



**DESARROLLO DE UN DRON INTELIGENTE
PARA SER UTILIZADO COMO HERRAMIENTA
DE VIGILANCIA EN LA SEGURIDAD
PÚBLICA Y PRIVADA, BASADO
EN LA PLATAFORMA ARDUINO.**

Autor: Mario Pontillo
C. I.: 24.553.938

Urb. Yuma II, calle n° 3. Municipio San Diego
Teléfono (0241) 8714240 (master)-Fax: (0241) 8712394



**REPÚBLICA BOLIVARIANA DE VENEZUELA
UNIVERSIDAD JOSÉ ANTONIO PÁEZ
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA ELECTRÓNICA**

**DESARROLLO DE UN DRON INTELIGENTE PARA SER
UTILIZADO COMO HERRAMIENTA DE VIGILANCIA EN LA
SEGURIDAD PÚBLICA Y PRIVADA BASADO EN LA
PLATAFORMA ARDUINO.**

**Trabajo de grado presentado como requisito para optar al título de Ingeniero
Electrónico**

Autor: Pontillo Mario

C. I.: 24.553.938

Tutor: Ing. Dinorah Giménez

San Diego, marzo de 2018



**REPÚBLICA BOLIVARIANA DE VENEZUELA
UNIVERSIDAD JOSÉ ANTONIO PÁEZ
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA ELECTRÓNICA**

ACEPTACIÓN DEL TUTOR

Quien suscribe, Ingeniero Dinorah Giménez portador de la cédula de identidad N° 4.071.295, en mi carácter de tutor del trabajo de grado presentado por el ciudadano Pontillo Avolio Mario Alberto, portador de la cédula de identidad N° 24.553.938, titulado **DESARROLLO DE UN DRON INTELIGENTE PARA SER UTILIZADO COMO HERRAMIENTA DE VIGILANCIA EN LA SEGURIDAD PÚBLICA Y PRIVADA BASADO EN LA PLATAFORMA ARDUINO**. Presentado como requisito parcial para optar al título de Ingeniero Electrónico, considero que dicho trabajo reúne los requisitos y méritos suficientes para ser sometido a la presentación pública y evaluación por parte del jurado examinador que se designe.

En San Diego, a los veintidós (22) días del mes de marzo del año dos mil dieciocho.

Ing. Dinorah Giménez.
C.I.: 4.071.295

AGRADECIMIENTOS

Agradezco principalmente a Dios, por darme esta vida y fortaleza para poder concluir la carrera, a mi querida familia que tanto apoyo incondicional me ha brindado en el transcurso de mis estudios. A mis amigos que hicieron de esta carrera una gran experiencia de vida.

Agradezco a Jesús Arocha, Wilfredo Mendoza, José Rey, Emmanuel Nieto, Whady Martínez y todos los demás que siempre estuvieron atentos en el desarrollo de este proyecto.

Por ultimo agradezco a mi estimada tutora Prof. Dinorah Giménez, es usted una excelente profesional que siempre fue guía en este proyecto y en la vida, agradezco también a todos los profesores que fueron de gran ayuda en la elaboración de este proyecto y a la Universidad José Antonio Páez.

Mario Pontillo

DEDICATORIA

Principalmente a Dios, por darme la oportunidad de vivir y por estar conmigo en cada paso que doy, por fortalecer mi corazón e iluminar mi mente y por haber puesto en mi camino a aquellas grandes personas que han sido mi soporte y compañía durante todo el periodo de estudio.

A mis padres por ser el pilar fundamental en todo lo que soy, en toda mi educación, tanto académica, como de la vida, por su incondicional apoyo perfectamente mantenido a través del tiempo.

Mario Pontillo

ÍNDICE

CONTENIDO	Pág.
ÍNDICE DE TABLAS	xii
ÍNDICE DE FIGURAS	xiii
RESUMEN	xvii
INTRODUCCIÓN	1
CAPÍTULO	
I EL PROBLEMA	
1.1 Planteamiento del Problema.....	3
1.2 Formulación del Problema.....	4
1.3 Objetivos de la Investigación.....	5
1.3.1 Objetivo General.....	5
1.3.2 Objetivos Específicos.....	5
1.4 Justificación.....	5
1.5 Alcance.....	6
II MARCO TEÓRICO	
2.1 Antecedentes.....	7
2.2 Bases Teóricas.....	9
2.2.1 Dron.....	9
2.2.2 Partes básicas de un dron inteligente.....	10
2.2.2.1 Parte mecánica de un dron cuadricóptero.....	10
2.2.2.2 Parte Electrónica de un dron Quadcopter.....	12
2.2.2.3 Software básico de un dron.....	39
2.2.3 funcionamiento y movimientos de un dron.....	39
2.2.3.1 Control de altitud.....	41
2.2.3.2 Control de Yaw.....	41

2.2.3.3 Control del Roll.....	42
2.2.3.4 Control del Pitch.....	43
2.2.4 Componentes de sistemas básicos que pueden implementarse en un dron.....	43
2.2.4.1 Sensores de movimiento y/o Presencia.....	43
2.2.4.2 Cámaras de media – alta definición con visión nocturna..	46
2.2.5 Aplicación Web.....	47
2.2.5.1 Lenguajes de programación y diseño de una aplicación Web.....	47
2.2.6 Comunicación bluetooth.....	48
2.3 Definición de Términos.....	49

III MARCO METODOLÓGICO

3.1 Tipo de Investigación.....	51
3.2 Diseño de la Investigación.....	51
3.3 Nivel de la Investigación.....	52
3.4 Población y Muestra.....	52
3.5 Técnicas e Instrumentos de Recolección de Datos.....	53
3.6 Validez.....	53
3.7 Confiabilidad.....	53
3.8 Técnica de análisis de datos.....	56
3.9 Fases Metodológicas.....	57

VI RESULTADOS

4.1 Fase I. Investigar las tecnologías actuales referente a drones.....	59
4.1.1 CPU.....	60
4.1.2 Sensorica.....	60
4.1.3 Sistema de propulsión.....	64

4.1.4 Sistema de Comunicación.....	68
4.2 Fase II. Diseñar la estructura, hardware y software del prototipo.....	70
4.2.1 Diseño de la estructura.....	74
4.2.1.1 Chasis.....	75
4.2.1.2 Hélices y porta hélices.....	76
4.2.2 Diseño del Hardware.....	76
4.2.2.1 Diseño del sistema digital.....	80
4.2.2.2 Diseño del sistema de potencia.....	83
4.2.2.3 Configuración Final del hardware.....	87
4.2.3 Diseño del software.....	88
4.2.3.1 Entorno de programación.....	87
4.2.3.2 Diseño del programa en Arduino del Dron.....	90
4.2.3.3 Modulo IMU.....	91
4.2.3.4 Modulo Barómetro.....	93
4.2.3.5 Modulo GPS.....	94
4.2.3.6 Modulo Magnetómetro.....	97
4.2.3.7 Modulo PIRAlarm.....	98
4.2.3.8 Modulo MODULOBT.....	97
4.2.3.9 Modulo Motores.....	100
4.2.3.10 Modulo PID.....	102
4.2.3.11 Modulo Principal.....	106
4.3 Fase III. Desarrollar una interfaz de control y supervisión.....	107
4.4 Fase IV. Implementar el dron inteligente diseñado.....	109
4.4.1 Ensayo con 1 grado de libertad.....	114
4.4.2 Ensayo con 3 grados de Libertad.....	115
4.4.3 Ensayo con 4 grados de libertad.....	117
4.4.4 Montaje final del trabajo de investigación.....	119

CONCLUSIONES.....	125
RECOMENDACIONES.....	127
REFERENCIAS BIBLIOGRÁFICAS.....	128 25
ANEXOS.....	131

ÍNDICE DE TABLAS

CONTENIDO

FIGURA		Pág.
1	Distribución de las respuestas del cuestionario.....	55
2	Interpretación de magnitud del coeficiente de confiabilidad.....	56

ÍNDICE DE FIGURAS

FIGURA	CONTENIDO	Pág.
1	Fibra de carbono.....	11
2	Hélice normal e inversa.....	12
3	Porta hélices de 3 mm.....	12
4	Raspberry Pi modelo 2.....	13
5	Arduino Uno R3.....	15
6	Arduino Nano.....	15
7	Circuito integrado MEMS.....	17
8	Composición mecánica de un acelerómetro.....	19
9	Estructura MEMS de un Acelerómetro.	20
10	DCL del sensor inclinado en Y con descomposición del vector aceleración de gravedad en 2D.....	21
11	proyección de ángulo del sensor inclinado en distintos ejes en 3D.....	22
12	Composición mecánica de un giroscopio.....	24
13	Estructura MEMS de un Giroscopio.....	24
14	Ángulos de navegación.....	25
15	Representación de un sensor Ultrasónico.....	28
16	Método de trilateración de un GPS.....	30
17	Rutas Programadas.....	31
18	Baterías LiPo.....	32
19	Batería de LiPo con sus respectivas características.	32
20	ESC.....	34
21	Motor de corriente continua sin escobillas o BLDC.....	35
22	Motores DC brushed de dron Syma.....	36
23	Motor Brushless Inrunner.	37
24	Motor Brushless Outrunner.....	38
25	Disposición del Quadcopter.	40

26	Rotaciones del Quadcopter.....	40
27	Control de Altitud.....	41
28	Control del Yaw.....	42
29	Control del Roll.....	42
30	Control del Pitch	43
31	PIR.....	44
32	Campos de medición del sensor IR	45
33	Rango de Alcance IR	45
34	Pi NoiR camera.....	46
35	MPU 6050	61
36	HMC 5883L	62
37	BMP180	62
38	GPS Ublox NEO 6M	63
39	Motor EMAX CF2822	65
40	Características Motor EMAX CF2822	65
41	ESC EMAX 20 A	66
42	Batería de LiPo Turnigy de 2.2 A.....	68
43	Batería de LiPo Turnigy de 2.2 A.....	68
44	Bluetooth Stick HC – 06	69
45	Chasis Frame Kit 450	75
46	Dimensiones Chasis Frame Kit 450	76
47	Rango de detección HC-SR501	78
48	PIR HC-SR501	79
49	Diagrama de conexión digital en Fritzing.....	86
50	Diagrama de conexión de potencia en Fritzing.....	82
51	Esquema circuito medidor de batería en Proteus 8.....	85
52	Esquema electrónico del Hardware completo en Proteus 8	87
53	Diagrama de bloques generalizado del Hardware.....	89

54	Funciones principales en IDE Arduino.....	88
55	Módulos del programa en IDE Arduino.....	90
56	Árbol y jerarquía de clases del programa.....	93
57	Código Modulo IMU en IDE Arduino.....	94
58	Código Modulo Barómetro en IDE Arduino.....	95
59	Código Modulo GPS en IDE Arduino.....	96
60	Código Modulo Magnetómetro en IDE Arduino.....	97
61	Código Modulo PIRAlarm en IDE Arduino.....	98
62	Código Modulo MODULOBT en IDE Arduino.....	99
63	Código Modulo Motores en IDE Arduino.....	101
64	Diagrama de bloques generalizado del controlador en un solo eje de navegación	104
65	Diagrama de bloques generalizado del controlador.....	104
66	Matriz de comandos.....	105
67	Código Modulo PID en IDE Arduino	105
68	Código Modulo Principal en IDE Arduino	105
69	Pantalla inicio de sesión.....	109
70	Interfaz Principal.....	110
71	Código de programación en MIT App Inventor.....	111
72	Cuadro.....	112
73	Paleta Abierta.....	113
74	Prototipo diseñado para el primer ensayo.....	114
75	Prototipo diseñado para el segundo ensayo.....	116
76	Prototipo diseñado para el segundo ensayo.....	116
77	Prototipo diseñado para el tercer ensayo	118
78	Monitoreo de datos del tercer ensayo.....	119
79	Estructura ensamblada con ESCs Soldados y conectados para el nuevo prototipo.	120
80	Estructura ensamblada con ESCs Soldados y conectados para el nuevo prototipo.....	117

81	PCB Tarjeta principal del prototipo diseñado en Proteus 8.....	118
82	PCB Tarjeta circuito medidor de carga del prototipo diseñado en Proteus 8.....	119
83	Dron Completamente Armado.....	120
84	Dron Completamente Armado.....	121



**REPÚBLICA BOLIVARIANA DE VENEZUELA
UNIVERSIDAD JOSÉ ANTONIO PÁEZ
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA ELECTRÓNICA**

**DESARROLLO DE UN DRON INTELIGENTE PARA SER UTILIZADO
COMO HERRAMIENTA DE VIGILANCIA EN LA SEGURIDAD PÚBLICA
Y PRIVADA BASADO EN LA PLATAFORMA ARDUINO.**

Autor: Pontillo Mario

Tutor: Dinorah Giménez

Fecha: marzo, 2018

RESUMEN

El presente trabajo está tipificado como un proyecto factible, desarrollado en base a un diseño de campo y de tipo descriptivo, cuyo objetivo general es el desarrollo de un dron inteligente para ser utilizado como herramienta de vigilancia en la seguridad pública y privada. El sistema se basa en una topología centralizada, cuyo control y automatización son dados y supervisados por medio de una aplicación Android de forma remota por medio de comunicación bluetooth. El dron no solo se mueve manualmente controlado por el usuario, si no que este también se mueve automáticamente con solo colocar las coordenadas y la altura deseada, para que el sistema pueda vigilar zonas autónomamente sin la necesidad del continuo manejo del usuario. Como elemento maestro se usó el microcontrolador Arduino, el cual procesa todos los de datos recibidos por los sensores, acciona los elementos finales de control, por medio de un control PID diseñado. La información del entorno adquirida por el dron es enviada al usuario mediante un módulo de bluetooth conectado al Arduino y recibida por una aplicación Android diseñada en MIT App Inventor 2. Este trabajo de investigación tuvo como propósito el desarrollo de un dron inteligente desde cero que tuviese cierto grado de autonomía y un sistema de seguridad básico para que fuese utilizado principalmente en la vigilancia.

