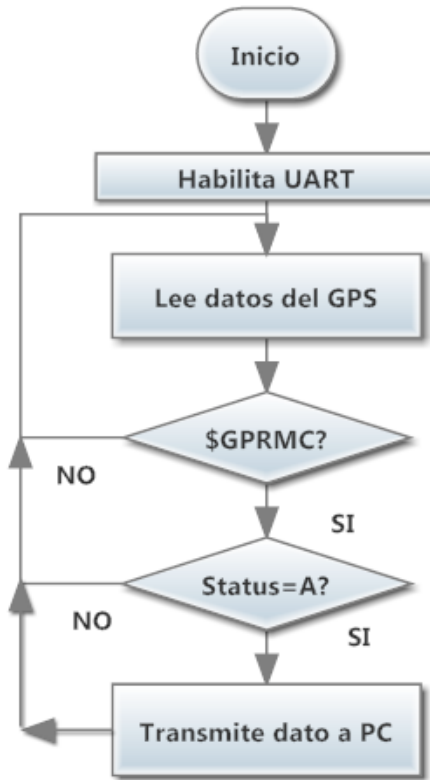


Figura 5. 11 Diagrama de flujo para el GPS. [45]

5.3 Tarjeta de telemetría para el control y el monitoreo del UAV

La integración de los elementos antes descritos conforman el sistema de abordaje encargado de mostrar la trayectoria establecida y enviar los datos necesarios a la GCS para su análisis y monitoreo.

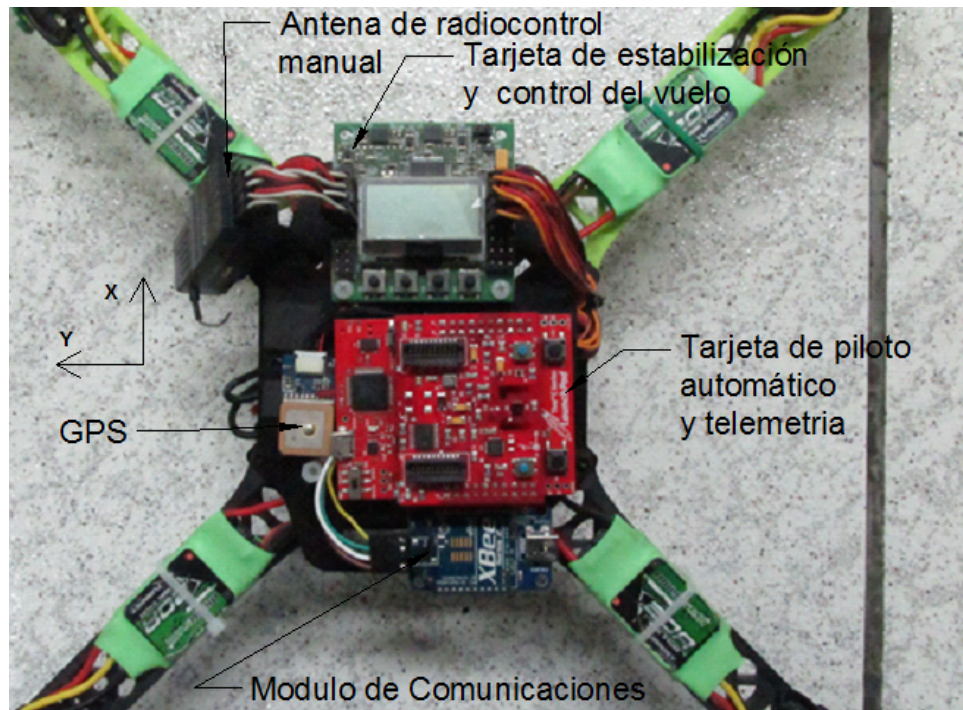


Figura 5. 12 Ubicación de elementos electrónicos a bordo del UAV. [46]

5.3.1 Control automático de navegación

En esta etapa se propone un controlador proporcional-derivativo, basado en las ecuaciones cinemáticas de movimiento rectilíneo se fija un punto coordinado el cual contiene las distancias a recorrer en los ejes x-y, y por medio de las lecturas de aceleraciones se emite una señal de control para realizar movimientos de roll-pitch sobre el plano X-Y.

Una manera de poder controlar la trayectoria del UAV es basándose en el uso de la navegación inercial y un controlador PD, el cual contempla el cálculo del error entre la posición deseada y la posición actual, esta última obtenida a partir de estimaciones por medio de la medición de las aceleraciones y las velocidades en los ejes y la ubicación inicial.

La manera básica de saber el movimiento de una partícula es por medio de ecuaciones de cinemática, las cuales involucran las aceleraciones y velocidades.

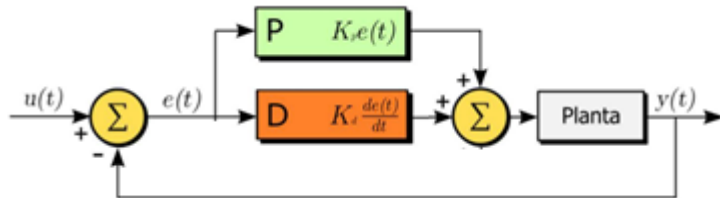


Figura 5. 13 Diagrama de bloques del control PD. [47]

En seguida se muestra el modelo matemático del controlador PD propuesto:

$$u_t = Kp([S_{des}] - [S_{t-1}]) + Kd(\dot{S}_{des} - \dot{S}_{t-1})$$

Siendo u_t la corrección que se debe realizar para lograr eliminar el error y llegar al punto deseado.

S_{des} = posición deseada;

S_{t-1} = posición actual;

Kp = Ganancia proporcional

Kd = Ganancia derivativa

\dot{S}_{des} = Velocidad deseada;

\dot{S}_{t-1} = Velocidad actual;

Expandiendo la ecuación anterior al plano X-Y queda como:

$$u_{t_x} = K_{p_x} * (S_{des_x} - V_x * \Delta t) + K_d * \left(\frac{S_{des_x}}{\Delta t} - V_x \right)$$

$$u_{t_y} = K_{p_y} * (S_{des_y} - V_y * \Delta t) + K_d * \left(\frac{S_{des_y}}{\Delta t} - V_y \right)$$

Donde:

$$V_x = (\text{lectura de acelerómetro en } x) * \Delta t$$

$$V_y = (\text{lectura de acelerómetro en } y) * \Delta t$$

Δt = tiempo transcurrido desde la última muestra

El periodo entre muestras debe ser muy pequeño para evitar acumular errores significativos que impacten al tiempo de retraso de la señal de control.

CAPITULO 6. Implementación de la estación terrena GCS

En este capítulo se presenta la configuración de los módulos Zigbee® utilizada para la transmisión-recepción de datos entre el UAV y la estación terrena, así como las características de esta última para lograr la interpretación y visualización grafica de los datos en la PC. En la figura 6.1 se muestra en bloques la estructura seleccionada para lograr los objetivos antes mencionados.

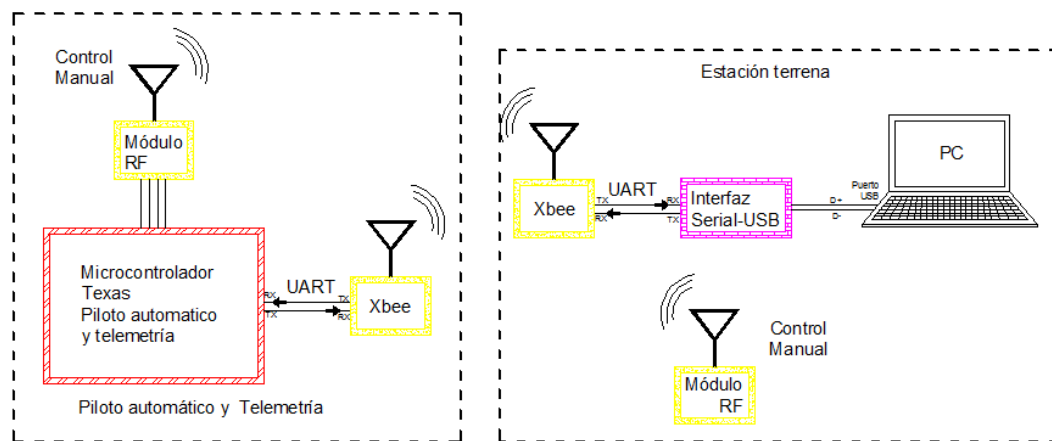


Figura 6. 1 Bloques de comunicaciones entre el UAV y la estación Terrena. [48]

6.1 Configuración de la red punto a punto con dispositivos ZigBee®

En todo momento es necesario saber la ubicación y el comportamiento del UAV, para lograr transmitir los datos a distancia se implementa una red de dos puntos por medio de tecnología Zigbee™ en la cual se tienen dos módulos, uno a cada extremo de la red, por medio de las cuales se puede realizar una comunicación half dúplex. En el lado de la estación terrena hay un elemento llamado coordinador y por el otro lado se encuentra el dispositivo final.

Para los fines de este proyecto se eligió el dispositivo XB24-B Series 2, esto por las características de bajo consumo energético en las modalidades de transmisión y espera, así como la distancia cubierta.



Figura 6. 2 Modulo XB24-B Series 2. [49]

Las características principales de este módulo son:

- Distancias en interiores de hasta 40m y en exterior en línea de vista hasta 120m.
- 15mA de consumo en modo de espera 40 mA para la transmisión y 40mA para la recepción.
- 2mW de potencia de transmisión.
- Alimentación de 2.1 a 3.6 V.
- Encriptación de 128 bits.
- Tasa de bits transmitidos de hasta 250Kbps.

La transmisión de datos vía radio se efectúa por medio de antenas Xbee®, las cuales deben programarse y configurarse por medio del software gratuito que dispone el fabricante llamado X-CTU.

Para realizar la configuración es necesario contar con un dispositivo electrónico serial-USB que realice la conversión entre ambos formatos en ambos sentidos, para esto se utilizó un adaptador comercial llamado Funduino el cual utiliza el chip FT232 de la marca FTDI como el mostrado en la figura 6.3 dado que contiene una base diseñada para alojar módulos Zigbee®, una vez conectado a la computadora debe reconocer la conexión como un puerto serial virtual asignado.

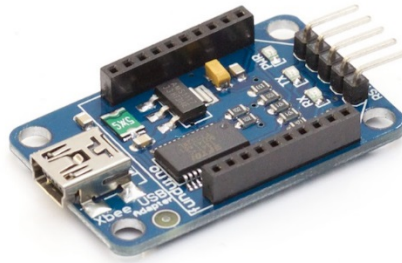


Figura 6. 3 Adaptador Xbee USB. [50]

Una vez abierto el programa hay que configurar los parámetros de la conexión serie, posteriormente presionar la opción Test/Query para entablar comunicación con el módulo como se muestra en la figura 6.4.

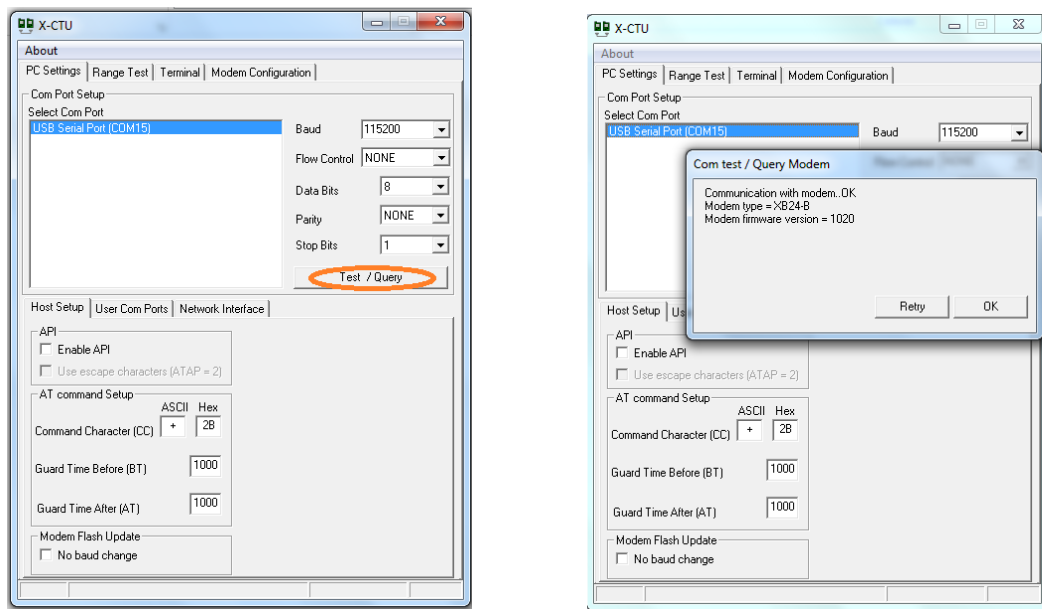


Figura 6. 4 Programa X-CTU detectando un módulo. [51]

Cuando aparece la ventana que nos indica el modelo del módulo conectado indica que está establecida la comunicación y se encuentra listo para su programación. Lo que se procede a modificar es el firmware del dispositivo, este se selecciona de una lista. En la pestaña Function Set seleccionar y programar para un módulo como coordinador AT y para el otro como end device AT como

se muestra en las figuras 6.5 y 6.6 se pulsa la pestaña “Write” y posteriormente se cambian los valores de PAN ID y valores de configuración de comunicación serie y se vuelve a pulsar “Write”. En ambos casos los valores anteriores tienen que ser los mismos.

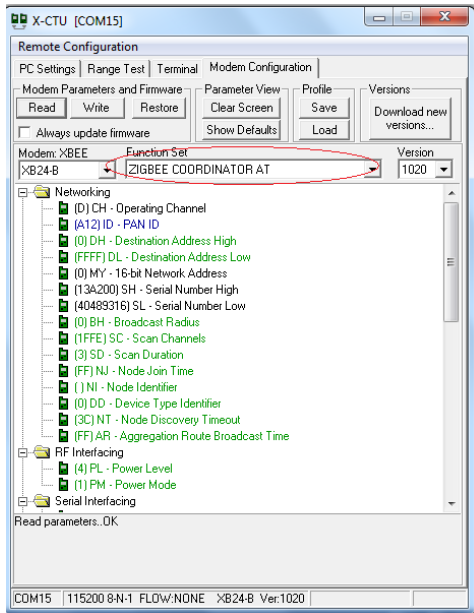


Figura 6. 5 Dispositivo Coordinador. [52]

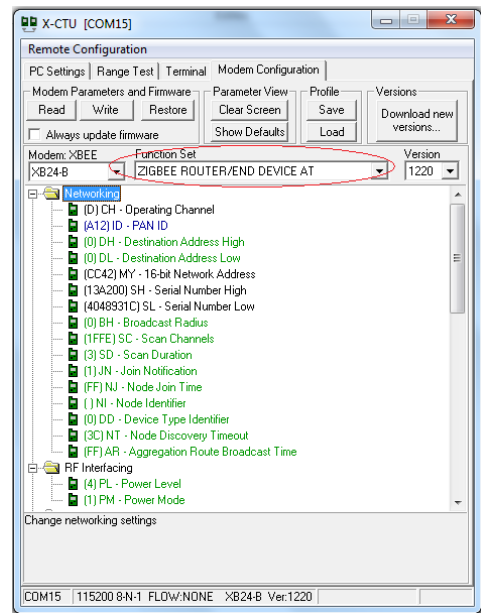


Figura 6. 6 Dispositivo final. [53]

Estos módulos se tienen dos modos de programarse, el modo transparente es elegido por ser el de menor complejidad que cumple con los requerimientos de comunicación, dado que se está utilizando una red punto a punto. Este modo permite actuar a los módulos como un cable, es decir todo lo que se transmita en un lado, será recibido por el otro lado sin distinguir codificaciones. La única restricción en esta configuración es que la muestra de datos enviada no sea mayor a la velocidad de transmisión pues se perderían datos en el proceso.

Estos se comunican con el microprocesador por medio de la interfaz UART como lo indica la figura 6.7.

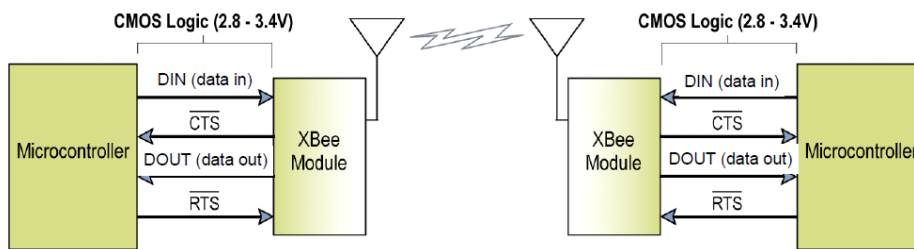


Figura 6. 7 Conexión básica entre dos módulos Zigbee®. [54]

6.2 Especificaciones del software de la estación terrena GCS

Esta será capaz de obtener datos de los instrumentos de abordo a una distancia máxima de 100m a línea de vista para monitorear las variables de vuelo, así como la ubicación por medio del GPS.

Para fines de este proyecto la estación se limita a las siguientes características, con las posibilidades de expandir en un futuro a la visualización de una cantidad mayor de variables o incluso instrumentación exclusiva de alguna misión en particular.

Los requerimientos mínimos necesarios de la GCS para el despliegue gráfico de los datos son:

- Enviar y recibir datos desde una PC por medio de una interfaz serial-USB conectada al módulo coordinador Zigbee®.
- Analizar una cadena de datos y separarla para su uso en los distintos instrumentos.
- Despliegue gráfico de instrumentos de vuelo por medio de una consola elaborada en Labview, la cual contendrá datos como latitud, longitud, altura, actitud del modelo en vuelo.
- Representación de la posición global por medio de un mapa satelital

6.3 Desarrollo de software y despliegue de datos en Labview

Labview es una de programación grafica de propósito general con extensas librerías de funciones diseñadas específicamente para realizar adquisición de datos, análisis de señales, control de instrumentos con interfaces paralelo, serie y USB, presentación de información y almacenamiento de datos.

El ambiente de desarrollo de este programa se basa en el lenguaje gráfico G, cuya principal característica es que la programación se realiza por medio de diagramas a bloques eliminando detalles de sintaxis.

El proceso de recepción de datos se realiza a través del puerto USB de la computadora por medio del circuito mostrado en la figura 6.8, esto permite recibir datos del exterior sin la necesidad de tarjetas de adquisición de datos.

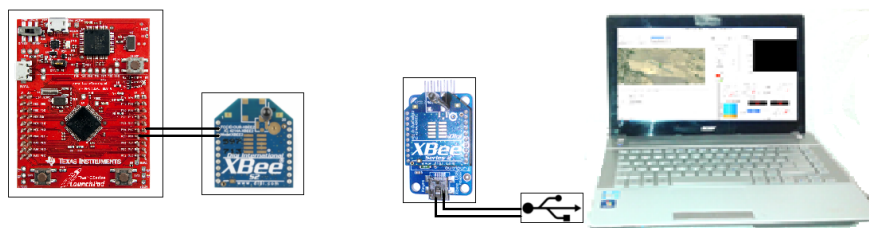


Figura 6. 8 Interfaz de comunicación entre la FMS y la PC. [55]

Una vez recibidos los datos en el módulo Zigbee® Coordinador, estos se pueden canalizar al programa Labview por medio de un puerto serie virtual, el cual en su totalidad es configurable en el mismo programa por medio de una herramienta llamada NI Visa así como la inicialización del mismo.

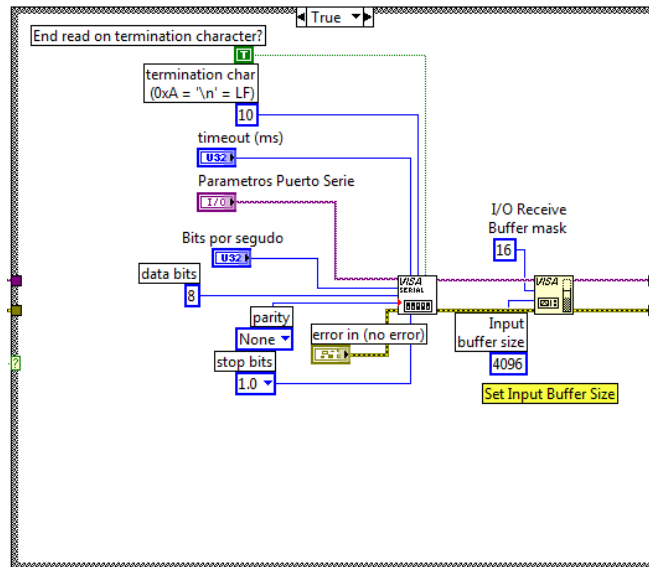


Figura 6. 9 Configuración e inicialización del puerto serial por medio de LabView. [56]

Una vez establecida la comunicación serial es necesario separar la totalidad de los datos para poder representarlos en sus instrumentos correspondientes, la siguiente figura muestra el formato de los datos recibidos en el puerto serie, a partir de ellos se inicia la separación e interpretación gráfica.

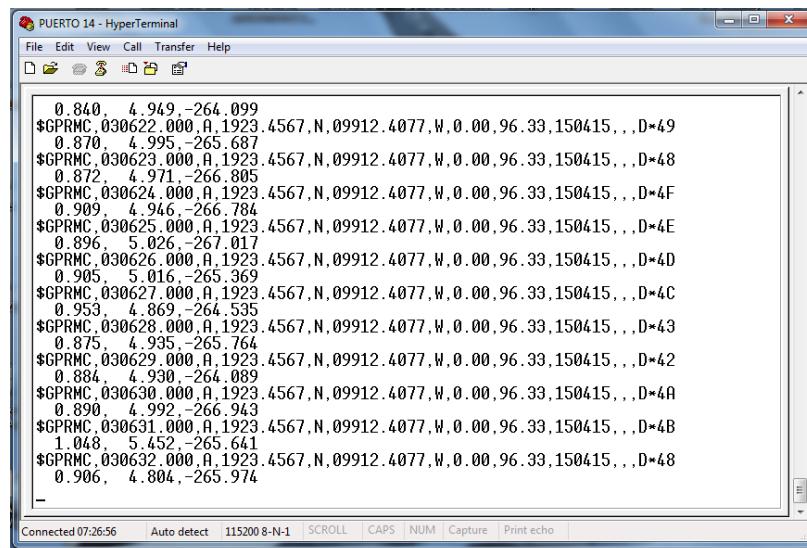


Figura 6. 10 Sentencia de datos NMEA y ángulos de Euler. [57]

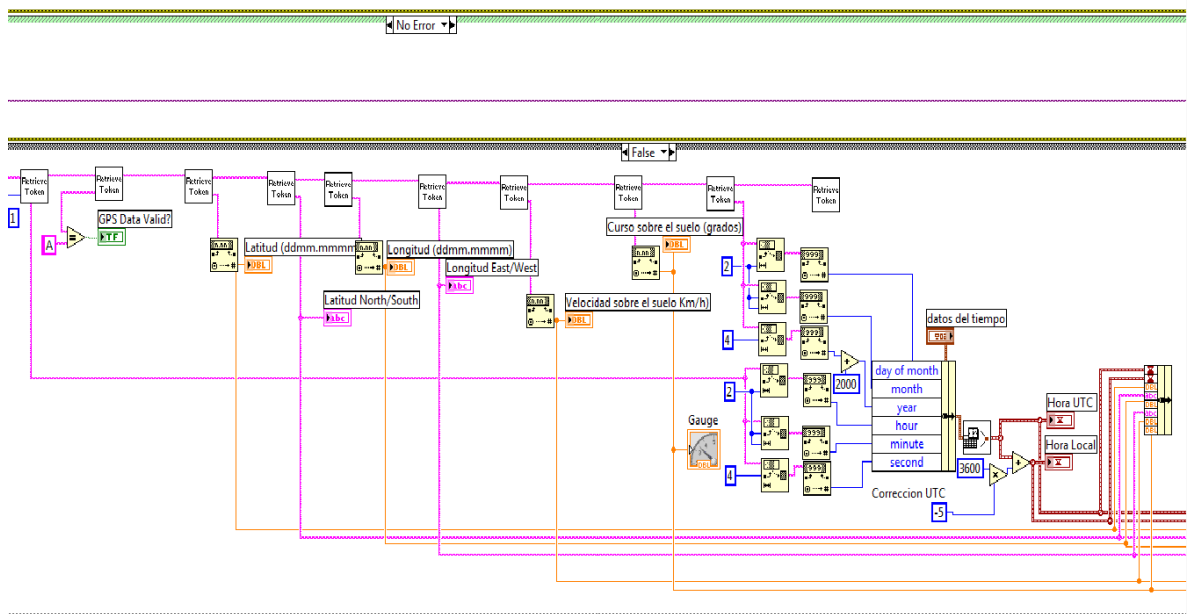


Figura 6. 11 Software de separación de datos GPS en LabView. [58]

```

cOM 14 - HyperTerminal
File Edit View Call Transfer Help
mx 26.100, my 11.100, mz 41.700
ey 0.076, ex 1.673, ez-334.304
$GPRMC,171840.000,A,1923.4561,N,09912.4085,W,0.00,240.04,150415,,D*71
T= 29.218 C, P= 24147.744 Pa, H= 10587.721 msnm
ax 0.044, ay 0.365, az 9.370
gx 0.027, gy 0.043, gz 0.030
mx 26.100, my 11.700, mz 41.700
ey 0.135, ex 1.657, ez-334.155
$GPRMC,171841.000,A,1923.4561,N,09912.4085,W,0.00,240.04,150415,,D*70
T= 29.206 C, P= 24146.584 Pa, H= 10588.030 msnm
ax 0.005, ay 0.320, az 9.458
gx 0.085, gy 0.036, gz 0.034
mx 25.200, my 10.800, mz 40.200
ey 0.151, ex 1.750, ez-335.117
$GPRMC,171842.000,A,1923.4561,N,09912.4085,W,0.00,240.04,150415,,D*73
T= 29.230 C, P= 24143.126 Pa, H= 10588.949 msnm
ax 0.003, ay 0.317, az 9.452
gx 0.012, gy 0.038, gz 0.030
mx 24.000, my 10.800, mz 39.000
ey 0.138, ex 2.226, ez-333.800
-
Connected 00:17:29 Auto detect 115200 8-N-1 SCROLL CAPS NUM Capture Print echo

```

Figura 6. 12 Datos recibidos de la tarjeta de abordo [59]

En las figuras 6.9 y 6.11 se muestra la programación gráfica en Labview para obtener y separar la cadena de datos proporcionada por el GPS.