Descriptores: Dron, Aplicación Android, Autonomía, Vigilancia.

INTRODUCCIÓN

De acuerdo con 3D Robotics, en estos últimos años, el desarrollo de vehículos no tripulados ha tenido un avance muy significativo, sobre todo para el uso en aplicaciones civiles. Las aplicaciones usuales de este tipo de vehículo se centran en tareas donde tienen entornos de difícil acceso o que representa algún peligro, como por ejemplo: tareas de vigilancia y seguimiento, reconocimiento geográfico, otros.

El desarrollo de estos sistemas ha sido posible gracias al desarrollo de microcontroladores, que ofrecen la posibilidad de hacer cálculos muy complejos en un chip muy pequeño, la miniaturización de los sensores (MEMS – Micro Electromechanical Systems), así como las mejoras en los sistemas de almacenamiento de energía.

El objetivo de este proyecto es diseñar e implementar un Quadcopter pilotado desde un dispositivo remoto por medio de una aplicación web. El Quadcopter dispone de una cámara que puede hacer fotografías y enviarlas a un dispositivo de almacenamiento y un GPS que permita la localización y el posicionamiento de este en el espacio a voluntad del usuario. La idea principal de este proyecto es conseguir desarrollar una plataforma que en el futuro pueda seguir ampliándose.

Una de las motivaciones principales es poder realizar un proyecto multidisciplinario donde interviene el diseño de hardware y software, sabiendo desde un principio que este sería un proyecto ambicioso y de gran dificultad técnica.

Para la investigación se definen cuatro capítulos, los cuales seccionan ordenadamente y se presenta de la siguiente manera:

Capítulo I. El Problema. En él se realiza el planteamiento del problema, el propósito de la investigación, los objetivos que se buscan alcanzar en el desarrollo del proyecto, la justificación de lo que se obtendrá con la aplicación de filtros activos de potencia, el alcance y limitaciones.

Capítulo II. Marco teórico. Se describieron los antecedentes, las bases teóricas y la definición de los términos básicos referentes a la investigación.

Capítulo III. Marco metodológico. Se planteó todo lo referente al proceso de investigación, definiendo el tipo, diseño y nivel de investigación, así como la población y la muestra para la recolección de datos. Igualmente se explican cada una de las fases a realizar durante el desarrollo del trabajo.

Capítulo IV. Resultados. Se presentan los resultados obtenidos por el logro de los objetivos específicos que se plantearon en el trabajo de grado.

CAPITULO I

EL PROBLEMA

1.1 Planteamiento del problema

Antes de comenzar a hablar del problema en el cual este proyecto se va a enfocar y para un mayor entendimiento de este trabajo de investigación, primero se necesita saber que es un dron y algunas de sus características básicas. Según la revista digital de ciencia de la Universidad Autónoma de Nuevo León (UANL) (2013).

Los drones son vehículos aéreos no tripulados que surgen de la tecnología militar hace unas cuantas décadas, entre principios del año 1917 y finales de 1918, sobre todo a partir de prototipos diseñados a raíz de la Segunda Guerra Mundial.

Estos son conocidos técnicamente por sus siglas VANT (Vehículo Aéreo No Tripulado) que es una traducción del inglés UAV (Unmanned Aerial Vehicle). La palabra drone, que significa zumbido o zángano en inglés, se ha utilizado como denominación coloquial de estos vehículos y ha sido aceptada recientemente en español la denominación dron y su plural drones.

En un principio, los drones eran vehículos similares en su forma a aviones o planeadores, pero han ido evolucionando a nuevos diseños aerodinámicos dependiendo de las funciones para los que son creados. Son vehículos aéreos a los que se pueden incorporar múltiples sistemas de medición, vigilancia o comunicación y dirigirse de manera remota a través de un sencillo sistema de navegación. Pueden alcanzar con gran exactitud objetivos o estar dotados de grandes sistemas de vigilancia y conectada vía remota a cualquier lugar del mundo, donde se reciben las imágenes captadas y se guía su posición.

Los drones son una de las tecnologías que más destacan y llaman la atención en estos últimos años, ya que son unos dispositivos muy versátiles y relativamente económicos. Pueden ser usados en infinidad de tareas que el humano no puede realizar porque son demasiado peligrosas como la exploración o la limpieza de residuos tóxicos.

Por otro lado, estos dispositivos pueden ser utilizados también para fines de vigilancia o de seguridad civil y organizacional, inclusive ya hay países como Italia, Estados Unidos, Alemania y Japón, donde utilizan drones en sus fuerzas policiales y organizaciones de seguridad. Es esta última función en la que se enfoca este proyecto, ya que además de ser en todas partes del mundo el factor más buscado por las personas y organizaciones para poder establecerse y sentir estabilidad, es una de las mayores necesidades en Venezuela.

Para resolver la dificultad que se presenta en el entorno en cuanto a la vigilancia y situación en tiempo real de la condición de una zona determinada, se desarrolló un prototipo de dron que por medio de una aplicación y de sensorica, este pueda reconocer el entorno en el que se encuentra para brindar una mayor seguridad y alivio a las personas y organizaciones.

En Venezuela existen algunas empresas que venden, alquilan y ofrecen servicio técnico de drones, a pesar de esto no son tan usados debido a los altos costos, los cuales se originan en su mayor parte a que estas empresas utilizan drones importados. Sin embargo, dichas empresas no ofrecen drones con las características anteriormente mencionadas, es decir, drones dedicados a la vigilancia, sino que son mayormente dedicados al entretenimiento.

1.2 Formulación del problema

Sabiendo y entendiendo ya el planteamiento inicial y el enfoque del mismo, se puede entonces comenzar el desarrollo del presente proyecto formulando así la siguiente pregunta.

¿Qué dispositivo se puede desarrollar que ayude a la vigilancia en tiempo real para la seguridad de las personas?

1.3 Objetivos de la investigación

1.3.1 Objetivo General

Desarrollar un dron inteligente para ser utilizado como herramienta de vigilancia en la seguridad pública y privada, basado en la plataforma Arduino.

1.3.2 Objetivos Específicos

- Investigar las tecnologías actuales referentes a drones.
- Diseñar la estructura, hardware y software del prototipo.
- Desarrollar una interfaz de control y supervisión del Dron para los usuarios en una aplicación Android.
- Implementar el dron inteligente diseñado.

1.4 Justificación de la investigación

Tal como refiere Salcedo-Bastardo (2004):

Desde los primeros tiempos de la agricultura o desde fines de la Edad del Hierro, la cultura humana ha contado con tecnología, es decir, ha contado con la capacidad de modificar la naturaleza en un grado u otro. Se considera que la tecnología proporciona estimables beneficios a la sociedad a corto y largo plazo. Venezuela ha estado rezagada en el desarrollo de ciencia y tecnología a lo largo de su historia. Es recién en el siglo XX que comienza una inversión perceptible aunque aún deficiente en la creación de tecnología propia (p.171).

El propósito de este trabajo de investigación es el desarrollo de un dron basado en la plataforma Arduino que sea usado como herramienta de vigilancia para el sector público y privado en Venezuela; además, que disponga de la capacidad de reconocer su entorno, de comunicar y reportar en tiempo real lo que sucede en él, agregándole valor tecnológico al país, fomentando y ayudando al crecimiento científico de este.

1.5 Alcance

El alcance de proyecto incluye todo el desarrollo necesario para tener el Quadcopter volando y controlado por medio de una aplicación Android vía bluetooth, el cual es:

- ✓ Diseño y Construcción de la Estructura.
- ✓ Diseño de los algoritmos de Control.
- ✓ Diseño y Construcción de la placa electrónica.
- ✓ Integración de la aplicación para Android.
- ✓ Integración y configuración de los sensores.
- ✓ Integración de la comunicación bluetooth.
- ✓ Integración de los motores.
- ✓ Con este trabajo de investigación se pretende implementar el prototipo y mejorar la forma de vigilancia en los diferentes sectores del país.

CAPÍTULO II

MARCO TEÓRICO

Para llevar a cabo esta investigación, se ha realizado una búsqueda y revisión de trabajos escritos anteriormente, consultando a una serie de investigaciones relacionadas con el tema que servirán como base y orientación para la elaboración del mismo.

2.1 Antecedentes de la Investigación

Juan Carmona Fernández (2013), en el trabajo realizado titulado “Diseño de un sistema de control para un cuadricóptero” para optar por el título de Ingeniero electrónico y automática en la Universidad Carlos III de Madrid (España), presenta el desarrollo de un sistema de navegación autónomo partiendo de un Ardrone 2.0 (nombre del modelo de un dron) diseñado por la empresa francesa Parrot, en el marco de un proyecto factible con apoyo documental. Basado en un estado del arte en donde presentan el tipo de control que el proyecto empleara, las ventajas e importancia del uso de la cámara como sensor y el programa con el que será desarrollado el mismo, posteriormente explica el tipo de control seleccionado, que en este caso será por visión (odometría). Utilizando su cámara frontal de alta definición, debido a que su sistema de navegación autónomo estará basado en odometría como se ha mencionado anteriormente, que significa que reconocerá su entorno por medio de la comparación de una fotografía con respecto a las imágenes que la cámara del ardrone está recolectando. Luego explica el sistema seleccionado con el que desarrollará el programa de control, en su caso el sistema operativo seleccionado es el ROS, especial para la programación orientada a la robótica. Este proyecto aporta un estilo de control autónomo donde la cámara no solo es utilizada para tomar imágenes y videos, sino como un sensor que sirva para reconocer el entorno, esto puede ser fundamental para

que un dron lleve a cabo acciones de vigilancia. También aporta nombres y explicación de programas y librerías útiles y eficientes para el desarrollo del proyecto que pueden servir en un dron del tipo que este proyecto desea desarrollar.

Sergio Moyano Díaz (2014) en el trabajo realizado titulado “Diseño y construcción de un Quadcopter”, presento una metodología de desarrollo ágil. El propósito principal fue desarrollar un proyecto multidisciplinar donde intervinieran el diseño de hardware y software, como es el diseño y la construcción de un Quadcopter, presentando una memoria que está dividida en tres grandes bloques. El primer bloque se centra en la parte mecánica, donde se detalla el proceso de diseño, los componentes mecánicos y su funcionalidad. También se trata la selección del material teniendo en cuenta la resistencia mecánica, peso y precio de los materiales. El segundo bloque se centra en la parte electrónica, donde se detalla todos los componentes que se han usado y la razón de su uso. Con esto se refiere a las baterías, ESC, Arduino, la placa de electrónica, los motores y los sensores, la cámara y el Bluetooth. El tercer bloque se detalla todo el software realizado; es decir, cada una de librerías implementadas en Arduino para usar cada componente, y la librería que contiene el algoritmo de control. Además se trata la aplicación que controlará el Quadcopter. Los resultados obtenidos fueron bastante favorables, el diseño de hardware fue un éxito, se consiguió integrar todos los componentes: microcontrolador, Cámara, Bluetooth, sensores y motores. El diseño está bastante bien elaborado para que en caso de avería de algunos de los componentes sea fácilmente reemplazable. En cuanto al diseño de software también fue un éxito, se diseñó una librería para cada dispositivo, otorgando alta independencia entre clases. En cuanto al control hay que destacar que los parámetros encontrados son muy experimentales y varían en función del tipo de vuelo que se le quiera otorgar, y requieren de ciertos ajustes dependiendo del sitio donde se quiera utilizar.

El principal aporte que este proyecto presenta son los distintos elementos que se utilizan en el momento de la construcción de un dron, mostrando los más aptos de

bajo costo que hay en el mercado y las características de estos, por cuales optar dependiendo del prototipo que se vaya a construir, como también la teoría necesaria a saber del comportamiento de un dron. También este proyecto aporta las distintas librerías de Arduino y software que pueden utilizarse para un funcional y óptimo control de vuelo. Como también distintas alternativas en las que hace una comparación para luego elegir la más factible.

Elías Rodríguez Martín (2015) en el trabajo realizado titulado “Sistema de posicionamiento para un dron” para optar por el título de Ingeniero informático en la Universidad de La Laguna (España), en el marco de un proyecto factible con apoyo documental. El presente trabajo trata sobre el desarrollo de un sistema de posicionamiento basado en odometría visual para un dron que vaya a utilizarse en interiores o en lugares en donde no se dispone señal GPS y mantener a un cuadricóptero volando lo más “quieto” y estable posible, y corregir en tiempo real cualquier desplazamiento o rotación producida por inestabilidades inherentes al sistema de sustentación. Los resultados obtenidos se han situado por encima de las expectativas, aunque se concluyó que es un sistema con muchas limitaciones y que en muchos casos no era viable. El aporte que este trabajo de investigación presenta son los sistemas de posicionamiento que existen, al igual que las teorías involucradas, las ventajas y desventajas que estos presentan y los más viables a utilizar dependiendo en donde el prototipo trabajará, si en exteriores, interiores, entre otros.

2.2 Bases Teóricas

A continuación, se describirán las bases teóricas sobre los temas necesarios para comprender y realizar exitosamente el presente trabajo de grado.