Una vez terminada la programación se realizó la interfaz gráfica que permite visualizar en instrumentos virtuales el valor de los sensores así como ubicar el aeromodelo mediante un mapa satelital.

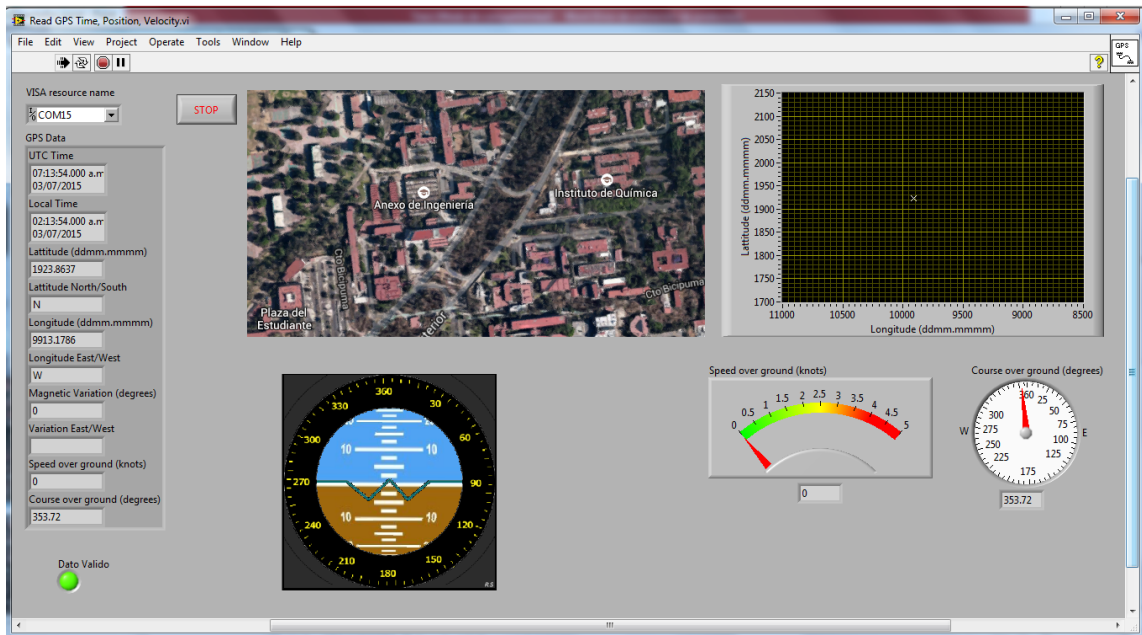


Figura 6. 13 Interfaz gráfica para la estación terrena [60]

CAPITULO 7. Realización de pruebas y adquisición de datos

Basado en los objetivos planteados en el capítulo 1 se desarrollaron las siguientes pruebas:

1. Verificación de la operatividad de transmisión-recepción de la red implementada con módulos de tecnología Zigbee
2. Verificación de obtención y envío de datos entre la tarjeta de telemetría, el piloto automático y la estación terrena.
3. Implementación de una interfaz gráfica en la PC por medio del programa LabView para la visualización del comportamiento y ubicación del UAV.
4. Implementación de un algoritmo de control en la tarjeta de abordo para lograr el vuelo autónomo de acuerdo a la navegación inercial.
5. Prueba del algoritmo de control en vuelo en el UAV Turnigy SK450

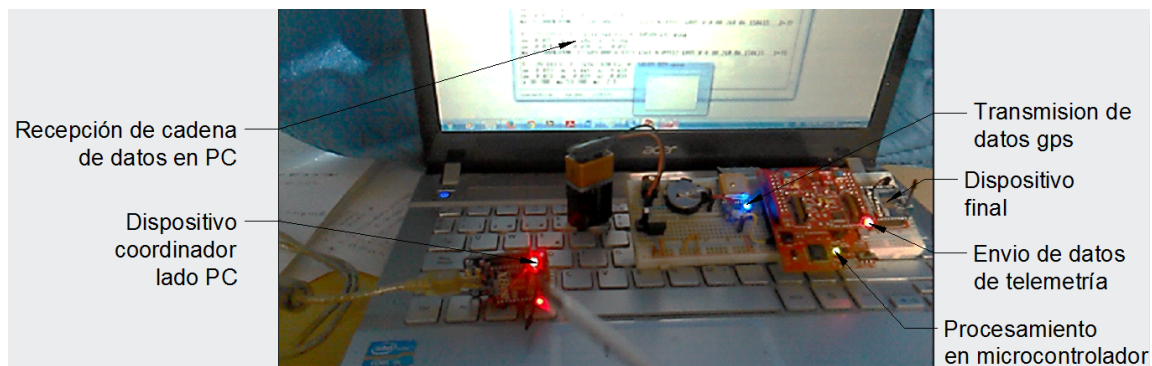


Figura 7. 1 Implementación y pruebas de transmisión-recepción de datos. [61]

La figura 7.1 muestra el funcionamiento de las etapas de adquisición de datos, ordenamiento en el microcontrolador y transmisión por medio de antenas Xbee de tecnología Zigbee. En la pantalla de la pc se muestra la cadena de datos obtenidos en un formato ASCII.

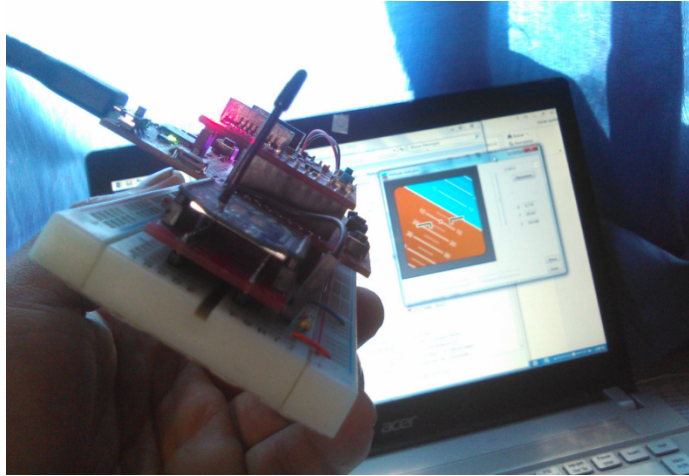


Figura 7. 2. Prueba de indicador de horizonte. [62]

En la figura 7.2 se muestra la operación del indicador de horizonte artificial de manera independiente a los demás instrumentos verificando sus niveles de operación y funcionalidad siendo estos óptimos para su operación.

Se verifico el funcionamiento de la estación terrena y la tarjeta de telemetría, adquiriendo los datos de longitud, latitud, altura, etc. remotamente y visualizándolos en la interfaz gráfica llamada indicador de horizonte (fig. 7.3).

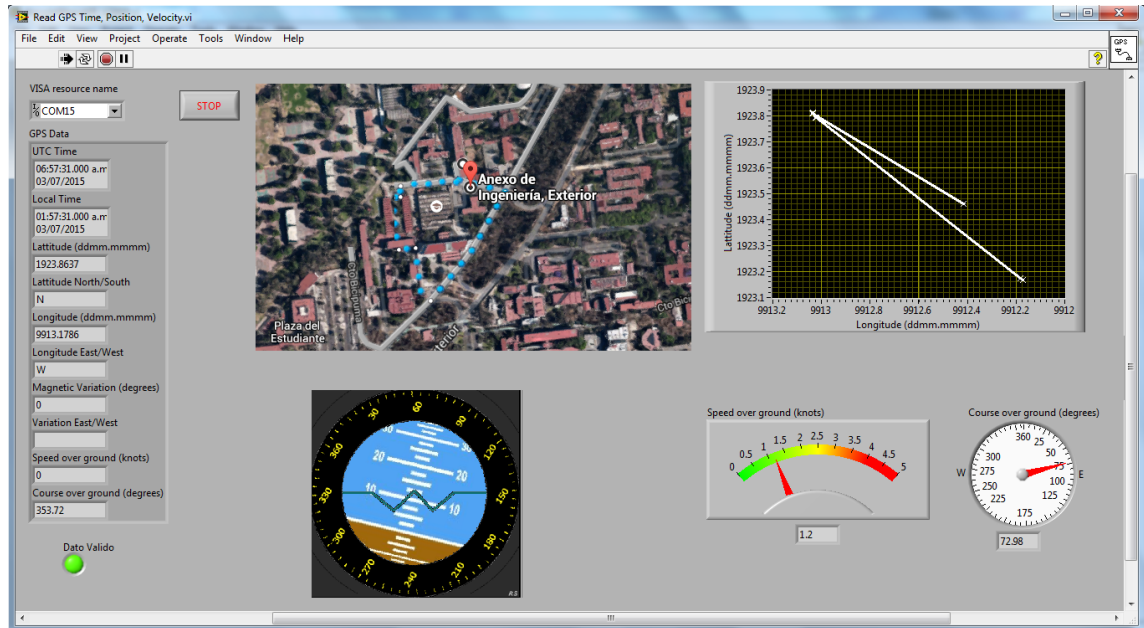


Figura 7. 3 Estación terrena (GCS). [63]

Para la realización de pruebas fue necesario estabilizar el vuelo del UAV por medio de la tarjeta kk2.0 que contiene el modelo, a la cual es necesario sintonizar valores de ganancias proporcionales e integrales propias del algoritmo de estabilización que esta provee. Una vez realizado esto se realiza el despegue manual y posteriormente se activa el modo autónomo para revisar el funcionamiento en vuelo del algoritmo.



Figura 7. 4 UAV en vuelo [64]

CAPITULO 8. Conclusiones

Como se ha mostrado en los capítulos anteriores se implementó un sistema de navegación inercial el cual depende de variables físicas las cuales están relacionadas a ecuaciones cinemáticas de movimiento rectilíneo uniforme. Como primer acercamiento a este sistema y debido a la investigación realizada, es notoria la susceptibilidad de la IMU a variables externas como lo son las vibraciones del propio UAS y a redundancias propias de los giroscopios así como de las capacidades del algoritmo de control. En la implementación presentada se ignoran los efectos aerodinámicos y de control y estabilización del modelo.

1. En este trabajo se utilizó el microcontrolador TM4C123GH6PM producido por Texas Instruments el cual tiene características de operación dentro de un rango aceptable, este maneja una longitud de 32 bits lo cual minimiza el error inherente en cuanto a la conversión digital de las muestras dado que incrementa la resolución de ellas. Para el caso en que se implemente un sistema de gestión de vuelo o FMS abordado es recomendable utilizar tecnologías como lo es el DSP, esto con la intención de disminuir el tiempo de procesamiento de datos y la respuesta del algoritmo de control sea más eficiente ya que es vital en sistemas de navegación y control de aeronaves.
2. En cuanto a la tarjeta utilizada para la medición de variables físicas elaborada por Texas Instruments tiene muy buenas prestaciones en cuanto a las librerías que suministra, asimismo el sensor de 9 ejes MPU9150 cuenta con herramientas que facilitan la minimización de error mediante la integración de datos por medio del algoritmo de Filtro Complementario, que proporciona el fabricante en las librerías del dispositivo.
3. El GPS GTPA010 presenta un comportamiento aceptable dentro de los alcances de este trabajo, este cuenta con una precisión de 3m lo cual en términos de posicionamiento global es aceptable. Dicho dispositivo está limitado a 10 muestras por segundo por lo que para aplicaciones en las que se requiera una exactitud mayor es recomendable sustituirlo por un circuito con una precisión y una resolución más altas pero en este caso son restringidos a la venta en algunos países en el ámbito civil y como todo instrumento debe tener puntos de operación óptimos al proyecto y magnificar su eficacia y eficiencia debido a variables externas. Además es conveniente complementar estos datos con algoritmos que integren las

lecturas con otros dispositivos para poder disminuir los errores comunes en el cálculo de la posición.