2.2.1 Dron

Según Ernesto Martínez de Carvajal Hedrich, Piloto de RPAS (2015), un dron es: Un vehículo aéreo no tripulado (VANT), UAV por sus siglas en inglés

(Unmanned Aerial Vehicle) o dron es una aeronave que vuela sin tripulación. Aunque hay VANT de uso civil, también son usados en aplicaciones militares, donde son denominados vehículo aéreo de combate no tripulado UCAV por su nombre en inglés. Para distinguir los VANT de los misiles, un VANT se define como un vehículo sin tripulación reutilizable, capaz de mantener de manera autónoma un nivel de vuelo controlado y sostenido, y propulsado por un motor de explosión o de reacción. Por tanto, los misiles de crucero no son considerados VANT porque, como la mayoría de los misiles, el propio vehículo es un arma que no se puede reutilizar, a pesar de que también es no tripulado y en algunos casos guiado remotamente.

Un artículo web del periódico El Economista (2012, Concha Raso), menciona que históricamente los UAV eran simplemente aviones pilotados remotamente (en inglés: drones), pero cada vez más se está empleando el control autónomo de los UAV. En este sentido se han creado dos variantes: algunos son controlados desde una ubicación remota, y otros vuelan de forma autónoma sobre la base de planes de vuelo pre programados usando sistemas más complejos de automatización dinámica.

2.2.2 Partes básicas de un dron inteligente

Un dron inteligente está compuesto por tres partes básicas: la mecánica, electrónica y la del software.

2.2.2.1 Parte mecánica de un dron cuadricóptero

En un dron se tiene que en su parte mecánica se encuentra el chasis y las hélices. A continuación, se describe cada una de las partes mecánicas.

El chasis de un dron Quadcopter dependiendo de la distribución que se elija debe ser en forma de equis (x) o cruz (+) y tienen que ser totalmente simétricos y balanceados para así no crear perturbación y desequilibrio en el sistema, ya que si no fuera el caso todo sería más complicado debido a que habría que ajustar el sistema y crear un algoritmo de control que contrarreste dichas perturbaciones. Este, debe ser lo más ligero y resistente posible, ósea que tenga un buen coeficiente de peso/rigidez; un

ejemplo de este, es la fibra de carbono que es una fibra constituida por finos filamentos de 5-

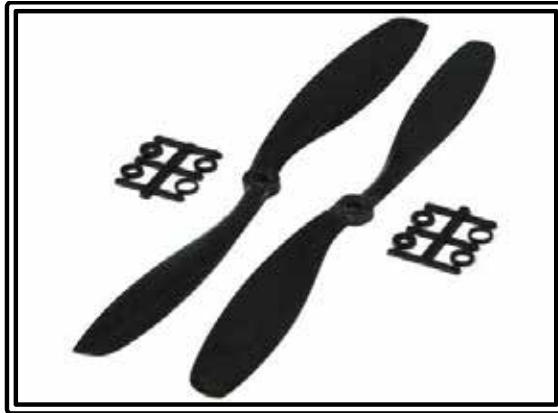


Figura 2. Hélice normal e inversa

Fuente: TDG Sergio Moyano Díaz (2014, p.38)



Figura 3. Porta hélices de 3 mm

Fuente: TDG Sergio Moyano Díaz (2014, p.38)

2.2.2.2 Parte electrónica de un dron Quadcopter

Comúnmente en un dron Quadcopter se tiene: el controlador de vuelo, los sensores, los ESC (electronic speed controller) o drivers, los motores y la alimentación.

El controlador de vuelo es el dispositivo encargado de administrar, ordenar, dirigir o regular el comportamiento del dron, con el fin de reducir las probabilidades de fallo y obtener los resultados deseados. Entre estos dispositivos de control se pueden encontrar:

- Raspberry Pi
- Arduino 1

Raspberry Pi que es un mini ordenador de placa reducida de bajo costo. Este mini ordenador tiene la capacidad de instalar sistemas operativos libres, lo que permite que pueda ser utilizado en proyectos de presupuesto limitado. El Raspberry Pi puede ser programado como servidor y central de control. Este incluye un System-on-a-chip Broadcom BCM2835, que contiene un procesador central (CPU) ARM1176JZF-S a 900 MHz, un procesador gráfico (GPU) VideoCore IV, y 1GB de memoria RAM (aunque originalmente al ser lanzado eran 256 MB). El diseño no incluye un disco duro ni unidad de estado sólido, ya que usa una tarjeta SD para el almacenamiento permanente; tampoco incluye fuente de alimentación ni carcasa (ver figura 4).



Figura 4. Raspberry Pi Modelo 2

Fuente: www.raspberrypi.org (2018)

La fundación Raspberry Pi da soporte para las descargas de las distribuciones para arquitectura ARM, Raspbian (derivada de Debian), RISC OS 5, Arch Linux ARM (derivado de Arch Linux) y Pidora (derivado de Fedora); y promueve

principalmente el aprendizaje del lenguaje de programación Python. Otros lenguajes también soportados, son Tiny BASIC, C, Perl5 y Ruby.

El Arduino, es una plataforma de hardware libre. Su principal componente es un microcontrolador con entorno de desarrollo libre que facilita el control de elementos actuadores presentes en un proyecto, ya que permite realizar operaciones de control de una manera sencilla y fácil de entender. Conectándola como esclavo a la Raspberry Pi permite un amplio control de actuadores sin las limitaciones que una sola placa de esta pueda tener.

El hardware consiste en una placa con un microcontrolador Atmel AVR y puertos de entrada/salida. Los microcontroladores más usados son el Atmega168, Atmega328, Atmega1280 y Atmega8 por su sencillez y bajo costo, que permiten el de múltiples diseños. Por otro lado, el software consiste en un entorno de desarrollo que implementa el lenguaje de programación Processing/Wiring y el cargador de arranque que es ejecutado en la placa.

Desde octubre de 2012, Arduino se utiliza también con microcontroladores CortexM3 de ARM de 32 bits, que coexistirán con las más limitadas, pero también económicas AVR de 8 bits. ARM y AVR no son plataformas compatibles a nivel binario, pero se pueden programar con el mismo IDE (ambiente de desarrollo integrado) de Arduino y hacerse programas que compilen sin cambios en las dos plataformas. Eso sí, los microcontroladores CortexM3 usan 3,3V, a diferencia de la mayoría de las placas con AVR, que generalmente usan 5V. Sin embargo, ya anteriormente se lanzaron placas Arduino con Atmel AVR a 3,3V como la Arduino Fio y existen compatibles de Arduino Nano y Pro como Meduino en que se puede conmutar el voltaje (ver figura 5 y 6).



Figura 5. Arduino Uno R3

Fuente: www.arduino.cc (2018)

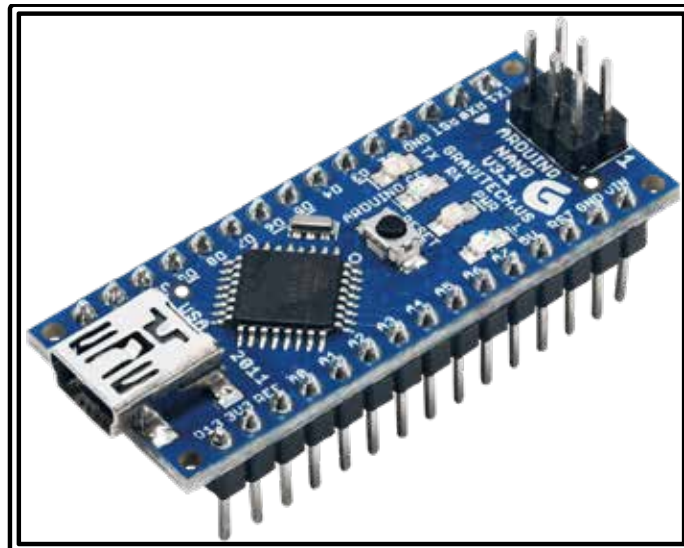


Figura 6: Arduino nano

Fuente: www.arduino.cc (2018)

Arduino se puede utilizar para desarrollar objetos interactivos autónomos o puede ser conectado a software tal como Adobe Flash, Processing, Max/MSP, Pure Data. El entorno de desarrollo integrado se puede descargar gratuitamente.

Este dispositivo puede tomar información del entorno a través de sus entradas analógicas y digitales, puede controlar luces, motores y otros actuadores. El microcontrolador en la placa Arduino se programa mediante el lenguaje de programación Arduino (basado en Wiring) y el entorno de desarrollo Arduino (basado en Processing). Los proyectos hechos con Arduino pueden ejecutarse sin necesidad de conectar a un computador.

También cuenta con su propio software que se puede descargar de su página oficial que ya incluye los drivers de todas las tarjetas disponibles lo que hace más fácil la carga de códigos desde el computador.

El proyecto Arduino recibió una mención honorífica en la categoría de Comunidad Digital en el Prix Ars Electrónica de 2006. Arduino posee una innumerable cantidad de librerías, que permiten el uso de ciertos módulos y sensores muy fácilmente, haciéndolo también compatible con muchos de ellos. A continuación se mencionaran algunos de estos sensores.

Un sensor, según Manuel Rodríguez, Ingeniero Industrial (2015), es:

El elemento que se encuentra en contacto directo con la magnitud que se va a evaluar, al interactuar con estas sufre cambios en sus propiedades. Por ejemplo la magnitud física puede ser la temperatura y la propiedad alterada puede ser la resistencia eléctrica que varía proporcionalmente a la variable medida.

Existen varios tipos de sensores que utiliza comúnmente o casi siempre un dron, y estos sensores son los siguientes:

- Acelerómetro
- Giroscopio
- Magnetómetro
- Ultrasónico

- Barométrico
- GPS

Pero entre este tipo de sensores destacan por su importancia, y que son prácticamente imprescindibles en un dron para poder estabilizarse, los acelerómetros y giroscopios, que hoy en día son denominados IMUs. En los últimos años han experimentado un gran desarrollo y han pasado a convertirse en dispositivos pequeños, compactos, duraderos y baratos. Esto es debido en gran medida a la aparición y mejora de sistemas microelectromecánicos (MEMS, Microelectromechanical Systems), dispositivos que unen comportamientos mecánicos y electrónicos a escalas microscópicas, típicamente del orden de nanómetros (ver figura 7). Muchos dispositivos cotidianos disponen de uno de estos sensores; por ejemplo, los móviles y tabletas, los emplean para detectar la orientación de la pantalla.

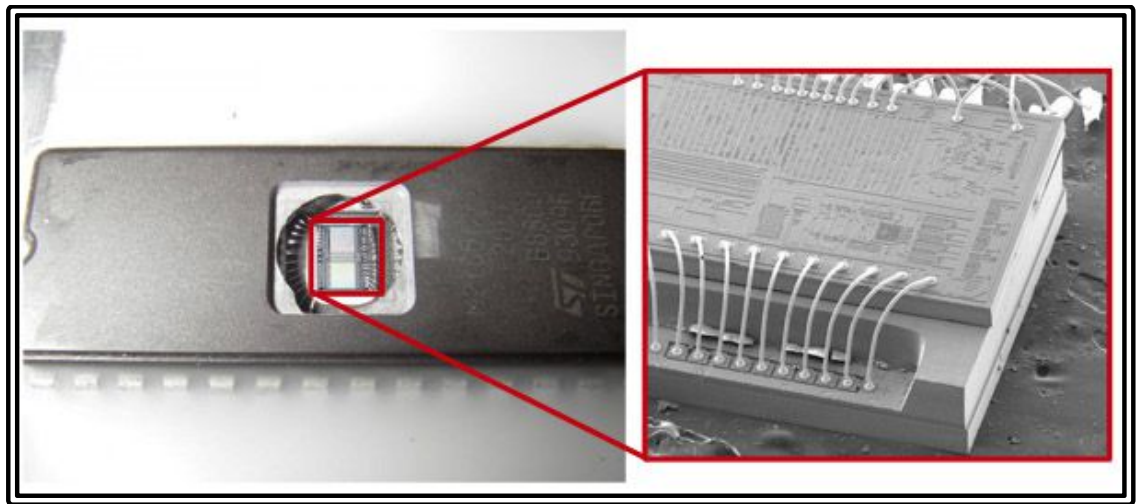


Figura 7. Circuito Integrado MEMS

Fuente: www.luisllamas.es (2016)

Una Inertial Measurement Unit (IMU) como su nombre lo indica, es una unidad de medición inercial, este es un nombre genérico para denominar a un

dispositivo que es capaz de medir la velocidad, orientación y aceleración de un sistema.

Al hablar de IMU's es frecuente hacer referencia a la cantidad de grados de libertad (DOF) que dispone. Los grados de libertad de un sensor representan la cantidad de magnitudes independientes que es capaz de medir; Así, un acelerómetro de 3 ejes ortogonales es un sensor de 3 DOF. Así mismo, un giroscopio que mida ángulos en 3 ejes ortogonales es un sensor de 3DOF.

Los IMUs que encontraremos más frecuentemente son:

- IMU de 6 DOF, combinando un acelerómetro de 3 ejes y un giroscopio de 3 ejes.
- IMU de 9 DOF, añadiendo una brújula magnética de 3 ejes.
- IMU de 10 DOF, que añade un barómetro para la estimación de la altura del sensor.

La mayoría de los IMUs parten de una combinación de acelerómetro y giroscopio, siendo este el IMU más habitual. El motivo es que ambos dispositivos combinan muy bien y compensan las limitaciones del otro. Los acelerómetros no tienen deriva (drift) a medio o largo plazo, ya que realizan la medición absoluta del ángulo que forma el sensor con la dirección vertical, marcada por la aceleración de gravedad. Sin embargo, se ven influenciados por los movimientos del sensor y el ruido por lo que no son fiables a corto plazo.

Los giroscopios funcionan muy bien para movimientos cortos o bruscos, pero al usar giroscopios de vibración, que realmente miden la velocidad angular y obtienen el ángulo por integración respecto al tiempo, acumulan los errores y el ruido en la medición, por lo que a medio o largo plazo tienen deriva (drift).