4. La comunicación por medio de las antenas de Zigbee es eficiente en el caso de transmitir datos hacia la estación terrena con la finalidad de monitoreo, para el caso de control vía la estación terrena no se recomienda utilizar esta tecnología por lo que para aplicaciones en un UAS es conveniente hacer el procesamiento de datos en la tarjeta de abordo para evitar la latencia y probables errores de propagación y así mejorar la respuesta en tiempo real y por consecuencia las acciones de control del modelo. Una de las ventajas sobre el uso de los módulos de tecnología Zigbee es el hecho que la FCC (Federal Communications Commission) permitió la operación de dispositivos que emitan como máximo 1 Watt de potencia, esto solo en tres bandas de frecuencia: 902 a 928Mhz, 2,400 a 2,483.5 Mhz y 5,725 a 5,850 Mhz. por lo tanto son de libre uso y con la seguridad de no intervenir con alguna otra frecuencia.

Como se vio en la comparativa de protocolos de comunicación inalámbrica, existen diversas tecnologías que nos permiten realizar una comunicación a corto alcance, dejando al usuario la evaluación de las necesidades del proyecto su consideración de tres factores clave: Ancho de banda, Rango y potencia.

5. Para lograr realizar una localización en tiempo real en Google Earth es necesario contratar permisos para poder acceder desde LabView, para el caso de este proyecto se limitó a almacenar los datos de un recorrido y posteriormente proyectarlos en una página con conexión a la red.
6. El comportamiento del UAV debido al algoritmo de control fue un tanto errático debido a factores como la estabilización del vuelo y la calibración de las ganancias KP y Kd por lo que se deduce que el modelo propuesto obtiene una respuesta lenta para llegar al punto deseado. Se observa en el vuelo que el algoritmo implementado tiene poca tendencia a seguir el patrón indicado tal como si la respuesta fuese tardía. Para poder corregir los resultados obtenidos es necesario estabilizar el vuelo del UAV en un grado que pueda ser manipulable desde un radio control. Una vez realizado esto el algoritmo PD presentado se puede expandir a un PID o inclusive para poder reducir el error en una mejor cantidad y así se logre efectivamente.

Las aportaciones de este trabajo se derivan de la integración de varios sistemas, entre ellos se encuentran el de vuelo en modo manual y autónomo, interfaces de acoplamiento entre subsistemas y capacidad de escalabilidad para la adaptabilidad del sistema a misiones específicas.

Se implementó una tarjeta para adaptar los niveles de señales entre las etapas de radio control, microcontrolador y tarjeta de control y estabilización de vuelo, así como de alimentación para el conjunto de dispositivos de señales. (Ver anexo F)

El trabajo presentado conjunta las bases y conocimientos mínimos para obtener un punto de vista más objetivo en cuanto a la implementación de un sistema de navegación más robusto, seleccionando de acuerdo a las necesidades y a la medida cada uno de los dispositivos a utilizar así como también el uso de algoritmos y filtros más poderosos que ayuden a corregir los problemas identificados.

Un problema que es importante mencionar es que la navegación inercial tiene el inconveniente de que los errores son acumulativos, es decir una pequeña desviación en las estimaciones iniciales de la posición se van convirtiendo con el paso del tiempo en un gran error, por esto es recomendable combinarse con un algoritmo de mayor capacidad que nos permita tomar las cualidades de cada uno de los instrumentos en diferentes situaciones y así poder disminuir la tasa de error que se adhiere en cada calculo.

La propuesta para trabajo a futuro es trabajar en los algoritmos de control tanto para la trayectoria deseada como para la estabilización de vuelo, la primera mediante el uso del controlador PID y utilizando técnicas de Kalman (Filtro de Kalman, Filtro de Kalman Extendido), logrando la integración del GPS a los cálculos los cuales tienen propiedades adecuadas para una mejora considerable en cuanto a la estabilidad de la trayectoria de navegación, dando continuidad al mejoramiento de dicho piloto automático.

BIBLIOGRAFIA

Mohinder s. Greywall, "Global Position Systems, Inertial Navigation and Integration", Ed. Jhon Wiley & Sons, 2001

Smith James & Weege Matt & Peters Kevin (2014). "All Purpose Mobile GPS. Senior Project", California Polytechnic State University, San Luis Obispo, CA

España, Martín. 2010; "Fundamentos de la Navegación Integrada", AADECA. Asociación Argentina de Control Automático. Argentina, 228p

Sanchis, Enrique, coord., "Fundamentos y electrónica de las comunicaciones", Universitat de Valencia, 2004.

Valvano, Jonathan W, "Embedded Systems Real-Time Operating Systems for the ARM Cortex-M3 ", 2012, Vol. 3

Artés Antonio, Pérez Fernando; "Comunicaciones Digitales", www.tsc.uc3m.es/~antonio/libro_comunicaciones, 2012

Kazem Sohraby, Minoli Daniel; "Wireless Sensor Networks", Canada, Ed. Wiley, 2007.

Nguyen Ho Quoc Phuong, Hee-Jun Kang¹, "A DCM Based Orientation Estimation Algorithm with an Inertial Measurement Unit and a Magnetic Compass", *Journal of Universal Computer Science*, vol. 15, no. 4 (2009), 859-876 submitted: 30/6/08, accepted: 27/2/09, appeared: 28/2/09 © J.UCS

Valvano, Jonathan W, "Real Time Interfacing to ARM Cortex™-M Microcontrollers", 2012, Vol.2 segunda edición.

"GPS The first Global Navigation Satellite System", California, Ed. Trimble 2007. www.trimble.com

Oliver J. Woodman "An introduction to inertial navigation", Inglaterra 2007 Technical report. University of Cambridge

Bibliografía Figuras:

[1] Recuperada el 3 de Octubre de 2014 de <http://www.zona-militar.com/foros/threads/sobre-avion-portaavion-submarino-portaavion-parasitos-uav-y-demas-yerba.18319/>

[2] Recuperada el 6 de Octubre de 2014 de <http://1000aircraftphotos.com/Contributions/Riabsev/2244.htm>

[3] Recuperada de cartel N. Kemper, L. Santamaría, F Rentería, L. Ochoa (2014), Desarrollo de un sistema difuso aplicado al despegue de micro-UAV, SOMI Congreso de Instrumentación XXIX Edición, CCADET, UNAM, México

[4] Recuperada el 15 de Octubre de 2014 de https://fenix.tecnico.ulisboa.pt/downloadFile/395139421058/Tese_de_Mestrado.pdf

[5] Recuperada el 15 de Octubre de 2014 de:

https://fenix.tecnico.ulisboa.pt/downloadFile/395139421058/Tese_de_Mestrado.pdf

[6] Recuperada el 15 de Octubre de 2014 de:

https://fenix.tecnico.ulisboa.pt/downloadFile/395139421058/Tese_de_Mestrado.pdf

[7] Recuperada el 18 de Octubre de 2014 de:

https://fenix.tecnico.ulisboa.pt/downloadFile/395139421058/Tese_de_Mestrado.pdf

[8] Recuperada el 25 de Octubre de 2014 de

https://www.cl.cam.ac.uk/techreports/UCAM-CL-TR-___696.pdf

[9] Rangel Vargas R. 2015

[10] Recuperada el 30 de Octubre de 2014 de

https://autonav.in.tum.de/handouts/lecture_3_part_1.pdf

- [11] Recuperada el 7 de Noviembre de 2014 de
<http://www.manualvuelo.com/INS/INS26.html>
- [12] Recuperada el 15 de Noviembre de 2014 de
<http://www.manualvuelo.com/INS/INS26.html>
- [13] Recuperada el 12 de Noviembre de 2014 de
<http://www.mecanicaindustrial.com.br/conteudo/237-o-que-e-um-altimetro>
- [14] Recuperada el 7 de Noviembre de 2014 de
<http://www.gisiberica.com/MANOMETROS/MP120E.htm>
- [15] Recuperada el 7 de Noviembre de 2014 de
<http://flystore.com.br/bussola.html>
- [16] Recuperada el 7 de Noviembre de 2014 de Trimble, "GPS The First Global Navigation Satellite System", Trimble Navigation Limited, 2007
- [17] Rangel Vargas R. 2014
- [18] Rangel Vargas R. 2014
- [19] Recuperado el 16 de enero de 2015 de:
<http://tecnologiadeconectividad.blogspot.mx/>
- [20] Recuperada el 20 de enero de 2015 de
<http://rua.ua.es/dspace/bitstream/10045/1109/1/InformeTecZB.pdf>
- [21] Recuperado el 22 de enero de 2015 de
http://catarina.udlap.mx/u_dl_a/tales/documentos/lem/archundia_p_fm/capitulo3.pdf
- [22] Recuperado el 27 de enero de 2015 de
http://es.slideshare.net/victerpaul/zig-bee-ieee-802154?qid=ec9b8974-8fc6-4646-bb4a-9692d8c46dc5&v=default&b=&from_search=1
- [23] Rangel Vargas R. 2015
- [24] Recuperado el 3 de Febrero de 2015 de
http://es.slideshare.net/victerpaul/zig-bee-ieee-802154?qid=ec9b8974-8fc6-4646-bb4a-9692d8c46dc5&v=default&b=&from_search=1

- [25] Recuperado el 10 de Febrero de 2015 de <http://www.ual.es/~vruiz/Docencia/Apuntes/Transmission/04-Modulacion/index.html>
- [26] Recuperado el 10 de Febrero de 2015 de https://es.wikipedia.org/wiki/Modulaci%C3%B3n_por_desplazamiento_de_fase
- [27] Recuperado el 10 de Febrero de 2015 de <http://www.monografias.com/trabajos33/implementacion-bpsk/implementacion-bpsk.shtml>
- [28] Rangel Vargas R. 2015
- [29] Valvano, Jonathan, "Real time interfacing to ARM Cortex Microcontrollers", 2nd edition, 2012.
- [30] Texas Instruments. "Tiva™ TM4C123GH6PM Microcontroller". Data Sheet, Junio 2014.
- [31] Valvano, Jonathan, "Real time interfacing to ARM Cortex Microcontrollers", 2nd edition, 2012.
- [32] Texas Instruments. "Tiva™ TM4C123GH6PM Microcontroller". Data Sheet, Junio 2014.
- [33] Texas Instruments. "Tiva™ TM4C123GH6PM Microcontroller". Data Sheet, Junio 2014.
- [34] Texas Instruments. "Tiva™ TM4C123GH6PM Microcontroller". Data Sheet, Junio 2014.
- [35] Rangel Vargas R. 2015
- [36] Recuperado el 20 de Marzo de 2015 de <http://www.ti.com/ww/en/launchpad/launchpad.html?DCMP=mcu-launchpad&HQS=launchpad>
- [37] Recuperado el 25 de Marzo de 2015 de Texas Instruments. "Boostxl-Senshub Sensor Hub Boosterpack". User Manual SPMU290, Abril 2013.
- [38] User's manual SPMU-290 Texas Instruments, Abril 2013

- [39] Recuperado el 28 de Marzo de 2015 de: <http://www.avnet-memec.eu/products/invensense-mpu-9150-9-axis-motion-tracking-device.html>
- [40] "MPU-9150 Product Specification". MPU-9150 Datasheet.
- [41] Invensense. "MPU-9150 Product Specification". MPU-9150 Datasheet.
- [42] Bosch Sensortec. "BMP180 Digital pressure sensor". BMP180 Datasheet.
- [43] Bosch Sensortec. "BMP180 Digital pressure sensor". BMP180 Datasheet.
- [44] Recuperado el 15 de Abril de 2015 de:
<http://www.buildyourowndrone.co.uk/mediatek-mt3329-gps-10hz-adapter-basic.html>
- [45] Rangel Vargas R. 2015
- [46] Rangel Vargas R. 2015
- [47] Recuperado el 6 de Mayo de 2015 de
http://issuu.com/ric_noc/docs/memoria_trico
- [48] Rangel Vargas R. 2015
- [49] Recuperado el 20 de Abril de 2015 de: <https://en.wikipedia.org/wiki/XBee>
- [50] Recuperado el 27 de Abril de 2015 de:
http://www.spikenzielabs.com/Catalog/index.php?main_page=product_info&products_id=950
- [51] Rangel Vargas R. 2015.
- [52] Rangel Vargas R. 2015.
- [53] Rangel Vargas R. 2015.
- [54] Recuperado el 3 de Mayo de 2015 de
<http://www.digi.com/support/supporttype?type=documentation>
- [55] Rangel Vargas R. 2015
- [56] Rangel Vargas R. 2015
- [57] Rangel Vargas R. 2015

[58] Rangel Vargas R. 2015

[59] Rangel Vargas R. 2015

[60] Rangel Vargas R. 2015

[61] Rangel Vargas R. 2015

[62] Rangel Vargas R. 2015

[63] Rangel Vargas R. 2015

[64] Rangel Vargas R. 2015

Bibliografía Tablas:

[1] Recuperado de cartel N. Kemper, L. Santamaría, F Rentería, L. Ochoa (2014), Desarrollo de un sistema difuso aplicado al despegue de micro-UAV, SOMI Congreso de Instrumentación XXIX Edición, CCADET, UNAM, México

[2] España, Martín. 2010; "Fundamentos de la Navegación Integrada", AADECA. Asociación Argentina de Control Automático. Argentina

[3] Recuperada el 6 de diciembre de 2014 de http://educativa.catedu.es/44700165/aula/archivos/repositorio/3000/3233/html/3_aplicaciones_de_las_ondas_electromagnticas_telecomunicaciones.html

[4] Rangel Vargas R. 2015

[5] Rangel Vargas R. 2015

[6] Texas Instruments. "Tiva™ TM4C123GH6PM Microcontroller". Data Sheet, Junio 2014.

[7] Invensense. "MPU-9150 Product Specification". MPU-9150 Datasheet.

[8] Global Top," FGPMOPA6E Datasheet", 2010

REFERENCIAS

Capítulo 2

[1] Ladislav G. Freire Buillon, Revista Perfiles IDS, 100p

Capítulo 3

[1] Mohinder s. Greywall, "Global Position Systems, Inertial Navigation and Integration", Ed. Jhon Wiley & Sons, 2001

[2] <http://www.grc.nasa.gov/WWW/k-12/airplane/rotations.html>

[3] España, Martín. 2010; "Fundamentos de la Navegación Integrada", AADECA. Asociación Argentina de Control Automático. Argentina, 228p

[4] http://psic.fi.uba.ar/Publicaciones/books/Fundamentos_INS_-MEspania-2010.pdf

[5] España, Martín. 2010; "Fundamentos de la Navegación Integrada", AADECA. Asociación Argentina de Control Automático. Argentina, 228p

Capítulo 4

[1] Sanchis, Enrique, coord., "Fundamentos y electrónica de las comunicaciones", Universitat de Valencia, 2004.

[2] <http://www.ptolomeo.unam.mx:8080/jspui/bitstream/132.248.52.100/229/5/A5.pdf>

[3] <http://es.slideshare.net/chicoral/estndares-inalmblicos>

[4] http://es.slideshare.net/victerpaul/zig-bee-ieee-802154?qid=ec9b8974-8fc6-4646-bb4a-9692d8c46dc5&v=default&b=&from_search=1

Capítulo 5

[1] http://www.nmea.org/content/nmea_standards/nmea_0183_v_410.asp

GLOSARIO

ASCII	Código Americano Estándar para el Intercambio de Información
DCM	Direct Cosine Matrix
DSP	Digital Signal Processor
E2PROM	Electrically Erasable Programmable Read Only Memory
FCS	Flight Control System
FMS	Flight Management System
IMU	Inertial Mass Unit
INS	Inertial Navigation System
I2C	Inter Integrated Communication
GCS	Ground Control System
GPS	Global Position System
GLONASS	Global Orbiting Navigation Satellite System
NMEA	National Marine Electronics Association
MEMS	Micro Electro Mechanical Systems
RPAS	Remotely Piloted Aircraft System
UART	Universal Asynchronous Receiver Transmitter
UAS	Unmanned Aircraft System
UAV	Unmanned Aerial Vehicle
UHF	Ultra High Frequency
VTOL	Vertical Take Off Landing

Anexos

Anexo A Matriz DCM

En general orientación cinemática se ocupa de calcular la orientación relativa de un cuerpo con respecto a un sistema global de coordenadas. Es útil para la conexión de un sistema de coordenadas a nuestro cuerpo en movimiento $Oxyz$, y otro fijo llamado marco global $OXYZ$. Ambos marcos global y local tienen el mismo origen fijo O (véase la Fig. 1). Definiendo los vectores unitarios \mathbf{i} , \mathbf{j} , \mathbf{k} como vectores directores de los ejes x , y y z ; así como los vectores unitarios \mathbf{I} , \mathbf{J} , \mathbf{K} lo son para el marco global $OXYZ$.

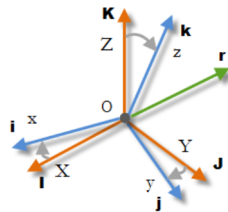


Figura 1

Por lo que, por definición, expresada en términos de coordenadas globales los vectores \mathbf{I} , \mathbf{J} , \mathbf{K} puede ser escrita como:

$$\mathbf{I}^G = \{1, 0, 0\}^T, \mathbf{J}^G = \{0, 1, 0\}^T, \mathbf{K}^G = \{0, 0, 1\}^T$$

Y de igual modo, en términos de vectores coordenadas locales \mathbf{i} , \mathbf{j} , \mathbf{k} se puede escribir como:

$$\mathbf{i} = \{1, 0, 0\}^T, \mathbf{j} = \{0, 1, 0\}^T, \mathbf{k} = \{0, 0, 1\}^T$$

Ahora vamos a ver si podemos escribir los vectores \mathbf{i} , \mathbf{j} , \mathbf{k} en términos de coordenadas globales. Tomando al vector \mathbf{i} como ejemplo para escribir sus coordenadas globales:

Sea el vector \mathbf{i}

$$\mathbf{i}^G = \{i_x^G, i_y^G, i_z^G\}^T$$

Por ejemplo, analizando la coordenada X i_x^G , es calculado como la longitud de la proyección del vector \mathbf{i} en el eje global X .

$$i_x^G = |i| \cos(X,i) = \cos(I,i)$$

Donde $|i|$ es la norma (longitud) del vector unitario i y $\cos(I,i)$ es el coseno del ángulo formado por los vectores I y i . Utilizando el hecho de que $|I| = 1$ y $|i| = 1$ (que son vectores unitarios por definición). Podemos escribir:

$$i_x^G = \cos(I,i) = |I||i| \cos(I,i) = I \cdot i$$

Donde $I \cdot i$ es el producto escalar entre vectores I . Para el cálculo de productos escalares $I \cdot i$ no importa en qué sistema de coordenadas estos vectores siempre que estén expresadas en el mismo sistema, ya que la rotación no modifica el ángulo entre vectores:

$I \cdot i = I^B \cdot i^B = I^G \cdot i^G = \cos(I^B \cdot i^B) = \cos(I^G \cdot i^G)$, de manera que por simplicidad omitiremos el superíndice en productos escalares $I \cdot i$, $J \cdot j$, $K \cdot k$ y cosenos $\cos(I,i)$, $\cos(J,j)$, $\cos(K,k)$.

De igual manera se puede demostrar que:

$i_y^G = J \cdot i$, $i_z^G = K \cdot i$, por lo que ahora podemos escribir vectores, en términos de sistema global de coordenadas como:

$$i^G = \{ I \cdot i, J \cdot i, K \cdot i \}^T$$

Además, del mismo modo, puede ser demostrado que $j^G = \{ I \cdot j, J \cdot j, K \cdot j \}^T$, $k^G = \{ I \cdot k, J \cdot k, K \cdot k \}^T$

Ahora tenemos un completo conjunto de coordenadas globales de nuestro cuerpo para vectores i, j, k y podemos organizar estos valores en una cómoda forma de matriz:

$$[i^G, j^G, k^G] = \begin{bmatrix} I \cdot i & I \cdot j & I \cdot k \\ J \cdot i & J \cdot j & J \cdot k \\ K \cdot i & K \cdot j & K \cdot k \end{bmatrix} = \begin{bmatrix} \cos(I,i) & \cos(I,j) & \cos(I,k) \\ \cos(J,i) & \cos(J,j) & \cos(J,k) \\ \cos(K,i) & \cos(K,j) & \cos(K,k) \end{bmatrix} = DCM^G \text{ (Eq. 1.1)}$$

Esta matriz se denomina Matriz de Coseno Dirección ahora motivos obvios, se compone de los cosenos de los ángulos de todas las combinaciones posibles de vectores globales y locales.

La tarea de expresar el marco global de los vectores unitarios I^G, J^G, K^G en coordenadas locales es por la naturaleza simétrica y se puede conseguir con sólo cambiar la notación I, J, K con i, j, k , y el resultado es:

$$I^B = \{ I \cdot i, I \cdot j, I \cdot k \}^T, \quad J^B = \{ J \cdot i, J \cdot j, J \cdot k \}^T, \quad K^B = \{ K \cdot i, K \cdot j, K \cdot k \}^T$$

y organizada en forma de matriz:

$$[\mathbf{I}^B, \mathbf{J}^B, \mathbf{K}^B] = \begin{bmatrix} \mathbf{I} \cdot \mathbf{i} & \mathbf{J} \cdot \mathbf{i} & \mathbf{K} \cdot \mathbf{i} \\ \mathbf{I} \cdot \mathbf{j} & \mathbf{J} \cdot \mathbf{j} & \mathbf{K} \cdot \mathbf{j} \\ \mathbf{I} \cdot \mathbf{k} & \mathbf{J} \cdot \mathbf{k} & \mathbf{K} \cdot \mathbf{k} \end{bmatrix} = \begin{bmatrix} \cos(\mathbf{I}, \mathbf{i}) & \cos(\mathbf{J}, \mathbf{i}) & \cos(\mathbf{K}, \mathbf{i}) \\ \cos(\mathbf{I}, \mathbf{j}) & \cos(\mathbf{J}, \mathbf{j}) & \cos(\mathbf{K}, \mathbf{j}) \\ \cos(\mathbf{I}, \mathbf{k}) & \cos(\mathbf{J}, \mathbf{k}) & \cos(\mathbf{K}, \mathbf{k}) \end{bmatrix} = \text{DCM}^B \quad (\text{Eq. 1.2})$$

Ahora es fácil notar que $\text{DCM}^B = (\text{DCM}^G)^T$ o bien $\text{DCM}^G = (\text{DCM}^B)^T$, en otras palabras las dos matrices son transpuestas una de otra, por lo que se dice son ortogonales. Utilizaremos esta propiedad importante más adelante.

Otra propiedad es que $\text{DCM}^B \cdot \text{DCM}^G = (\text{DCM}^G)^T \cdot \text{DCM}^G = \text{DCM}^B \cdot (\text{DCM}^B)^T = \mathbf{I}_3$, donde \mathbf{I}_3 es una matriz identidad de 3x3. En otras palabras, las matrices DCM son ortogonales.

La matriz DCM (también llamada matriz de rotación), tiene una gran importancia en la orientación dentro de la cinemática ya que define el giro de un marco con relación a otro. También se puede utilizar para determinar las coordenadas globales de un vector arbitrario si sabemos sus coordenadas en el marco local de la carrocería y viceversa.

Considerando un vector con coordenadas locales:

$\mathbf{r}^B = \{ r_x^B, r_y^B, r_z^B \}^T$ determinaremos sus coordenadas en el marco global, mediante el uso de la matriz de rotación giro DCM^G .

Partiendo del vector $\mathbf{r}^G = \{ r_x^G, r_y^G, r_z^G \}^T$

La primera coordenada r_x^G se puede definir como $r_x^G = |\mathbf{r}^G| \cos(\mathbf{I}^G, \mathbf{r}^G)$, esto porque r_x^G es la proyección del vector \mathbf{r}^G sobre el eje X el cual es co-direccional con \mathbf{I}^G .

A continuación por definición, una rotación es una transformación que no cambia la escala de un vector y no cambia el ángulo entre dos vectores que están sujetos a la misma rotación, por lo tanto, si queremos expresar algún

os vectores en un diferente sistema de coordenadas girado la norma y el ángulo entre vectores no va a cambiar:

$|\mathbf{r}^G| = |\mathbf{r}^B|$, $|\mathbf{I}^G| = |\mathbf{I}^B| = 1$ y $\cos(\mathbf{I}^G, \mathbf{r}^G) = \cos(\mathbf{I}^B, \mathbf{r}^B)$, así que podemos utilizar esta propiedad para escribir

$r_x^G = |r^G| \cos(\mathbf{l}^G, \mathbf{r}^G) = |\mathbf{l}^B| |\mathbf{r}^B| \cos(\mathbf{l}^B, \mathbf{r}^B) = \mathbf{l}^B \cdot \mathbf{r}^B = \mathbf{l}^B \cdot \{r_x^B, r_y^B, r_z^B\}^T$, utilizando una de las dos definiciones del producto escalar.