Por tanto, combinar las mediciones de ambos dispositivos permite a las IMU obtener mediciones de la orientación más precisas que la de un acelerómetro y un giroscopio por separado.

El acelerómetro es un dispositivo que permite medir la aceleración a la que está sometido. Si se recuerda brevemente que la aceleración es la variación de la velocidad respecto del tiempo o, expresado matemáticamente como se muestra en la ecuación,

$$\vec{a} = \frac{\partial \vec{v}}{\partial t} = \frac{\partial^2 \vec{r}}{\partial t^2} \quad (1)$$

Asimismo, se tiene que por la primera ley de Newton

$$\vec{F} = m \cdot \vec{a} \quad (2)$$

Es decir, que cualquier cuerpo con una masa “m” requiere una cierta fuerza para variar su velocidad. Equivalentemente, cualquier cuerpo sometido a una aceleración experimentará una cierta fuerza. Empleando este hecho para entender el funcionamiento del acelerómetro; si se quiere fabricar un dispositivo que pueda medir aceleración; se podría construir un sensor formado por un cuerpo sólido, en cuyo interior suspenda una masa conocida sujeta por muelles al cuerpo exterior. Al aplicar una aceleración al conjunto la masa suspendida ejercerá una fuerza sobre los muelles causando que uno se contraiga y otro se alargue, por lo que la posición relativa de la masa dentro del sensor variará (ver figura 8)

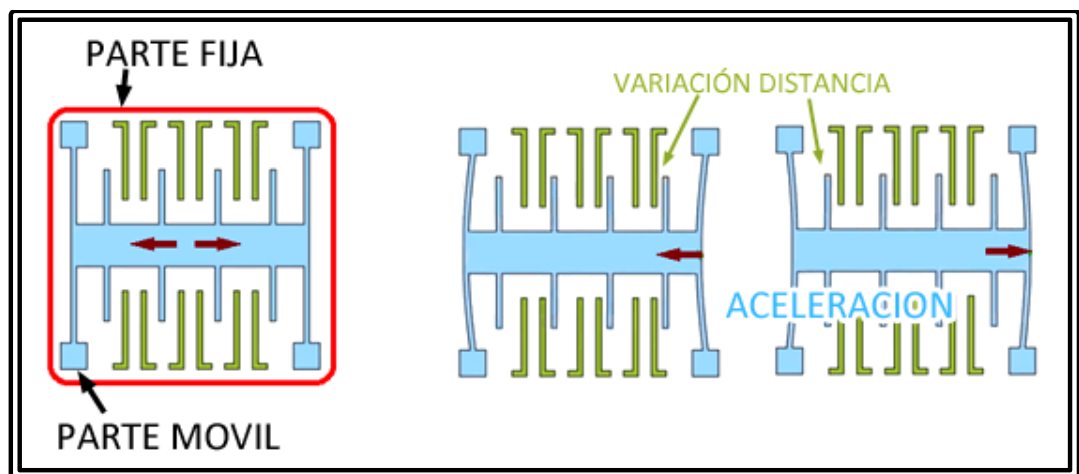


Figura 8: Composición mecánica de un Acelerometro
Fuente: www.luisllamas.es (2016)

Este desplazamiento de la masa libre interior puede ser medido para determinar la magnitud de la aceleración. El desplazamiento será proporcional a la aceleración soportada, y se mantendrá constante mientras la aceleración sea constante.

Para construir este acelerómetro en un MEMS se emplea una estructura micromecanizada de polisilicio construida sobre una oblea de silicio. Los muelles de polisilicio suspenden la estructura sobre la oblea de silicio, a la vez que proporciona resistencia para soportar las aceleraciones a las que se verá sometido el sensor; tal como se muestra en la figura (ver figura 9). Al someter el dispositivo a aceleraciones la parte interior se deforma y se desplaza respecto a la parte fija; el desplazamiento es detectado por la variación de la capacitancia del sistema.

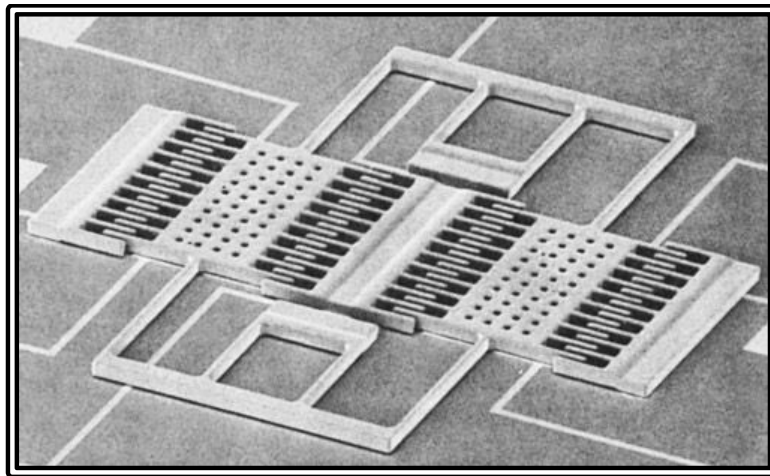


Figura 9: Estructura MEMS de un Acelerómetro.
Fuente: www.luisllamas.es (2016)

Los acelerómetros disponibles normalmente son de 3 ejes, es decir, son capaces de medir la aceleración a la que está sometido el sensor en X, Y y Z independientemente, lo que permite saber simultáneamente la magnitud y dirección de la aceleración medida. La capacidad de medir la aceleración de un sistema proporciona en si misma funcionalidades interesantes, como registrar vibraciones o golpes. Por ejemplo, los podómetros emplean un acelerómetro para detectar los pasos que se dan. Pero no es la única función que se puede obtener de un acelerómetro. El

sensor se ve afectado por la gravedad terrestre, que supone una aceleración de aproximadamente 9.81 m/s^2 en la superficie de la tierra, que por supuesto es registrada constantemente por el sensor.

Una primera consecuencia es que se puede detectar condiciones de caída libre, por ejemplo, cuando un sensor cae de una mesa, o un robot pierde el equilibrio y cae al suelo; simplemente se debe comprobar que la aceleración resultante medida es inferior o igual a 9.81 m/s^2 . Incluso más interesante la medición de la gravedad puede emplearse para determinar la orientación del sensor. Dado que se puede registrar la aceleración en tres ejes, en ausencia de otras aceleraciones, se puede determinar la orientación del sensor con un poco de trigonometría.

En el caso de 2D, asumiendo que pone el sensor horizontal en su plano X-Y, con Z apuntando hacia arriba, y únicamente se gira en el eje Y (ver figura 10), la ecuación resultante del ángulo es la ecuación 4.

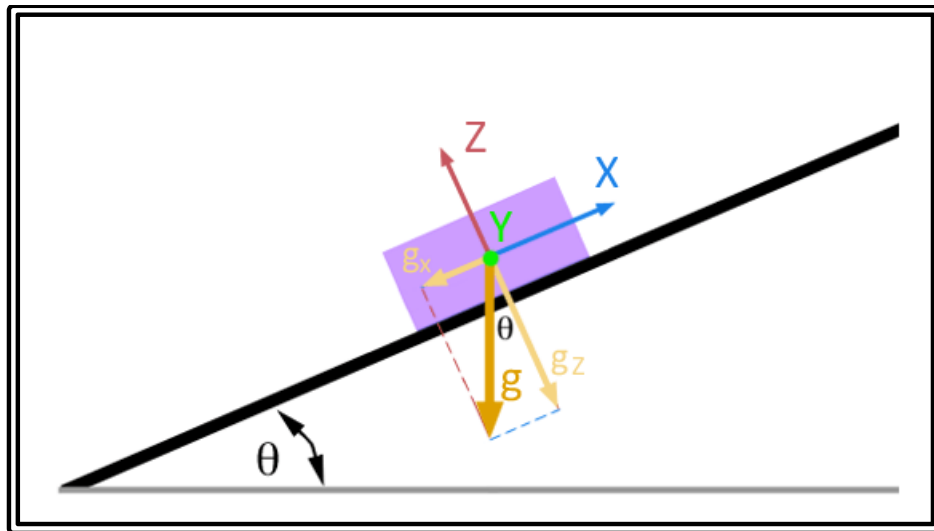


Figura 10: DCL del sensor inclinado en Y con descomposición del vector aceleración de gravedad en 2D

Fuente: www.luisllamas.es (2016)

$$\theta = \text{atan} \frac{A_x}{A_z} \quad (3)$$

De forma similar, en el caso 3D (ver figura 11), las ecuaciones resultantes son las ecuaciones 5, 6 y 7.

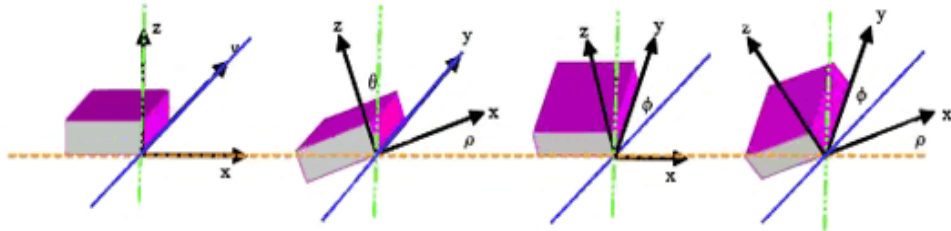


Figura 11: proyección de ángulo del sensor inclinado en distintos ejes en 3D
Fuente: www.luisllamas.com. (2016)

$$\theta_x = \operatorname{atan} \frac{A_x}{\sqrt{A_y^2 + A_z^2}} \quad (4)$$

$$\theta_y = \operatorname{atan} \frac{A_y}{\sqrt{A_x^2 + A_z^2}} \quad (5)$$

$$\theta_z = \operatorname{atan} \frac{\sqrt{A_x^2 + A_y^2}}{A_z} \quad (6)$$

Sin embargo, un acelerómetro no resulta un sensor adecuado para intentar determinar la velocidad de un sistema y mucho menos su posición. Se podría estar tentado a estimar la velocidad, simplemente multiplicando la aceleración por el tiempo. Con frecuencia, este tipo de mediciones integradas respecto al tiempo acumulan los errores de medición y el ruido, lo que conlleva efectos de deriva. Debería ser siempre el último recurso y nunca tenerlo en consideración más allá de una estimación burda.

Por lo tanto, un acelerómetro es un dispositivo muy versátil, que permite obtener mediciones de aceleraciones y orientación. Además, son sensores absolutos,

es decir, que existe un punto 0, 0, 0 de medición, que correspondería con el sensor cayendo libremente en el vacío. Finalmente, la medición no es integrada, por lo que no presentan deriva en la medición a medio o largo plazo.

Como aspectos negativos, son dispositivos muy sensibles a las vibraciones, por lo que la medición presentará ruido de alta frecuencia. En general, se debe filtrar la señal antes de poderla usar. En el caso más simple, simplemente hacer el promedio de varias mediciones será suficiente.

Un giroscopio (también llamados giróscopo) es un dispositivo que permite medir el ángulo de rotación girado por un determinado mecanismo. A diferencia de los acelerómetros, los giroscopios son dispositivos puramente diferenciales, es decir, no existe una referencia absoluta, sino que siempre se miden ángulos relativos a una referencia arbitraria.

Existen diversos tipos de giroscopios (mecánicos, de anillo láser, de fibra óptica). Los que se emplean en MEMS son denominados giroscopios vibratorios de efecto Coriolis (CVG), que en cierta forma pueden considerarse una evolución del péndulo de Foucault; recordando que la fuerza de Coriolis es una fuerza ficticia que aparece sobre un cuerpo en movimiento cuando se encuentra en un sistema en rotación. El valor de esta fuerza es:

$$\vec{F}_c = -2m (\vec{\omega} \times \vec{v}) \quad (7)$$

El principio de funcionamiento en un CVG es que un objeto vibratorio tiende a vibrar en el mismo plano, incluso si este rota. El efecto Coriolis provoca que el objeto vibratorio ejerza una fuerza sobre el soporte, y midiendo esta fuerza se pueda determinar la rotación a la que está sometido el giroscopio (ver figura 12).

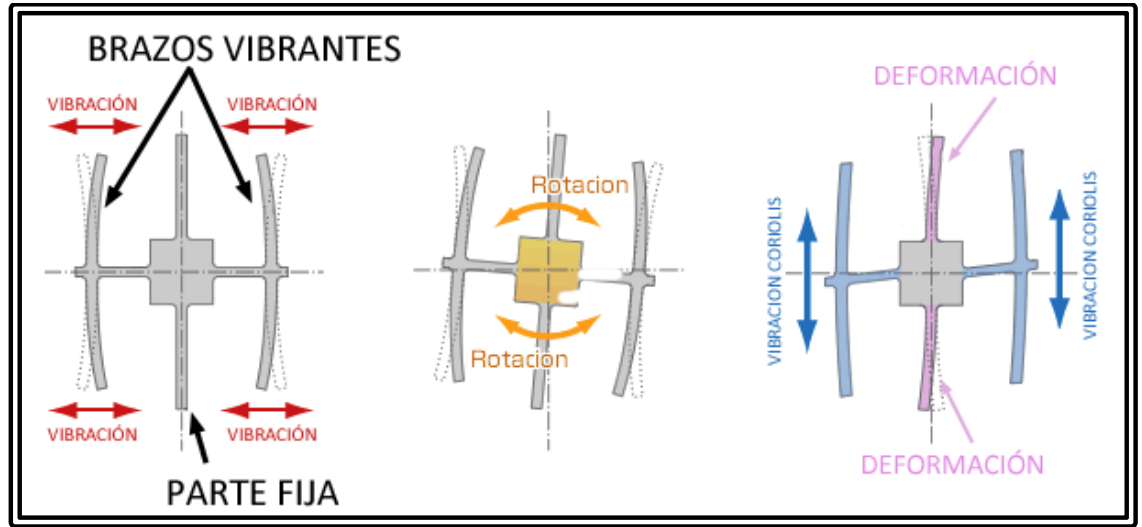


Figura 12: Composición mecánica de un giroscopio.