Retomando que $\mathbf{l}^B = \{ \mathbf{l} \cdot \mathbf{i}, \mathbf{l} \cdot \mathbf{j}, \mathbf{l} \cdot \mathbf{k} \}^T$ y usando la definición del producto escalar:

$$r_x^G = \mathbf{l}^B \cdot \mathbf{r}^B = \{ \mathbf{l} \cdot \mathbf{i}, \mathbf{l} \cdot \mathbf{j}, \mathbf{l} \cdot \mathbf{k} \}^T \cdot \{ r_x^B, r_y^B, r_z^B \}^T = r_x^B \mathbf{l} \cdot \mathbf{i} + r_y^B \mathbf{l} \cdot \mathbf{j} + r_z^B \mathbf{l} \cdot \mathbf{k}$$

En la misma forma en que se puede demostrar que:

$$r_y^G = r_x^B \mathbf{j} \cdot \mathbf{i} + r_y^B \mathbf{j} \cdot \mathbf{j} + r_z^B \mathbf{j} \cdot \mathbf{k}$$

$$r_z^G = r_x^B \mathbf{k} \cdot \mathbf{i} + r_y^B \mathbf{k} \cdot \mathbf{j} + r_z^B \mathbf{k} \cdot \mathbf{k}$$

Por último escribiendo lo anterior en forma matricial queda como:

$$\mathbf{r}^G = \begin{bmatrix} r_x^G \\ r_y^G \\ r_z^G \end{bmatrix} = \begin{bmatrix} \mathbf{l} \cdot \mathbf{i} & \mathbf{l} \cdot \mathbf{j} & \mathbf{l} \cdot \mathbf{k} \\ \mathbf{j} \cdot \mathbf{i} & \mathbf{j} \cdot \mathbf{j} & \mathbf{j} \cdot \mathbf{k} \\ \mathbf{k} \cdot \mathbf{i} & \mathbf{k} \cdot \mathbf{j} & \mathbf{k} \cdot \mathbf{k} \end{bmatrix} \begin{bmatrix} r_x^B \\ r_y^B \\ r_z^B \end{bmatrix} = \text{DCM}^G \mathbf{r}^B \quad (\text{Eq. 1.4})$$

Por lo tanto la matriz DCM puede utilizarse para convertir un vector arbitrario \mathbf{r}^B expresado en un sistema de coordenadas B, a un sistema de coordenadas girado G.

Es posible usar una lógica similar para probar el proceso inverso:

$$\mathbf{r}^B = \text{DCM}^B \mathbf{r}^G \quad (\text{Eq. 1.5})$$

O bien podemos llegar a la misma conclusión multiplicando ambos lados en (Eq. 1.4) por DCM^B que equivale a DCM^{GT} , y utilizando la propiedad que $\text{DCM}^{GT} \cdot \text{DCM}^G = \mathbf{I}_3$, ver (Eq. 1.3) :

$$\text{DCM}^B \mathbf{r}^G = \text{DCM}^B \cdot \text{DCM}^G \cdot \mathbf{r}^B = \text{DCM}^{GT} \cdot \text{DCM}^G \cdot \mathbf{r}^B = \mathbf{I}_3 \mathbf{r}^B = \mathbf{r}^B$$

Anexo B Rutina de programación de la telemetría

```
//*****  
//  
//  
// Copyright (c) 2013 Texas Instruments Incorporated. All rights reserved.  
// Software License Agreement  
//  
// Texas Instruments (TI) is supplying this software for use solely and  
// exclusively on TI's microcontroller products. The software is owned by  
// TI and/or its suppliers, and is protected under applicable copyright  
// laws. You may not combine this software with "viral" open-source  
// software in order to form a larger program.  
//  
// THIS SOFTWARE IS PROVIDED "AS IS" AND WITH ALL FAULTS.  
// NO WARRANTIES, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING, BUT  
// NOT LIMITED TO, IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR  
// A PARTICULAR PURPOSE APPLY TO THIS SOFTWARE. TI SHALL NOT, UNDER ANY  
// CIRCUMSTANCES, BE LIABLE FOR SPECIAL, INCIDENTAL, OR CONSEQUENTIAL  
// DAMAGES, FOR ANY REASON WHATSOEVER.  
//  
// This is part of revision 1.1 of the EK-TM4C123GXL Firmware Package.  
//  
//*****  
  
#include <stdint.h>  
#include <stdbool.h>  
#include <math.h> //BMP180  
#include "inc/hw_memmap.h"  
#include "inc/hw_ints.h"  
#include "inc/hw_types.h" //BMP180  
#include "driverlib/debug.h"  
#include "driverlib/gpio.h"  
#include "driverlib/interrupt.h"  
#include "driverlib/pin_map.h"  
#include "driverlib/rom.h"  
#include "driverlib/sysctl.h"  
#include "driverlib/systick.h" //BMP180  
#include "driverlib/uart.h"  
#include "utils/uartstdio.h"  
#include "sensorlib/hw_mpu9150.h"  
#include "sensorlib/hw_bmp180.h" //BMP180  
#include "sensorlib/hw_ak8975.h"  
#include "sensorlib/i2cm_drv.h"  
#include "sensorlib/ak8975.h"  
#include "sensorlib/mpu9150.h"  
#include "sensorlib/bmp180.h" //BMP180  
#include "sensorlib/comp_dcm.h"  
#include "drivers/rgb.h"
```



```

//*****
// MPU9150 I2C Address.
//*****
#define MPU9150_I2C_ADDRESS 0x68
#define BMP180_I2C_ADDRESS 0x77 //BMP180

//*****
//
// \addtogroup example_list
// Nine Axis Sensor Fusion with the MPU9150 and Complimentary-Filtered
// DCM (compdcm_mpu9150)
//*****

//*****
// Global array for holding the color values for the RGB.
//*****
uint32_t g_pui32Colors[3]; //EN BMP180

//*****
// Global instance structure for the I2C master driver.
//*****
tI2CInstance g_sl2CInst; //EN BMP180

//*****
// Global instance structure for the ISL29023 sensor driver.
//*****
tMPU9150 g_sMPU9150Inst;
tBMP180 g_sBMP180Inst; //BMP180

//*****
// Global Instance structure to manage the DCM state.
//*****
tCompDCM g_sCompDCMInst;
//*****
// Global new data flag to alert main that BMP180 data is ready.
//*****
volatile uint_fast8_t g_vui8DataFlag; ///SE REPITE ABAJO

//*****
// Global flags to alert main that MPU9150 I2C transaction is complete
//*****
volatile uint_fast8_t g_vui8I2CDoneFlag;

//*****
// Global flags to alert main that MPU9150 I2C transaction error has occurred.
//*****
volatile uint_fast8_t g_vui8ErrorFlag;

//*****
// Global flags to alert main that MPU9150 data is ready to be retrieved.
//

```

```

//*****
volatile uint_fast8_t g_vui8DataFlag;

//*****
// Global counter to control and slow down the rate of data to the terminal.
//*****
#define PRINT_SKIP_COUNT    10

uint32_t g_ui32PrintSkipCounter;

//*****
// The error routine that is called if the driver library encounters an error.
//*****
#ifdef DEBUG
void
__error__(char *pcFilename, uint32_t ui32Line)
{
}
#endif

//*****
// MPU9150 Sensor callback function.
//*****
void
MPU9150AppCallback(void *pvCallbackData, uint_fast8_t ui8Status)
{
    //
    // If the transaction succeeded set the data flag to indicate to
    // application that this transaction is complete and data may be ready.
    //
    if(ui8Status == I2CM_STATUS_SUCCESS)
    {
        g_vui8I2CDoneFlag = 1;
    }

    //
    // Store the most recent status in case it was an error condition
    //
    g_vui8ErrorFlag = ui8Status;
}

//*****
// BMP180 Sensor callback function. Called at the end of BMP180 sensor driver
// transactions. This is called from I2C interrupt context. Therefore, we just
// set a flag and let main do the bulk of the computations and display.
//*****
void BMP180AppCallback(void* pvCallbackData, uint_fast8_t ui8Status)
{
    if(ui8Status == I2CM_STATUS_SUCCESS)
    {
        g_vui8DataFlag = 1;
    }
}

```

```

    }
}

//*****
//
// Called by the NVIC as a result of GPIO port B interrupt event. For this
// application GPIO port B pin 2 is the interrupt line for the MPU9150
//
//*****
void
IntGPIOB(void)
{
    unsigned long ulStatus;

    ulStatus = GPIOIntStatus(GPIO_PORTB_BASE, true);

    //
    // Clear all the pin interrupts that are set
    //
    GPIOIntClear(GPIO_PORTB_BASE, ulStatus);

    if(ulStatus & GPIO_PIN_2)
    {
        //
        // MPU9150 Data is ready for retrieval and processing.
        //
        MPU9150DataRead(&g_sMPU9150Inst, MPU9150AppCallback, &g_sMPU9150Inst);
    }
}

//*****
// Called by the NVIC as a result of I2C3 Interrupt. I2C3 is the I2C connection
// to the MPU9150.
//*****
void
MPU9150I2CIntHandler(void)
{
    //
    // Pass through to the I2CM interrupt handler provided by sensor library.
    // This is required to be at application level so that I2CMIntHandler can
    // receive the instance structure pointer as an argument.
    //
    I2CMIntHandler(&g_sl2CInst);
}

//*****
// Called by the NVIC as a SysTick interrupt, which is used to generate the
// sample interval
//*****
void

```

```

SysTickIntHandler()
{
    GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_1, GPIO_PIN_1);
    BMP180DataRead(&g_sBMP180Inst, BMP180AppCallback, &g_sBMP180Inst);
    GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_1, 0x00);
}

//*****
// MPU9150 Application error handler. Show the user if we have encountered an
// I2C error.
//*****
void
MPU9150AppErrorHandler(char *pcFilename, uint_fast32_t ui32Line)
{
    // Set terminal color to red and print error status and locations
    UARTprintf("\033[31;1m");
    UARTprintf("Error: %d, File: %s, Line: %d\n"
               "See I2C status definitions in sensorlib\i2cm_drv.h\n",
               g_vui8ErrorFlag, pcFilename, ui32Line);

    //
    // Return terminal color to normal
    //
    UARTprintf("\033[0m");

    //
    // Set RGB Color to RED
    //
    g_pui32Colors[0] = 0xFFFF;
    g_pui32Colors[1] = 0;
    g_pui32Colors[2] = 0;
    RGBColorSet(g_pui32Colors);

    //
    // Increase blink rate to get attention
    //
    RGBBlinkRateSet(10.0f);

    // Go to sleep wait for interventions. A more robust application could
    // attempt corrective actions here.
    //
    while(1)
    {
        //
        // Do Nothing
        //
    }
}

//*****

```

```

//
// Function to wait for the MPU9150 transactions to complete. Use this to spin
// wait on the I2C bus.
//
//*****
void
MPU9150ApplI2CWait(char *pcFilename, uint_fast32_t ui32Line)
{
    //
    // Put the processor to sleep while we wait for the I2C driver to
    // indicate that the transaction is complete.
    //
    while((g_vui8I2CDoneFlag == 0) && (g_vui8ErrorFlag == 0))
    {
        //
        // Do Nothing
        //
    }

    // If an error occurred call the error handler immediately.
    //
    if(g_vui8ErrorFlag)
    {
        MPU9150AppErrorHandler(pcFilename, ui32Line);
    }

    // clear the data flag for next use.
    //
    g_vui8I2CDoneFlag = 0;
}

//*****
// Configure the UART and its pins. This must be called before UARTprintf().
//*****
void
ConfigureUART(void)
{
    //
    // Enable the GPIO Peripheral used by the UART.
    //
    ROM_SysCtlPeripheralEnable(SYSCTL_PERIPH_GPIOA);

    // Enable UART0
    //
    ROM_SysCtlPeripheralEnable(SYSCTL_PERIPH_UART0);

    //
    // Configure GPIO Pins for UART mode.
    //
    ROM_GPIOPinConfigure(GPIO_PA0_U0RX);
    ROM_GPIOPinConfigure(GPIO_PA1_U0TX);
    ROM_GPIOPinTypeUART(GPIO_PORTA_BASE, GPIO_PIN_0 | GPIO_PIN_1);
}

```

```

//
// Use the internal 16MHz oscillator as the UART clock source.
//
UARTClockSourceSet(UART0_BASE, UART_CLOCK_PIOSC);

//
// Initialize the UART for console I/O.
//
UARTStdioConfig(0, 115200, 16000000);
}

//*****
//
// Main Prog..
//
//*****
int
main(void)
{
    int_fast32_t i32IPart[16], i32FPart[16];
    uint_fast32_t ui32Idx, ui32CompDCMStarted;
    float pfData[16];
    float *pfAccel, *pfGyro, *pfMag, *pfEulers, *pfQuaternion;

        float fTemperatura, fPresion, fAltitud; //BMP180
    int32_t i32IntegerPart; //BMP180
    int32_t i32FractionPart; //BMP180
    int gps=0;

    //
    // Initialize convenience pointers that clean up and clarify the code
    // meaning. We want all the data in a single contiguous array so that
    // we can make our pretty printing easier later.
    //
    pfAccel = pfData;
    pfGyro = pfData + 3;
    pfMag = pfData + 6;
    pfEulers = pfData + 9;
    pfQuaternion = pfData + 12;

    //
    // Setup the system clock to run at 40 Mhz from PLL with crystal reference
    //
    ROM_SysCtlClockSet(SYSCTL_SYSDIV_5 | SYSCTL_USE_PLL | SYSCTL_XTAL_16MHZ |
        SYSCTL_OSC_MAIN);

    //
    // Enable port B used for motion interrupt.
    //
    ROM_SysCtlPeripheralEnable(SYSCTL_PERIPH_GPIOB);

```

```

//
// Initialize the UART.
//
ConfigureUART();

//
// Print the welcome message to the terminal.
//
//UARTprintf("\033[2JMPU9150 Raw Example\n");
// UARTprintf("\033[2JBMP180 Example\n"); //BMP180
//
// Set the color to a purple approximation.
//
g_pui32Colors[RED] = 0x8000;
g_pui32Colors[BLUE] = 0x8000;
g_pui32Colors[GREEN] = 0x0000;

//
// Initialize RGB driver.
//
RGBInit(0);
RGBColorSet(g_pui32Colors);
RGBIntensitySet(0.5f);
RGBEnable();

//
// The I2C3 peripheral must be enabled before use.
//
ROM_SysCtlPeripheralEnable(SYSCTL_PERIPH_I2C3);
ROM_SysCtlPeripheralEnable(SYSCTL_PERIPH_GPIOD);

//
// Configure the pin muxing for I2C3 functions on port D0 and D1.
//
ROM_GPIOPinConfigure(GPIO_PD0_I2C3SCL);
ROM_GPIOPinConfigure(GPIO_PD1_I2C3SDA);

//
// Select the I2C function for these pins. This function will also
// configure the GPIO pins for I2C operation, setting them to
// open-drain operation with weak pull-ups. Consult the data sheet
// to see which functions are allocated per pin.
//
GPIOPinTypeI2CSCL(GPIO_PORTD_BASE, GPIO_PIN_0);
ROM_GPIOPinTypeI2C(GPIO_PORTD_BASE, GPIO_PIN_1);

//
// Configure and Enable the GPIO interrupt. Used for INT signal from the
// MPU9150
//
ROM_GPIOPinTypeGPIOInput(GPIO_PORTB_BASE, GPIO_PIN_2);
GPIOIntEnable(GPIO_PORTB_BASE, GPIO_PIN_2);

```

```

ROM_GPIOIntTypeSet(GPIO_PORTB_BASE, GPIO_PIN_2, GPIO_FALLING_EDGE);
ROM_IntEnable(INT_GPIOB);

//
// Keep only some parts of the systems running while in sleep mode.
// GPIOB is for the MPU9150 interrupt pin.
// UART0 is the virtual serial port
// TIMER0, TIMER1 and WTIMER5 are used by the RGB driver
// I2C3 is the I2C interface to the ISL29023
//
ROM_SysCtlPeripheralClockGating(true);
ROM_SysCtlPeripheralSleepEnable(SYSCTL_PERIPH_GPIOB);
ROM_SysCtlPeripheralSleepEnable(SYSCTL_PERIPH_UART0);
ROM_SysCtlPeripheralSleepEnable(SYSCTL_PERIPH_TIMER0);
ROM_SysCtlPeripheralSleepEnable(SYSCTL_PERIPH_TIMER1);
ROM_SysCtlPeripheralSleepEnable(SYSCTL_PERIPH_I2C3);
ROM_SysCtlPeripheralSleepEnable(SYSCTL_PERIPH_WTIMER5);

//
// Enable interrupts to the processor.
//
ROM_IntMasterEnable();

//
// Initialize I2C3 peripheral.
//
I2CMInit(&g_sl2CInst, I2C3_BASE, INT_I2C3, 0xff, 0xff,
        ROM_SysCtlClockGet());

//
// Initialize the MPU9150 Driver.
//
MPU9150Init(&g_sMPU9150Inst, &g_sl2CInst, MPU9150_I2C_ADDRESS,
        MPU9150AppCallback, &g_sMPU9150Inst);

//
// Wait for transaction to complete
//
MPU9150AppI2CWait(__FILE__, __LINE__);

//
// Write application specific sensor configuration such as filter settings
// and sensor range settings.
//
g_sMPU9150Inst.pui8Data[0] = MPU9150_CONFIG_DLPF_CFG_94_98;
g_sMPU9150Inst.pui8Data[1] = MPU9150_GYRO_CONFIG_FS_SEL_250;
g_sMPU9150Inst.pui8Data[2] = (MPU9150_ACCEL_CONFIG_ACCEL_HPF_5HZ |
        MPU9150_ACCEL_CONFIG_AFS_SEL_2G);
MPU9150Write(&g_sMPU9150Inst, MPU9150_O_CONFIG, g_sMPU9150Inst.pui8Data, 3,
        MPU9150AppCallback, &g_sMPU9150Inst);

//

```



```

//
// Configure the data ready interrupt pin output of the MPU9150.
//
g_sMPU9150Inst.pui8Data[0] = MPU9150_INT_PIN_CFG_INT_LEVEL |
    MPU9150_INT_PIN_CFG_INT_RD_CLEAR |
    MPU9150_INT_PIN_CFG_LATCH_INT_EN;
g_sMPU9150Inst.pui8Data[1] = MPU9150_INT_ENABLE_DATA_RDY_EN;
MPU9150Write(&g_sMPU9150Inst, MPU9150_O_INT_PIN_CFG,
    g_sMPU9150Inst.pui8Data, 2, MPU9150AppCallback,
    &g_sMPU9150Inst);

//
// Initialize the DCM system. 50 hz sample rate.
// accel weight = .2, gyro weight = .8, mag weight = .2
//
CompDCMInit(&g_sCompDCMInst, 1.0f / 50.0f, 0.2f, 0.6f, 0.2f);

/////BMP180
/////
//
// Initialize the BMP180
//
BMP180Init(&g_sBMP180Inst, &g_sl2CInst, BMP180_I2C_ADDRESS,
    BMP180AppCallback, &g_sBMP180Inst);

//
// Wait for initialization callback to indicate reset request is complete.
//
while(g_vui8DataFlag == 0)
{
    //
    // Wait for I2C Transactions to complete.
    //
}

//
//
g_vui8DataFlag = 0;

///// BMP180

//
// Enable blinking indicates config finished successfully
//
RGBBlinkRateSet(1.0f);

ui32CompDCMStarted = 0;

while(1)

```

```

{
    //
    // Go to sleep mode while waiting for data ready.
    //
    while(!g_vui8I2CDoneFlag)
    {
        ROM_SysCtlSleep();
    }

    //
    // bandera=0
    //
    g_vui8I2CDoneFlag = 0;

    //
    // Get floating point version of the Accel Data in m/s^2.
    //
    MPU9150DataAccelGetFloat(&g_sMPU9150Inst, pfAccel, pfAccel + 1,
        pfAccel + 2);

    //
    // Punto flotante vel ang rad/sec
    //
    MPU9150DataGyroGetFloat(&g_sMPU9150Inst, pfGyro, pfGyro + 1,
        pfGyro + 2);

    //
    // Get floating point version of magnetic fields strength in tesla
    //
    MPU9150DataMagnetoGetFloat(&g_sMPU9150Inst, pfMag, pfMag + 1,
        pfMag + 2);

    //
    // Check if this is our first data ever.
    //
    if(ui32CompDCMStarted == 0)
    {
        //
        // Set flag indicating that DCM is started.
        // Perform the seeding of the DCM with the first data set.
        //
        ui32CompDCMStarted = 1;
        CompDCMMagnetoUpdate(&g_sCompDCMInst, pfMag[0], pfMag[1],
            pfMag[2]);
        CompDCMAccelUpdate(&g_sCompDCMInst, pfAccel[0], pfAccel[1],
            pfAccel[2]);
        CompDCMGyroUpdate(&g_sCompDCMInst, pfGyro[0], pfGyro[1],
            pfGyro[2]);
        CompDCMStart(&g_sCompDCMInst);
    }
    else

```

```

{
//
// DCM Is already started. Perform the incremental update.
//
CompDCMMagnetoUpdate(&g_sCompDCMInst, pfMag[0], pfMag[1],
                    pfMag[2]);
CompDCMAccelUpdate(&g_sCompDCMInst, pfAccel[0], pfAccel[1],
                  pfAccel[2]);
CompDCMGyroUpdate(&g_sCompDCMInst, -pfGyro[0], -pfGyro[1],
                 -pfGyro[2]);
CompDCMUpdate(&g_sCompDCMInst);
}

//
// Increment the skip counter. Skip counter is used so we do not
// overflow the UART with data.
//
g_ui32PrintSkipCounter++;
if(g_ui32PrintSkipCounter >= PRINT_SKIP_COUNT)
{

//
g_ui32PrintSkipCounter = 0;

//
// Get Euler data. (Roll Pitch Yaw)
//
CompDCMComputeEulers(&g_sCompDCMInst, pfEulers, pfEulers + 1,
                    pfEulers + 2);
// convert mag data to micro-tesla for better human interpretation.
//
pfMag[0] *= 1000000;
pfMag[1] *= 1000000;
pfMag[2] *= 1000000;

//
// Convert Eulers to degrees. 180/PI = 57.29...
// Convert Yaw to 0 to 360 to approximate compass headings.
//
pfEulers[0] *= 57.295779513082320876798154814105f;
pfEulers[1] *= 57.295779513082320876798154814105f;
pfEulers[2] *= 57.295779513082320876798154814105f;
if(pfEulers[2] < 0)
{
    pfEulers[2] += 360.0f;
}

//
for(ui32Idx = 0; ui32Idx < 16; ui32Idx++)
{
//
//

```

```

i32IPart[ui32Idx] = (int32_t) pfData[ui32Idx];

//
// Multiply by 1000 to preserve first three decimal values.
// Truncates at the 3rd decimal place.
//
i32FPart[ui32Idx] = (int32_t) (pfData[ui32Idx] * 1000.0f);

//
// Subtract off the integer part from this newly formed decimal
// part.
//
i32FPart[ui32Idx] = i32FPart[ui32Idx] -
                    (i32IPart[ui32Idx] * 1000);

//
// make the decimal part a positive number for display.
//
if(i32FPart[ui32Idx] < 0)
{
    i32FPart[ui32Idx] *= -1;
}
}

```

```

BMP180DataRead(&g_sBMP180Inst,
BMP180AppCallback, &g_sBMP180Inst);
while(g_vui8DataFlag == 0)
{
    //
    // Wait for the new data set to be available.
    //
}
// Reset the data ready flag.
//
g_vui8DataFlag = 0;

//
// Ultima temperatura en flotante
//
BMP180DataTemperatureGetFloat(&g_sBMP180Inst, &fTemperatura);

//
// Convert the floats to an integer part and fraction part for easy
// print.
//
i32IntegerPart = (int32_t) fTemperatura;
i32FractionPart = (int32_t) (fTemperatura * 1000.0f);
i32FractionPart = i32FractionPart - (i32IntegerPart * 1000);
if(i32FractionPart < 0)
{
    i32FractionPart *= -1;
}

```

```

}

//
// Temperatura con tres digitos
// UARTprintf("\nT= %3d.%03d C, ", i32IntegerPart, i32FractionPart);

// Ultima presion atmosferica, flotante
BMP180DataPressureGetFloat(&g_sBMP180Inst, &fPresion);

// Conversion de flotantes a enteros
i32IntegerPart = (int32_t) fPresion;
i32FractionPart =(int32_t) (fPresion * 1000.0f);
i32FractionPart = i32FractionPart - (i32IntegerPart * 1000);
if(i32FractionPart < 0)
{
    i32FractionPart *= -1;
}

//
// Presion con tres decimales
//
UARTprintf("P= %3d.%03d Pa, ", i32IntegerPart, i32FractionPart);

//
// Calculando la altitud de acuerdo a la ecuacion del fabricante.
//
fAltitud = 44330.0f * (1.0f - powf(fPresion / 101325.0f,
    1.0f / 5.255f));

//
// Convierte float a int para impresion uart
//
i32IntegerPart = (int32_t) fAltitud;
i32FractionPart =(int32_t) (fAltitud * 1000.0f);
i32FractionPart = i32FractionPart - (i32IntegerPart * 1000);
if(i32FractionPart < 0)
{
    i32FractionPart *= -1;
}

//
// Impresion de altitud con tres decimales de precision.
//
UARTprintf("H= %3d.%03d msnm", i32IntegerPart, i32FractionPart);

//
// Print new line.
//
//UARTprintf("\n");

// Imprime cadena de datos aceleraciones
UARTprintf("ax%3d.%03d, ay%3d.%03d, az%3d.%03d\n",
i32IPart[0], i32FPart[0], i32IPart[1], i32FPart[1], i32IPart[2], i32FPart[2]);