Fuente: www.luisllamas.es (2016)

Para registrar el efecto de la fuerza Coriolis un MEMS dispone de estructuras similares a las del acelerómetro. Ciertas partes del cuerpo se someten a vibración por resonancia y el efecto de la fuerza de Coriolis deforma la estructura, lo cuál puede ser medido por la variación de la capacitancia del sistema (ver figura 13).

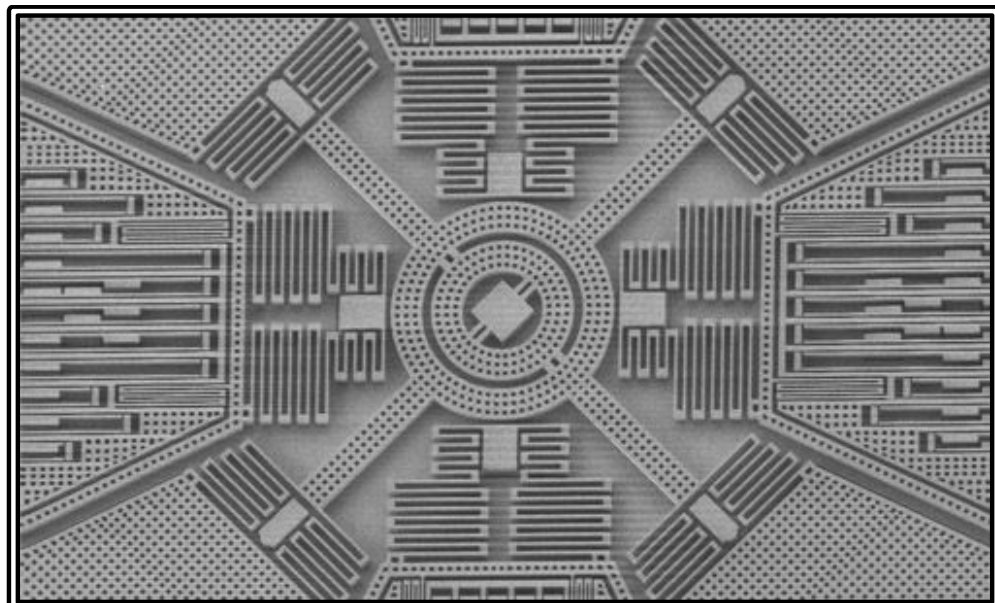


Figura 13: Estructura MEMS de un Giroscopio.

Fuente: www.luisllamas.es (2016)

Al igual que en el caso de los acelerómetros, lo normal es que los giroscopios que se emplean sean de 3 ejes, es decir, que registren de forma independiente la rotación en X, Y, y Z (ver figura 14), lo cual permite determinar la magnitud y dirección de la rotación.

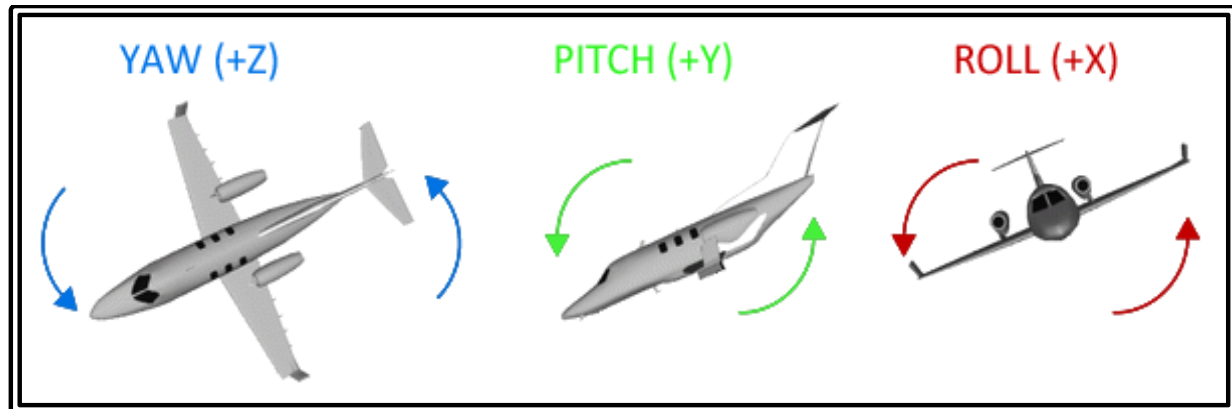


Figura 14: Ángulos de navegación.
Fuente: www.luisllamas.es (2016)

Una de las consecuencias de emplear la fuerza Coriolis es que los giroscopios vibratorios, a diferencia de otros tipos de giroscopios, no registran el ángulo girado si no la velocidad angular, que es la relación de variación del ángulo respecto del tiempo.

$$\omega = \frac{\partial \theta}{\partial t} \quad (8)$$

Para obtener el ángulo de posición del sensor es necesario realizar la integración respecto del tiempo.

$$\theta_{gyro} = \omega_{gyro} \cdot \Delta t \quad (9)$$

Las mediciones integrales nunca son una buena idea porque supone la acumulación de errores de medición y ruido, lo que provoca deriva en la medición (drift). Efectivamente, el mayor problema que se encuentra con los giroscopios de

vibración es que a medio y largo plazo tienen deriva, es decir, que la medida se va desviando progresivamente del valor real (incluso con el sensor estático).

Por el contrario, los giroscopios son sensores de respuesta rápida y elevada precisión en tiempos cortos. Además, responden bien a cambios bruscos, y son relativamente inmunes al ruido, siempre en rangos temporales cortos. Se puede apreciar que sus características de la medición son opuestas a las de los acelerómetros, por lo que como se mencionó anteriormente se complementan muy bien entre sí. Como se puede notar ninguno de estos dispositivos, tanto el acelerómetro, como el giroscopio pueden dar una referencia y una medición del eje Z o eje del YAW. Esto sucede porque girar el sensor en el eje Z no existe una variación en el vector aceleración de gravedad, ya que este vector es paralelo a dicho eje. Para la medición del ángulo del YAW lo más recomendable es el Magnetómetro.

El magnetómetro es un dispositivo que sirve para cuantificar en fuerza o dirección el campo magnético al que están sometidos. Según las magnitudes que pueden medir, los magnetómetros se pueden dividir en dos grandes grupos:

- Magnetómetros escalares: sólo miden la fuerza del campo magnético al que están sometidos.
- Magnetómetros vectoriales: además de medir la intensidad del campo magnético al que están sometidos tienen la capacidad de medir su dirección, así como la componente en una dirección particular.

El funcionamiento del magnetómetro se basa en que una masa de material ferromagnético altera el campo magnético terrestre, y por tanto, puede ser detectada. Muchos drones hoy en día utilizan un magnetómetro para poder orientarse en el eje del YAW o eje Z en el espacio como una brújula. Esto sucede ya que con un giroscopio o acelerómetro no se puede tener una referencia verdadera u orientación en dicho eje y no se podría crear un control autónomo para ese grado de libertad del dron.

Para determinar el campo magnético en un lugar y momento, se utilizan programas como el International Geomagnetic Reference Field, (IGRF), el cual

estima, mediante complejos modelos matemáticos, el campo en cualquier momento y lugar del planeta, a partir de las mediciones realizadas en diversos lugares del mundo por las estaciones magnetométricas.

Para conocer el campo en un lugar determinado basta conocer la declinación magnética del lugar donde se encuentra, que esta declinación no es más que el ángulo de desfase que existe entre el norte magnético y el norte geográfico. Con este valor se obtienen los datos necesarios para que se pueda programar y calibrar el magnetómetro para que al sensar el campo magnético se pueda luego obtener una lectura correcta con respecto al norte del lugar como una brújula. Esto se logra también con un poco de trigonometría como en el caso del acelerómetro que se explicó anteriormente (Ver ecuación 11).

$$\text{—} \quad (10)$$

Con esto concluimos con todos los sensores que se utilizan para medir los ángulos de navegación y se comenzaría con los sensores que miden la altitud, para así concluir con la sensorica que se utiliza para el control, monitoreo y estabilización de un dron, entre algunos de estos sensores tenemos los ultrasónicos y los de presión barométrica.

Los sensores ultrasónicos como su nombre lo indica, miden la distancia mediante el uso de ondas ultrasónicas. El cabezal emite una onda ultrasónica y recibe la onda reflejada que retorna desde el objeto. Los sensores ultrasónicos miden la distancia al objeto contando el tiempo entre la emisión y la recepción (ver figura 15).

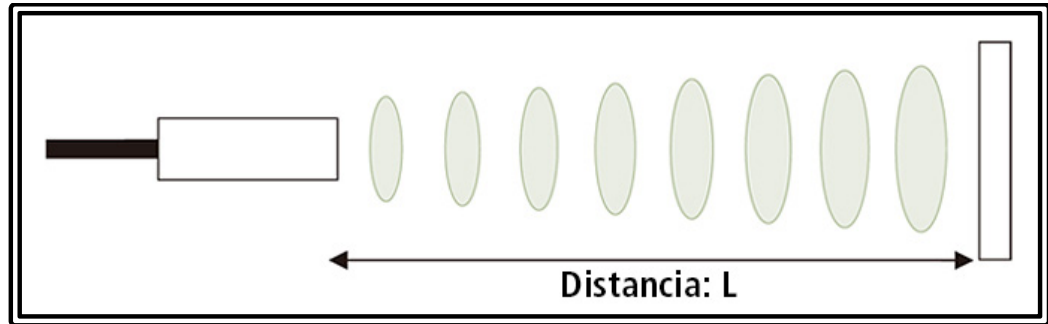


Figura 15: Representación de un sensor Ultrasónico.
 Fuente: www.naylampsmechatronics.com (2015)

La distancia se puede calcular con la siguiente fórmula:

$$L = \frac{C \cdot T}{2} \quad (11)$$

Donde L es la distancia, T es el tiempo entre la emisión y la recepción, y C es la velocidad del sonido. (El valor se multiplica por 1/2 ya que T es el tiempo de recorrido de ida y vuelta).

El sensor de presión barométrica en cambio está diseñado para leer la presión atmosférica y de esta forma estimar indirectamente la altura sobre el nivel del mar. La presión atmosférica es la fuerza que ejerce el aire (atmósfera) sobre la superficie de la tierra. La presión atmosférica se debe al peso de la columna de aire sobre determinada área, es por esta razón que al medir la presión atmosférica en puntos con mayor altitud, el valor de la presión es menor por ser menor la cantidad de aire. La presión atmosférica también varía con el clima, principalmente con la temperatura, pues esta hace cambiar la densidad del aire, que se ve reflejado en un cambio en el peso y por consiguiente en un cambio de presión.

La presión atmosférica varía con la temperatura y la altitud, estas dos variables son las más representativas para el cambio de presión. Factores como la humedad relativa y la velocidad del viento también influyen en la presión atmosférica en menor forma y pueden ser obviados. Lo que mide el sensor es la presión absoluta (Barométrica) y la temperatura, al sentir la temperatura se puede compensar su

influencia en la presión y así determinar con mayor exactitud la altitud. Se debe tener en cuenta que la presión barométrica varía continuamente debido a las condiciones climatológicas por lo que no proporciona una medición absoluta de la altitud con precisión. Sin embargo, sí puede ser de utilidad en mediciones diferenciales de altitud, es decir, diferencias de cota en un desplazamiento vertical. Para concluir con la sensorica se tiene el GPS que es el encargado de ubicar al dron en el globo o mapa.

Un GPS es un dispositivo recomendado para una ubicación u orientación en el mapa; con este, se logra obtener la posición exacta de un objeto por medio de la trilateración.

La trilateración es un método matemático para determinar las posiciones relativas de objetos usando la geometría de triángulos de forma análoga a la triangulación. A diferencia de ésta, que usa medidas de ángulo (junto con al menos una distancia conocida para calcular la localización del sujeto), la trilateración usa las localizaciones conocidas de dos o más puntos de referencia, y la distancia medida entre el sujeto y cada punto de referencia. Para determinar de forma única y precisa la localización relativa de un punto en un plano bidimensional usando sólo trilateración, se necesitan generalmente al menos 3 puntos de referencia (ver figura 16).

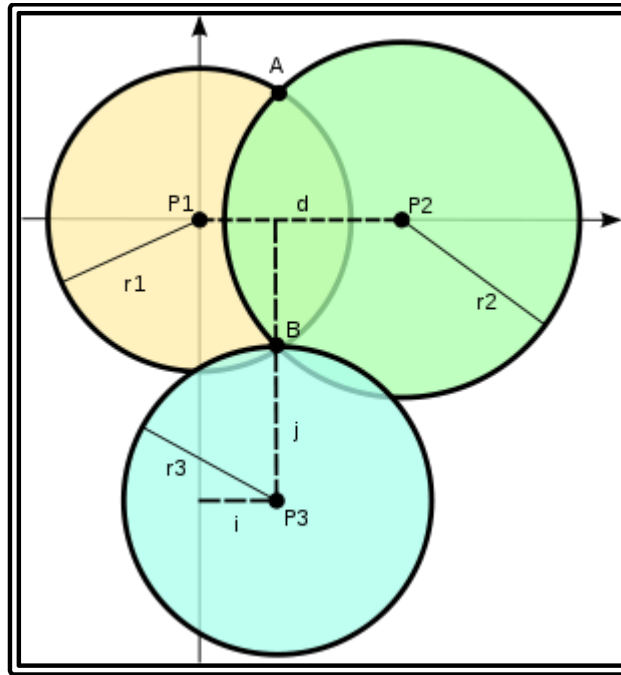


Figura 16: Método de trilateración de un GPS.
Fuente: www.wikipedia.com (2016)

Los drones con GPS ofrecen en todo momento el punto exacto en el que se encuentran, por lo que la posibilidad de disfrutar de nuevas formas de vuelo se amplían. Algunos drones que poseen este tipo de tecnología son el Parrot Disco FPV, un drone de ala fija capaz de alcanzar los 80 kilómetros por hora y volar hasta 45 minutos de forma ininterrumpida. Su emisora incluye un módulo GPS con la que se puede disfrutar de una experiencia de vuelo única con las gafas FPV que incorpora. Otro es el AR Drone 2.0 de Parrot en el que puedes incorporarle el GPS Flight Recorder, un módulo GPS externo compatible con su modelo para que así pueda disfrutar de las ventajas que esta tecnología ofrece.

El módulo GPS que incorporan ciertos drones permite a los pilotos establecer una ruta e indicarle al UAV que la siga. Para ello basta con indicar los puntos por los que debe pasar el drone – normalmente desde la emisora de control – y pulsar el botón de inicio de vuelo (ver figura 17).



Figura 17: Rutas Programadas.
 Fuente: www.naylampsmechatronics.com (2017)

Otras de las funciones que se pueden programar e incorporar en un dron es la del regreso a casa. El secreto de dicho funcionamiento reside en el módulo GPS que incorpora. El dron guarda el punto desde el cual despegó para volver en caso de que así lo indique el piloto. Estas son algunas de las funciones que se pueden incorporar en un dron que cuente con GPS, pero tengan por seguro que se puede hacer mucho más que eso con un GPS incorporado.

En cuanto a la parte de la electrónica de potencia, se tiene que los elementos que lo conforman son los siguientes:

- Baterías
- ESC
- Motores

Hay muchos tipos de baterías en el mercado, pero hay un solo tipo de baterías que es el más usado para drones por sus esplendidas características que la hacen ideal para ese trabajo y este tipo de batería son las baterías de Li –Po (Ver figura 18). El nombre viene de batería de polímero de iones de litio, de ion de litio polímero o más comúnmente batería de polímero de litio (abreviadamente Li-poli, Li-Pol, LiPo, LIP,

PLI o LiP). Estas son pilas recargables, compuestas generalmente de varias células secundarias idénticas en paralelo para aumentar la capacidad de la corriente de descarga, y están a menudo disponibles en serie de “packs” para aumentar el voltaje total disponible.



Figura 18: Baterías LiPo.
Fuente: www.dedrones.es (2016)

Estas baterías poseen 3 principales valores en los que hay que fijarse (ver figura 19):

- Número de celdas.
- mAh (miliamperios hora).
- tasa de descarga o también conocido como C's.



Figura 19: Batería de LiPo con sus respectivas características.
Fuente: www.dedrones.es (2016)

El número acompañado de la S, indica el número de celdas del cual está formada la batería y por ende su voltaje total. Las celdas son baterías de 3,7 V casi siempre conectadas en serie y por lo tanto, los valores se van sumando. (1S = 3,7 V) (2S = 7,4 V) (3S = 11,1 V) 4S (4S = 14,8 V) 5S=18,5 V), otros.

Los mAh es la capacidad de almacenamiento que tiene la batería. Cuanto mayor es ese valor, más minutos de vuelo tendrá el dron, pero pesan más, son más grandes y también más caras. Supóngase que tiene un Quadcopter con una batería 3S 4000 mAh que tiene una autonomía de 10 minutos de vuelo. Si se añadiera otra batería igual en paralelo se obtendrá un Quadcopter con una batería total de 3S 8000 mAh ¿debería volar aproximadamente 20 minutos? Pues lo normal es que no pase de los 15 con suerte. La explicación a este fenómeno está en el peso. Añadir más capacidad no siempre resulta productivo si se mantienen constantes el resto de características de la aeronave. A mayor peso, mayor consumo de energía y por tanto llega un momento en el que añadir más baterías (es el elemento más pesado de la aeronave) puede resultar contraproducente.

Por último se tiene los C's, que para resumirlo de una manera sencilla y sin tecnicismos, es la capacidad de la batería para entregar la potencia solicitada al instante sin problemas y sin dañarse. Viene indicado con una C (por ejemplo 75C), contra mas C's, mayor entrega de potencia podrá dar la batería. A más C's mejor, pero más cara y puede que nunca te hagan falta tantos C's por tu forma de pilotar o por tus componentes. Por ejemplo, una batería de 6000 miliamperios, si fuese de tipo 1C, sería capaz de entregar esa energía de 6 amperios durante una hora. Si fuese de tipo 2C, entregaría el doble de energía, 12 amperios, pero en la mitad de tiempo. Si llegamos a 20C, lo haría en apenas 5 minutos, pero con nada menos que 120 amperios. Por ello lo primero que hay que buscar o saber es el motor que se quiere, ya que es sabiendo la cantidad de corriente que consuma dicho motor se escogerán entonces los ESC y las baterías correspondientes para que satisfagan la demanda de corriente.

En cuanto a los motores que utilizan este tipo de aeronaves, no son motores de corriente continua “normales”. La cuestión es que necesitamos convertir la corriente continua de la batería con una tensión “constante” a una fuente de tensión variable y de sentido reversible por cada polo del motor, en pocas palabras podría decirse que estos ESC son inversores trifásicos de onda cuadrada. El dispositivo electrónico responsable de realizar dicha conversión es el llamado controlador de motor, controlador de velocidad de motor o simplemente ESC (de sus siglas en inglés Electronic Speed Controller) y tiene una apariencia similar a la mostrada en la figura (ver figura 20).



Figura 20: ESC.

Fuente: www.hobbyking.com (2016)

La característica a tomar en cuenta en un ESC a la hora de diseño es el amperaje que este maneja y el voltaje que soportan. Como se puede observar en la figura 20 el amperaje que maneja ese ESC es de 18 A.

En cuanto a los motores, como se mencionó anteriormente, no son de corriente continua comunes, en los drones se usan motores DC brushless (sin escobillas). Son motores mucho más eficientes que los motores con escobillas, ya que tienen mucho menos rozamiento, consumen menos, se calientan menos y necesitan menos mantenimiento. Se alimentan con corriente alterna (de ahí los 3 cables de cada motor) y para ello se necesitará un ESC o variador de velocidad que transforma la

corriente continua en corriente alterna y a una determinada frecuencia para darle la velocidad apropiada en cada momento según la decisión del controlador de vuelo (ver figura 21).



Figura 21: Motor de corriente continúa sin escobillas o BLDC

Fuente: www.hobbyking.com (2016)

El mejor motor para un Dron, dependerá del uso o estilo de vuelo, ya que al igual de lo que pasa con las hélices, se puede buscar una opción para mejorar eficiencia y por lo tanto autonomía u otra opción que dé más potencia y velocidad sacrificando un poco de tiempo de vuelo.

Un motor de dron o mini dron en la mayoría de los casos es brushless, solo en el micro dron, que tienen motores muy pequeños, se montan motores brushed (con escobillas). Las principales razones son porque son motores más baratos y porque necesitan menos electrónica (ESC). Son muy usados en los drones para niños precisamente por su precio y peso reducido (ver figura 22).

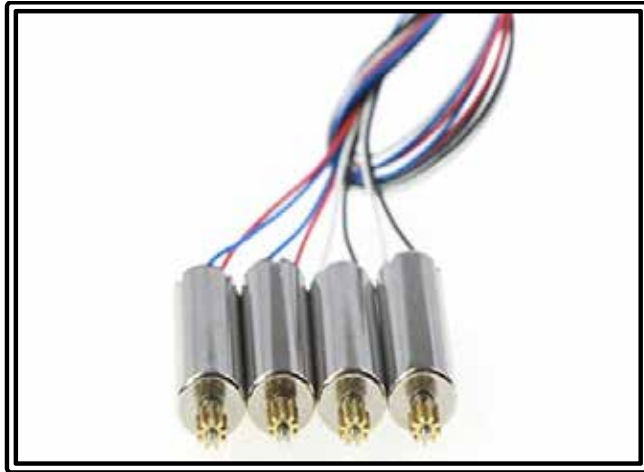


Figura 22: Motores DC brushed de dron Syma.
Fuente: www.maxtedrone.com (2015)

De modelos de motores brushless para Drones, los hay:

- Outrunner
- Inrunner

Los outrunner son los que la parte móvil esta fuera, que es la campana (los más comunes) y los inrunner la parte móvil está dentro.

La forma en la que están contruidos los motores brushless inrunner es disponiendo los imanes directamente en torno al eje, mientras que el bobinado es exterior y rodea el eje con los imanes, este tipo de motores tiene la ventaja de proporcionar un alto número de revoluciones por lo que su Kv (revoluciones por voltio) es muy alto, sin embargo esto tiene la desventaja de proporcionar un par muy bajo y si se quiere utilizar una hélice grande no se tendrá más remedio que emplear una reductora. Este tipo de motores es más que todo empleado en donde se requiere una hélice pequeña funcionando con un gran número de revoluciones como lo son los drones de carreras (ver figura 23).

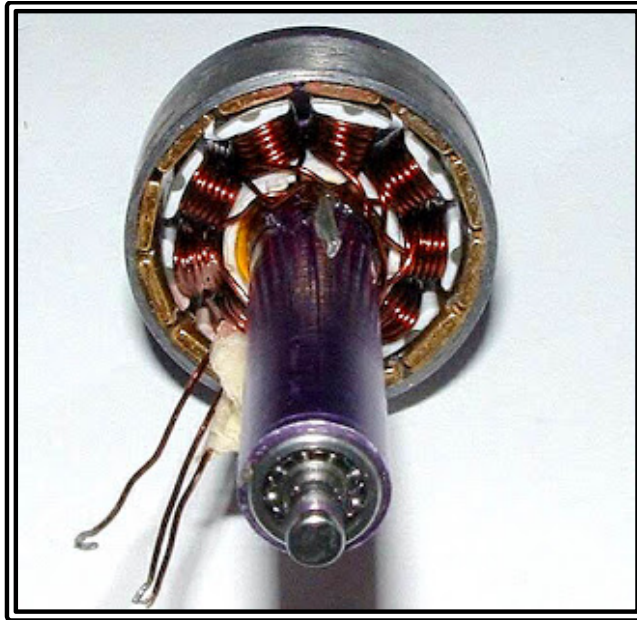


Figura 23: Motor Brushless Inrunner.
Fuente: www.maxtedrone.com (2015)

En los motores outrunner sucede al revés, en este tipo de motores los bobinados de cobre se disponen en la parte interior central, mientras que los imanes están situados en una campana exterior que rodea al bobinado y a la que se conecta el eje, cuando la campana gira lo hace también el eje, estos motores producen un menor número de revoluciones, sin embargo la campana al actuar como un volante de inercia, les dota de un mayor par por lo que dan más potencia sin necesidad del empleo de una reductora, esto les hace ser más ligeros, silenciosos y económicos (ver figura 24).



Figura 24: Motor Brushless Outrunner.
Fuente: www.maxtedrone.com (2015)

Los parámetros o características que describen a un motor, pero que son las más importantes y que se deberían encontrar en todas las fichas técnicas son:

- KV: El parámetro KV refiere al número de vueltas que gira un motor por minuto por cada voltio al que esta alimentado. Por ejemplo un motor de 2400 KV con una batería 3S (12V) girará a unas 28.800 vueltas por minuto. Los motores para drones medianos o grandes suelen andar alrededor de los 800-900 kV, ya que al montar hélices mucho más grandes no necesitan girar tan rápido para conseguir un empuje mucho superior.
- Tamaño: Aparte de los KV siempre suele haber otro valor al lado del valor de KV y este hace referencia al tamaño del motor. Por ejemplo los valores más comunes para motores pequeños son 1804, 1806, 2204... Las dos primeras cifras siempre harán referencia al diámetro del motor, más concretamente del diámetro de las bobinas (rotor) en mm, y el segundo valor a la altura del motor (del eje) en mm también.
- Empuje: Esta característica solo se podrá saber, analizando la ficha técnica del motor en la columna llamada habitualmente "Thrust (g)". Esta fuerza se mide en gramos, y es el dato determinante para decidir si los motores son apropiados para el dron que se quiere construir o no, ya que hace referencia al peso que es capaz de levantar el motor.

- Corriente: el dato de la corriente es importante saber cuál es su valor máximo cuando el acelerador esta al máximo. Este dato también aparece en la ficha técnica y determina los ESC que se debe usar. Por ejemplo, si consume un máximo de 11 A será recomendado montar ESC de 12 A.
- Agujeros: En la ficha técnica del acotamiento de las medidas del motor es esencial fijarse en el tamaño y posición de los agujeros de anclaje del motor. Esto es para asegurarse que son compatibles con el chasis o soporte de los motores de tu dron y evitar sorpresas inesperadas.

2.2.2.3 Software básico de un dron

A nivel de software para un dron si utilizamos Arduino se tiene el IDE oficial de Arduino que incorpora las herramientas para cargar el programa ya compilado en la memoria flash del hardware. Es el encargado de gestionar toda la información que recibe del exterior (sensores físicos, radio, bluetooth, entre otros) y actuar en consecuencia sobre los actuadores de la aeronave mediante un control PID (Proporcional Integral Derivativo), que es el encargado de mantener estable la aeronave y moverla hasta donde el usuario desee. Aparte el software también puede poseer una aplicación aparte que puede estar programada en distintos lenguajes web o Android, entre otros; para el adecuado monitoreo e incluso control de un dron inteligente.