```

```

// Imprime cadena de datos giroscopio
    UARTprintf("gx%3d.%03d, gy%3d.%03d, gz%3d.%03d\n",
i32IPart[3], i32FPart[3], i32IPart[4], i32FPart[4], i32IPart[5], i32FPart[5]);
    // Imprime cadena de datos magneticos
    UARTprintf("mx%3d.%03d, my%3d.%03d, mz%3d.%03d\n",
i32IPart[6], i32FPart[6], i32IPart[7], i32FPart[7], i32IPart[8], i32FPart[8]);
    // Imprime cadena de ángulos de Euler
    i32IPart[10]=i32IPart[10]*(-1);
    i32IPart[9]=i32IPart[9]*(-1);
    i32IPart[11]=i32IPart[11]*(-1);

    //

    UARTprintf("%3d.%03d,%3d.%03d,%3d.%03d\n", i32IPart[10],
i32FPart[10], i32IPart[9], i32FPart[9], i32IPart[11], i32FPart[11]);
    // Obtiene datos gps e imprime
    UARTgets("%c",gps);

//
// Delay para estabilizar.
//
ROM_SysCtlDelay(ROM_SysCtlClockGet() / (10 * 3));

}
}
}

```

Anexo C Índices

Índice de Imágenes

Capítulo 1

Figura 1.1 Avión hércules cargando dos drones Firebee.....	12
Figura 1.2 Dron la-17 de origen soviético.....	13

Capítulo 2

Figura 2.1 Sistema Aéreo no tripulado.....	19
---	----

Capítulo 3

Figura 3. 1 Movimiento de Alabeo o Roll.....	30
Figura 3. 2 Movimiento de cabeceo o pitch.....	30
Figura 3. 3 Movimiento de guiñada o yaw.....	31
Figura 3. 4 Momentos sobre ejes ortogonales en una aeronave y un UAV.....	31
Figura 3. 5 Marcos de referencia del INS.....	32
Figura 3. 6 Matriz DCM.....	33
Figura 3. 7 Conversión de Matriz DCM a ángulos de Euler.....	33
Figura 3. 8 Interpretación grafica del horizonte artificial.....	34
Figura 3. 9 Componentes de un horizonte artificial mecánico.....	35
Figura 3. 10 Caratula de un altímetro barométrico.....	35
Figura 3. 11 Configuración de un tubo de pitot.....	36
Figura 3. 12 Compás magnético.....	37
Figura 3. 13 Constelación de satélites.....	37

Capítulo 4

Figura 4.1 Sistema básico de Comunicaciones.....	39
Figura 4.2 Comunicación Infrarroja.....	41
Figura 4. 3 Categorías de redes inalámbricas.....	43
Figura 4.4 Comparativo de velocidades en estándares de redes inalámbricas.....	43
Figura 4.5 Clasificación de Clases de Conexiones en Bluetooth.....	44
Figura 4.6 Pila del protocolo IEEE 802.15.4 y Zigbee.....	49
Figura 4.7 Operación en modo transparente (half-dúplex).....	50
Figura 4.8 Canales de operación para ZigBee.....	51
Figura 4.9 Modulación QPSK.....	52
Figura 4. 10 Diferencias entre QPSK y O-QPSK y constelación para O-QPSK respectivamente.....	52
Figura 4.11 Modulación BPSK.....	53
Figura 4. 12 Diagrama de bloques de la tarjeta FMS.....	55
Figura 4.13 Trama de datos seriales con 8 bits, 1 bit inicio, 1 bit de paro y sin bit de paridad.....	56
Figura 4.14 Diagrama de bloques del módulo UART del TM4C123GH6PM.....	56
Figura 4.15 Transmisión serial asíncrona UART.....	57
Figura 4.16 Diagrama de bloques de una red de comunicación I ² C en el TM4C123GH6PM....	58
Figura 4.17 Condiciones de inicio y paro de comunicación I ² C.....	58
Figura 4.18 Trama de datos I2C.....	59

Capítulo 5

Figura 5.1 Diagrama de bloques del sistema de abordaje propuesto.	61
Figura 5.2 Launchpad TIVA de Texas Instruments.....	62
Figura 5.3 Tarjeta IMU Boosterpack TI.	63
Figura 5. 4 Diagrama de bloques tarjeta Booster Pack.....	64
Figura 5.5 Chip MPU-9150.	64
Figura 5.6 Orientación de ejes en el dispositivo. Invensense.....	65
Figura 5. 7 Diagrama de bloques del MPU 9150.....	66
Figura 5.8 Sensor de temperatura y presión BOSCH® BMP180.....	67
Figura 5. 9 Diagramas de flujo de obtención de datos y calcular mediciones compensadas. ...	68
Figura 5. 10 Circuito GPS.....	69
Figura 5. 11 Diagrama de flujo para el GPS.....	71
Figura 5. 12 Ubicación de elementos electrónicos a bordo del UAV.....	72
Figura 5. 13 Diagrama de bloques del control PD.....	73

Capítulo 6

Figura 6. 1 Bloques de comunicaciones entre el UAV y la estación Terrena.....	75
Figura 6. 2 Módulo XB24-B Series 2.....	76
Figura 6. 3 Adaptador Xbee USB.....	77
Figura 6. 4 Programa X-CTU detectando un módulo.....	77
Figura 6. 5 Dispositivo Coordinador. Figura 6. 6 Dispositivo final.	78
Figura 6. 7 Conexión básica entre dos módulos Zigbee®.	79
Figura 6. 8 Interfaz de comunicación entre la FMS y la PC.	80
Figura 6. 9 Configuración e inicialización del puerto serial por medio de LabView.....	81
Figura 6. 10 Sentencia de datos NMEA y ángulos de Euler.....	81
Figura 6. 11 Software de separación de datos GPS en LabView.....	82
Figura 6. 12 Datos recibidos de la tarjeta de abordaje.....	82
Figura 6. 13 Interfaz gráfica para la estación terrena.....	83

Capítulo 7

Figura 7. 1 UAV en vuelo.....	84
Figura 7. 2 Implementación y pruebas de transmisión-recepción de datos.....	84
Figura 7. 3 Prueba de indicador de actitud.....	86
Figura 7. 4 Estación terrena (GCS).....	87

Índice de Tablas

Capítulo 2

Tabla 2. 1 Clasificación por características del UAV	20
--	----

Capítulo 3

Tabla 3. 1 Características orbitales de las principales constelaciones satelitales	38
--	----

Capítulo 4

Tabla 4. 1 Denominación de las bandas de RF	40
Tabla 4. 2 Características de estándares físicos.....	47
Tabla 4. 3 Estándares inalámbricos.	53
Tabla 4. 4 Velocidades de transmisión I2C en el TM4C123GH6PM.....	57

Capítulo 5

Tabla 5. 1 Resolución de mediciones del MPU 9150.....	66
Tabla 5. 2 Descripción del enunciado GPRMC.....	70

Anexo D Certificaciones asociadas al proyecto de tesis.



The certificate is a rectangular document with a blue border. At the top left is the 'ID VERIFIED' logo, a circular emblem with 'ID' in the center and 'VERIFIED' around the perimeter. To its right, the text 'CERTIFICATE of ACHIEVEMENT' is written in blue, with 'of' in a smaller font. Below this, the name 'Rigoberto Rangel Vargas' is printed in a large, bold, black font. To the left of the name, the text reads: 'This is to certify that successfully completed and received a passing grade in AUTONAVX: Autonomous Navigation for Flying Robots'. The course name 'AUTONAVX: Autonomous Navigation for Flying Robots' is highlighted in red. Below this, it states: 'a course of study offered by TUMx, an online learning initiative of Technische Universität München through edX.' At the bottom left, it says 'VERIFIED CERTIFICATE Issued July 3rd 2014'. At the bottom right, there are two signatures: 'Daniel Cremers' and 'Dr. Jürgen Sturm'. Below the signatures are their titles and affiliations: 'Prof. Dr. Daniel Cremers, Professor for Computer Science and Mathematics, Chair for Computer Vision & Pattern Recognition, Department of Informatics, TUM' and 'Dr. Jürgen Sturm, Postdoctoral Researcher, Chair for Computer Vision & Pattern Recognition, Department of Informatics, TUM'. In the bottom right corner, the 'edX' logo and the 'TUM' logo (Technische Universität München) are displayed. A large, faint watermark of a laurel wreath is visible in the background.

ID VERIFIED

CERTIFICATE of ACHIEVEMENT

Rigoberto Rangel Vargas

This is to certify that successfully completed and received a passing grade in **AUTONAVX: Autonomous Navigation for Flying Robots** a course of study offered by TUMx, an online learning initiative of Technische Universität München through edX.

VERIFIED CERTIFICATE
Issued July 3rd 2014

Verify the authenticity of this certificate at
<https://verify.edx.org/cert/717096f6ac046f6a5cadbae1f545ed3>

Daniel Cremers
Prof. Dr. Daniel Cremers
Professor for Computer Science and Mathematics
Chair for Computer Vision & Pattern Recognition
Department of Informatics, TUM

Jürgen Sturm
Dr. Jürgen Sturm
Postdoctoral Researcher
Chair for Computer Vision & Pattern Recognition
Department of Informatics, TUM

edX
Technische Universität München

TUM



CERTIFICATE of ACHIEVEMENT



THE UNIVERSITY of TEXAS SYSTEM

Steven Mintz

Steven Mintz

Executive Director, Institute for Transformational Learning
The University of Texas System

Harrison Keller

Harrison Keller

Vice Provost for Higher Education Policy and Research and
Executive Director, Center for Teaching and Learning
The University of Texas at Austin

Jonathan W. Valvano

Dr. Jonathan W. Valvano

Professor, Engineering Foundation Centennial Teaching
Fellow in Electrical Engineering No. 1
The University of Texas at Austin

Dr. Ramesh Yerraballi

Dr. Ramesh Yerraballi

Senior Lecturer
The University of Texas at Austin

This is to certify that

Rigoberto Rangel Vargas

successfully completed and received a passing grade in

6.02x: Embedded Systems - Shape the World

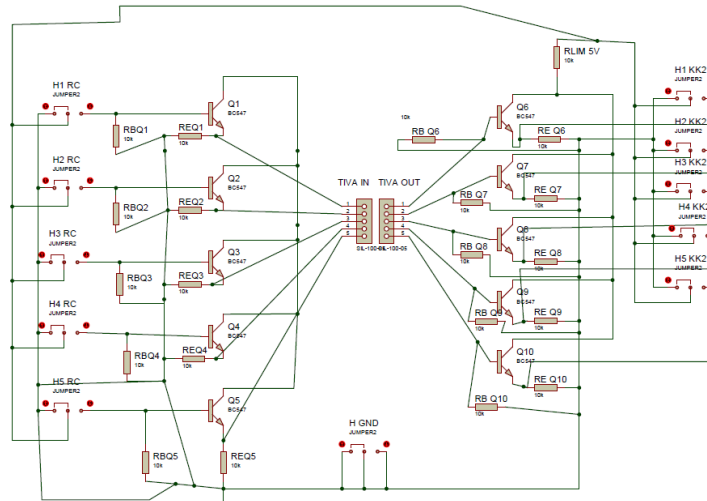
a course of study offered by UTAustinX, an online learning initiative of The University of Texas System through edX.

VERIFIED CERTIFICATE
Issued May 14, 2015

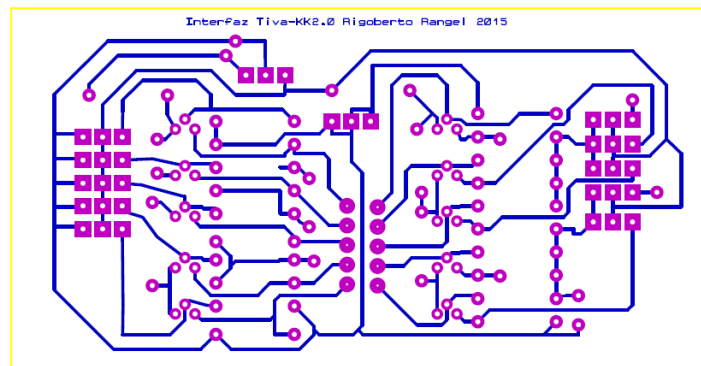
Verify the authenticity of this certificate at
<https://verify.edx.org/cent/7e282bc08e994d15a97d25c8d17ed9b>

Anexo E Interfaz Radio Control-Tiva-KK2.0

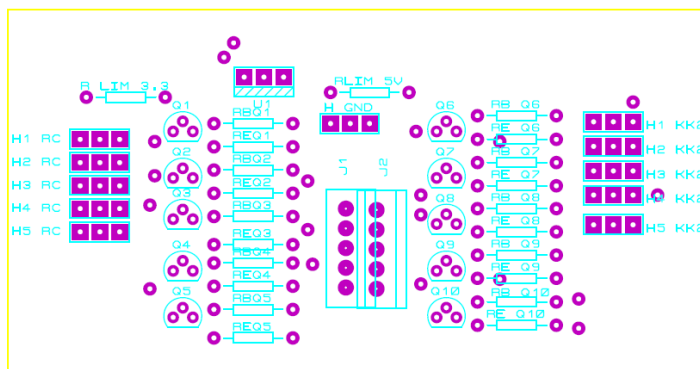
Diagrama esquemático



PCB Interfaz



Superior Interfaz



Anexo F Tarjeta de estabilización KK2.0