2.2.3 Funcionamiento y movimientos de un Dron

La característica principal del Quadcopter son la disposición simétrica de los motores y su estructura, así como, la rotación asimétrica entre cada par de motores. Esto le permite equilibrarse aplicando un torque igual en los 4 motores (ver figura 25).

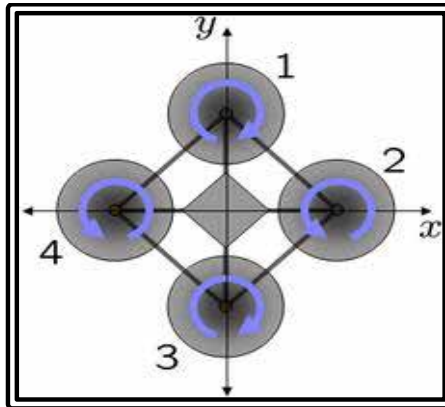


Figura 25: Disposición del Quadcopter.
 Fuente: TDG Sergio Moyano Díaz (2014, p.23).

El torque generado por 1,3 y 2,4 de la figura 3 son contrarios y eso auto-estabiliza el Quadcopter siempre y cuando se apliquen empujes de mismo módulo en los 4 motores. De esta manera no se necesita la hélice lateral de un helicóptero, simplificando el cálculo y el diseño.

En aeronáutica, las rotaciones que puede ejecutar un aparato según sus ejes se llaman Yaw, Roll y Pitch. En la siguiente figura se puede ver a qué ejes se refiere cada término (ver figura 26).

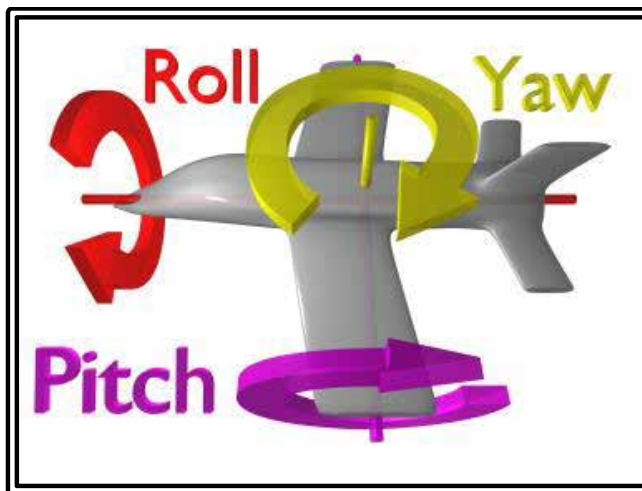


Figura 26: Rotaciones del Quadcopter.
 Fuente: TDG Sergio Moyano Díaz (2014, p.23).

2.2.3.1 Control de Altitud

La posición base es aplicando el mismo empuje a los 4 motores. Con la cantidad total de empuje se puede llegar a 3 movimientos (ver figura 27):

- Estabilizarse en el aire
- Subir altura
- Bajar altura

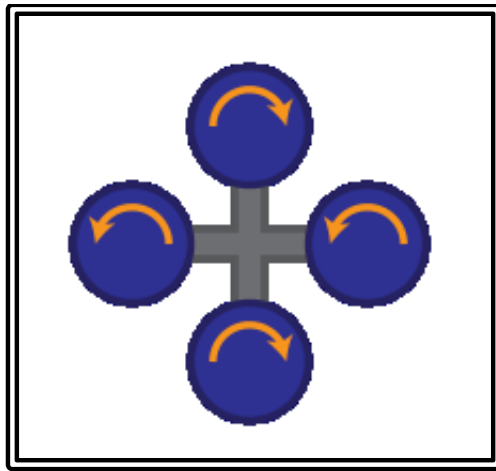


Figura 27: Control de Altitud

Fuente: TDG Sergio Moyano Díaz (2014, p.24).

2.2.3.2 Control del Yaw

Para ajustar el Yaw se sube el empuje de dos motores opuestos mientras los otros dos se mantienen estables (ver figura 28).

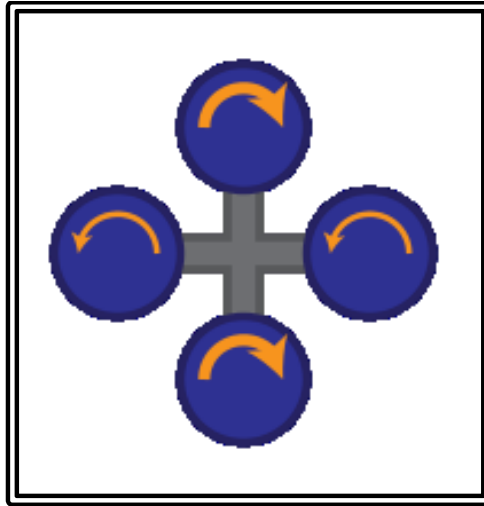


Figura 28: Control del Yaw
Fuente: TDG Sergio Moyano Díaz (2014, p.24).

2.2.3.3 Control del Roll.

El movimiento de roll se consigue modificando dos motores opuestos con la misma diferencia, pero en uno se sube y en otro se baja (ver figura 29).

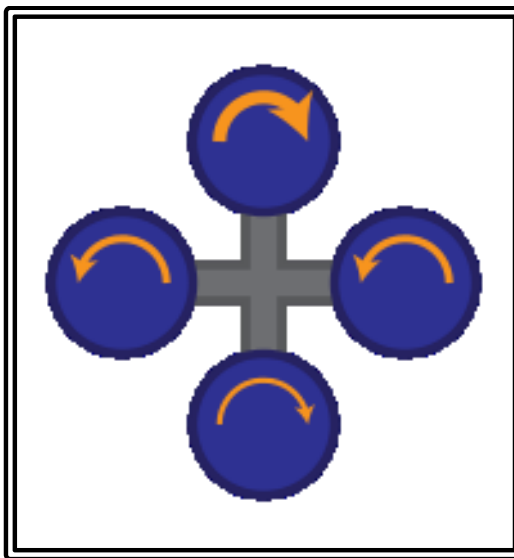


Figura 29: Control del Roll
Fuente: TDG Sergio Moyano Díaz (2014, p.25).

2.2.3.4 Control del Pitch

El movimiento de pitch se hace de la misma manera que el roll, pero tocando los otros dos motores (ver figura 30).

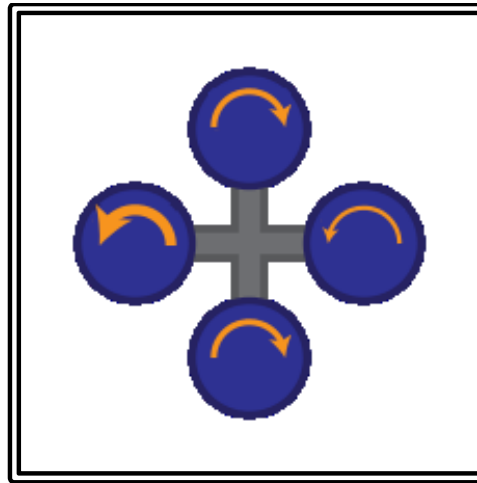


Figura 30: Control del Pitch

Fuente: TDG Sergio Moyano Díaz (2014, p.25).

2.2.4 Componentes de sistemas de vigilancia básicos que pueden implementarse en un dron

Según David Abedul (2005) en un artículo digital de The Guardian. “La vigilancia es el monitoreo del comportamiento” La vigilancia por sistema es el proceso de monitoreo de personas, objetos o procesos dentro de sistemas para la conformidad de normas esperadas o deseadas en sistemas confiables para control de seguridad o social. Existen diversos componentes para sistemas de vigilancia básicos, entre los más comunes se tienen:

- sensores de movimiento o presencia
- cámaras de media – alta definición con visión nocturna.

2.2.4.1 Sensores de movimiento y/o presencia

En los sensores de movimiento y/o presencia, el más utilizado es el sensor PIR (Passive Infra Red). Los sensores infrarrojos pasivos (PIR) son dispositivos para la detección de movimiento. Son baratos, pequeños, de baja potencia, y fáciles de usar y los hay de varios modelos. Por esta razón son frecuentemente usados en juguetes, aplicaciones domóticas o sistemas de seguridad (ver figura 31).



Figura 31: PIR

Fuente: www.luisllamas.es (2016)

Los sensores PIR se basan en la medición de la radiación infrarroja. Todos los cuerpos (vivos o no) emiten una cierta cantidad de energía infrarroja, mayor cuanto mayor es su temperatura. Los dispositivos PIR disponen de un sensor piezoeléctrico capaz de captar esta radiación y convertirla en una señal eléctrica.

En realidad cada sensor está dividido en dos campos y se dispone de un circuito eléctrico que compensa ambas mediciones. Si ambos campos reciben la misma cantidad de infrarrojos la señal eléctrica resultante es nula. Por el contrario, si los dos campos realizan una medición diferente, se genera una señal eléctrica. De esta forma, si un objeto atraviesa uno de los campos se genera una señal eléctrica

diferencial, que es captada por el sensor, y se emite una señal digital como se observa en la figura (ver figura 32).

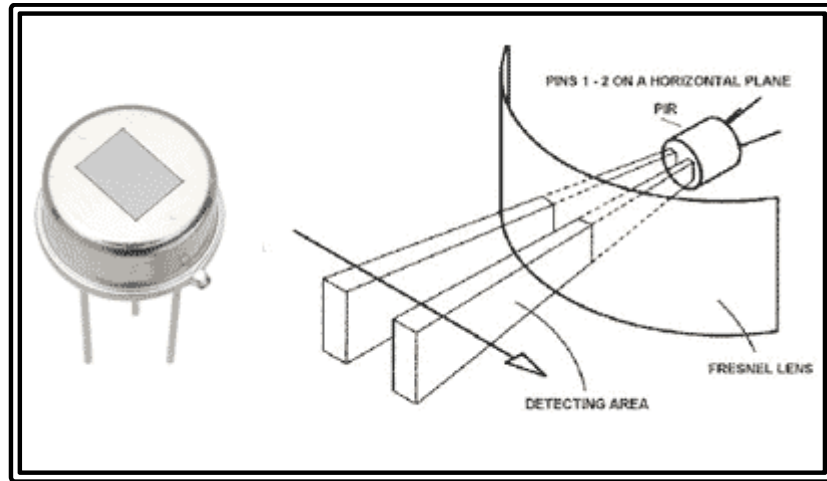


Figura 32: Campos de medición del sensor IR
Fuente: www.luisllamas.es (2016)

El otro elemento restante para que todo funcione es la óptica del sensor. Básicamente es una cúpula de plástico formada por lentes de fresnel, que divide el espacio en zonas, y enfoca la radiación infrarroja a cada uno de los campos del PIR. De esta manera, cada uno de los sensores capta un promedio de la radiación infrarroja del entorno. Cuando un objeto entra en el rango del sensor, alguna de las zonas marcadas por la óptica recibirá una cantidad distinta de radiación, que será captado por uno de los campos del sensor PIR, disparando la alarma (ver figura 33).

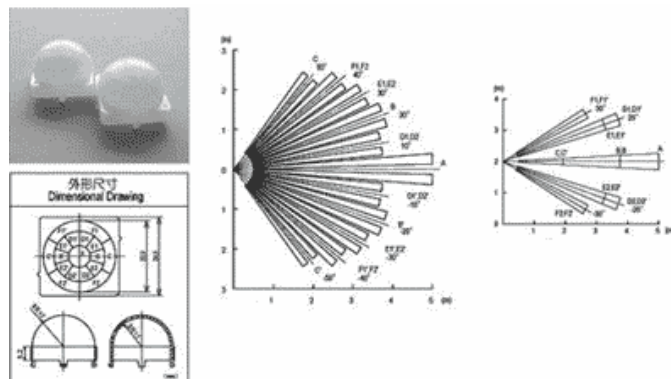


Figura 33: Rango de Alcance IR
Fuente: www.luisllamas.es (2016)

2.2.4.2. Cámaras de media – alta definición con visión nocturna

Las cámaras de vigilancia son las encargadas de captar todo lo que ocurra, Su sistema de visión nocturna permite ver lo que ocurre frente a la cámara incluso sin luz. Contar con una cámara de vigilancia con estas características puede reportarnos multitud de ventajas, ya que además de proporcionar una sensación de seguridad y protección adicional, también permiten identificar a personas, elementos y cualquier tipo de circunstancia que se pudiera generar y que resultase peligrosa o preocupante. Una cámara que cumple con todos estos aspectos es la PI NoIR. Pi NoIR es una cámara infrarroja de alta definición para Raspberry Pi. Tiene el mismo sensor de 5 megapíxeles que la cámara de alta definición estándar, pero sin el filtro infrarrojo, lo que le convierte en la herramienta perfecta para la captura de imagen en condiciones muy pobres de luz (ver figura 34).

La cámara Pi NoIR es compatible con el modelo A y el modelo B de Raspberry Pi. Su sensor es capaz de tomar fotos fijas con una resolución de 2592 x 1944, además de grabar vídeos en alta definición de hasta 1080p a 30 cuadros por segundo. Su modelo compacto, pequeño, su bajo costo y lo fácil de usar en una Raspberry Pi lo hacen ideal para ser implementado en un dron o cualquier sistema de vigilancia.

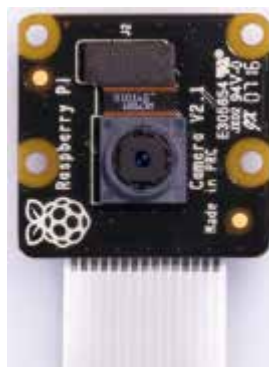


Figura 34: Pi NoiR camera
Fuente: www.amazon.com (2016)

2.2.5 Aplicación web

En la ingeniería de software se denomina aplicación web a aquellas herramientas que los usuarios pueden utilizar accediendo a un servidor web a través Internet o de una intranet mediante un navegador. En otras palabras, es una aplicación software que se codifica en un lenguaje soportado por los navegadores web en la que se confía la ejecución al navegador.

Las aplicaciones web son populares debido a lo práctico del navegador web como cliente ligero, a la independencia del sistema operativo, así como a la facilidad para actualizar y mantener aplicaciones web sin distribuir e instalar software a miles de usuarios potenciales. Existen aplicaciones como los webmails, wikis, weblogs, tiendas en línea y la propia Wikipedia que son ejemplos bastante conocidos de aplicaciones web.

Es importante mencionar que una página Web puede contener elementos que permiten una comunicación activa entre el usuario y la información. Esto permite que el usuario acceda a los datos de modo interactivo, gracias a que la página responderá a cada una de sus acciones, como por ejemplo rellenar y enviar formularios, participar en juegos diversos y acceder a gestores de base de datos de todo tipo.

2.2.5.1 Lenguajes de programación y diseño de una aplicación web

- **HTML5:** es la última versión de HTML (lenguaje de marcado de hipertexto) el cual nos permite hacer la estructura de la aplicación web (tablas, botones, textos, formularios, entre otros).
- **CSS3:** es la última versión de CSS (hoja de estilo en cascada) el cual nos permite agregarle estilo y animación a la estructura realizada con HTML.
- **JavaScript o JQuery:** JavaScript es un lenguaje de programación que se ejecuta del lado del cliente (a diferencia de PHP (pre-procesador de hipertexto) que se ejecuta del lado del servidor lo cual veremos más adelante) este se

encarga principalmente de agregarle dinámica a los objetos de HTML5 y de CSS3, pero recalando que es solo del lado del cliente. JQuery se podría decir que es una librería de JavaScript que realiza las funciones de JavaScript de una forma más sencilla al momento de la programación.

- **PHP:** como ya adelantamos nos permite agregarle dinámica a los objetos de HTML5 y CSS3, pero solo del lado del servidor.
- **MySql:** permite la conexión, los cambios, consultas, actualización de la información que se encuentran en la base de datos.
- **MIT App Inventor:** es un entorno de programación intuitiva y visual que les permite a todos, incluso a los niños, crear aplicaciones totalmente funcionales para teléfonos inteligentes y tabletas. Su programación es basada en bloques facilita la creación de aplicaciones complejas de alto impacto en mucho menos tiempo que los entornos de programación tradicionales.

2.2.6 Comunicación bluetooth

Bluetooth es una especificación industrial para Redes Inalámbricas de Área Personal (WPAN) creado por Bluetooth Special Interest Group, Inc. que posibilita la transmisión de voz y datos entre diferentes dispositivos mediante un enlace por radiofrecuencia en la banda ISM de los 2.4 GHz. Los principales objetivos que se pretenden conseguir con esta norma son:

- Facilitar las comunicaciones entre equipos móviles.
- Eliminar los cables y conectores entre estos.
- Ofrecer la posibilidad de crear pequeñas redes inalámbricas y facilitar la sincronización de datos entre equipos personales.

Los dispositivos que con mayor frecuencia utilizan esta tecnología pertenecen a sectores de las telecomunicaciones y la informática personal, como PDA, teléfonos móviles, computadoras portátiles, ordenadores personales, impresoras o cámaras digitales.

2.3 Definición de Términos Básicos

- **Quadcopter:** también llamado un helicóptero cuadrirrotor, es un helicóptero multirrotor que es levantado y propulsado por cuatro rotores. Los Quadcopter se clasifican como helicópteros, a diferencia de las aeronaves de ala fija, debido a que su elevación es generada por cuatro hélices orientadas verticalmente.
- **IMU:** una IMU (del inglés inertial measurement unit) es una unidad de medición inercial, es un dispositivo electrónico que mide e informa acerca de la velocidad, orientación y fuerzas gravitacionales (aceleración) de un aparato, usando una combinación de acelerómetros y giróscopos.
- **MEMS:** son una tecnología que, en su forma más general puede ser definida como elementos mecánicos y electro-mecánicos miniaturizados que se realizan con técnicas de micro-fabricación. Las dimensiones físicas críticas de estos dispositivos pueden variar desde muy por debajo de una micra en el extremo inferior del espectro de dimensiones, hasta varios milímetros.
- **Giroscopio:** El giróscopo o giroscopio es un dispositivo que sirve para medir, mantener o cambiar la orientación en el espacio de algún aparato o vehículo; estos miden o mantienen el movimiento de rotación y están compuestos por pequeños sensores, de bajo costo para medir la velocidad angular.
- **I2C:** Es un circuito inter-integrado (I²C, del inglés Inter-Integrated Circuit) es un bus serie de datos desarrollado en 1982 por Philips Semiconductors. Se utiliza principalmente internamente para la comunicación entre diferentes partes de un circuito, por ejemplo, entre un controlador y circuitos periféricos integrados. Cada dispositivo conectado al bus I2C, tiene su dirección exclusiva de 7 bits, (Así que, en teoría, se puede conectar 128, dispositivos).
- **Miniordenador:** ordenador de pequeño tamaño con una capacidad mediana de memoria que puede funcionar de forma automática o estar conectado a una red, funcionando como terminal o como servidor.

- **Microcontrolador:** es un circuito integrado programable, capaz de ejecutar las órdenes grabadas en su memoria. Está compuesto de varios bloques funcionales, los cuales cumplen una tarea específica.
- **Framework:** es un esquema (esqueleto o patrón) para el desarrollo y/o la implementación de una aplicación.
- **Firmware:** El firmware es un bloque de instrucciones de máquina para propósitos específicos, grabado en una memoria, normalmente de lectura/escritura, que establece la lógica de más bajo nivel que controla los circuitos electrónicos de un dispositivo de cualquier tipo.

CAPÍTULO III

MARCO METODOLÓGICO

3.1 Tipo de Investigación

Este trabajo de investigación cumple con el marco de un proyecto factible, el cual según define El Manual de Tesis de Grado y Especialización y Maestría y Tesis Doctorales de la Universidad Pedagógica Libertador, (2006):

Proyecto factible consiste en la investigación, elaboración y desarrollo de un modelo operativo viable para solucionar problemas, requerimientos necesidades de organizaciones o grupos sociales que pueden referirse a la formulación de políticas, programas, tecnologías, métodos, o procesos. El proyecto debe tener el apoyo de una investigación de tipo documental, y de campo, o un diseño que incluya ambas modalidades (p. 16).

3.2 Diseño de la Investigación

Luego de haber delimitado el tipo de investigación asumido en el estudio de tipo de proyecto factible, se define el diseño de la investigación el cual es un plan global que integra de un modo coherente y adecuadamente correcto técnicas de recolección de datos a utilizar, análisis previstos y objetivos, esta investigación se encuentra apoyada en el modelo de investigación de campo. Tal como lo refiere Tamayo y Tamayo (2005): “Cuando recurrimos a la utilización de datos secundarios, es decir, aquellos que han sido regidos por otros y nos llegan elaborados y procesados de acuerdo con los fines de quienes inicialmente”. Tal como lo refiere Fidas G. Arias (2012):

La investigación de campo es aquella que consiste en la recolección de todos directamente de los sujetos investigados, o de la realidad donde ocurren los hechos (datos primarios), sin manipular o controlar variables alguna es decir, el investigador obtiene la información, pero no altera las condiciones existentes.

Claro está, en una investigación de campo también se emplea datos secundarios, sobre todo los provenientes de fuentes bibliográficas, a partir de los cuales se elabora el marco teórico. No obstante, son los datos primarios obtenidos a través del diseño de campo, lo esenciales para el logro de los objetivos y la solución del problema planteado. (pag.31)

Por lo tanto, para apoyar el desarrollo del presente trabajo se sustentaron las herramientas utilizadas con material bibliográfico desarrollado por autores reconocidos en el área.

3.3 Nivel de la investigación

El nivel de investigación es descriptivo ya que pretende conocer las situaciones predominantes, a través de la descripción de las actividades, problemas y personas involucradas para así poner de manifiesto la estructura del fenómeno o problema de estudio. Según el autor Fidias G. Arias (2012), define:

La investigación descriptiva consiste en la caracterización de un hecho, fenómeno, individuo o grupo, con el fin de establecer su estructura o comportamiento. Los resultados de este tipo de investigación se ubican en un nivel intermedio en cuanto a la profundidad de los conocimientos se refiere. (pag.24).

3.4 Población y Muestra

En este estudio se considera utilizar como población la Universidad José Antonio Páez. La muestra es aleatoria y representada solo por una pequeña parte de los estudiantes de ingeniería electrónica y telecomunicaciones de la universidad cursantes del octavo, noveno y décimo semestre, ya que como poseen conocimientos de microcontroladores, programación y electrónica en general pueden captar y entender mejor lo que es y lo que este proyecto representa, por lo tanto, será a los que se les aplicará la técnica o instrumento de recolección de datos.

3.5 Técnicas e Instrumentos de Recolección de Datos

Arias (1999), menciona que “las técnicas de recolección de datos son las distintas formas de obtener información”. (pág.53). al igual que Hurtado (2000), define que las técnicas tienen que ver con los procedimientos utilizados para la recolección de datos, es decir, el cómo. Estas pueden ser revisión documental, observación, encuestas, entrevistas, entre otras. Para este trabajo de grado se utilizará la encuesta como técnica de recolección de datos, que según Grasso (2006) explica

” La encuesta es un procedimiento que permite explorar cuestiones que hacen a la subjetividad y al mismo tiempo obtener esa información de un número considerable de personas así por ejemplo: Permite explorar la opinión pública y los valores vigentes de una sociedad, temas de significación científica y de importancia en las sociedades democráticas. (pag.13).”

Al respecto, Mayntz (1976) citados por Díaz de Rada (2001), describen a la encuesta como la búsqueda sistemática de información en la que el investigador pregunta a los investigados sobre los datos que desea obtener, y posteriormente reúne estos datos individuales para obtener durante la evaluación datos agregados.

3.6 Validez

El instrumento de recolección de datos seleccionado (encuesta), fue validado por tres profesores de la escuela de ingeniería preparados para avalar la pertinencia de las preguntas formuladas en la entrevista, donde dieron sus observaciones y sugerencias, las cuales fueron tomadas en cuenta al momento de realizar la recolección de los datos.

3.7 Confiabilidad

Hernández (2003) expresa que:

La confiabilidad de un instrumento de medición se determina mediante diversas técnicas, y se refieren al grado en la cual su aplicación repetida al mismo sujeto produce iguales resultados.

Existen diversos procedimientos para calcular la confiabilidad de un instrumento de medición. Todos utilizan fórmulas que producen coeficientes de confiabilidad y que pueden oscilar entre 0 (nula confiabilidad) y 1 (máxima confiabilidad), es decir, mientras más se acerque a cero, mayor será el error” (p.47).

La encuesta se realizó a una población que garantizó que los datos obtenidos a partir de esta sean exactos, para ello se utilizó la técnica estadística: Método de Kuder-Richarson 20. Siendo el método más efectivo para obtener la confiabilidad a partir de datos de encuestas dicotómicas o binarias.

Dónde:

n: Número total de ítems.

: Varianza de las puntuaciones totales.

P: Cantidad de respuestas afirmativas a cada ítem

Q: Cantidad de respuestas negativa a cada ítem.

Se interpretará en magnitud del coeficiente de confiabilidad. Donde sus valores oscilan entre 0 y 1.

Se realizó el cálculo de confiabilidad para de la encuesta con el método de Kuder-Richarson 20, para ello fue necesario agrupar los datos en una tabla de distribución de respuesta (ver cuadro 1). Al ser una encuesta dicotómica el resultado se expresa de forma binaria siendo Si (1) y No (0).

Cuadro 1. Distribución de las respuestas del cuestionario.

Sujetos	Ítems									
	1	2	3	4	5	6	7	8	9	10
1	1	1	1	1	1	1	1	1	1	1
2	1	1	1	1	1	1	1	1	1	1
3	1	1	1	0	1	1	1	1	1	1
4	1	1	1	1	1	1	1	1	1	0
5	1	1	1	1	1	1	1	1	1	1
6	1	1	1	1	1	1	1	1	0	1
7	1	1	1	0	1	0	1	1	1	1
8	1	1	1	1	1	1	1	1	1	1
9	1	1	1	1	1	1	1	1	1	0
10	1	1	1	1	1	1	1	1	1	1
11	1	1	1	1	1	1	1	1	1	0
12	1	1	1	1	1	1	1	1	1	1
13	1	1	1	0	1	1	1	1	1	1
14	1	1	1	1	1	1	1	1	1	1
15	1	1	1	0	1	1	1	1	1	0
16	1	1	1	1	1	1	1	1	1	1

Fuente: Mario Pontillo (2017)

Con la ecuación, se tiene:

Dónde:

—

A: respuestas afirmativas

N: Total de sujetos (Personas encuestadas)

Con el valor obtenido de se busca en el cuadro 2:

Cuadro 2. Interpretación de magnitud del coeficiente de confiabilidad.

Coeficiente de correlación	Magnitud
1 a 0,81	Muy alta
0,80 a 0,61	Alta
0,60 a 0,41	Moderada
0,40 a 0,21	Baja
0,20 a 0,01	Muy baja

Fuente: Ruiz (2002), Pallella y Martins (2003).

El valor de fue de magnitud alta. Lo que se traduce a que la encuesta es confiable y es consistente con los resultados, al ser confiable se puede hablar de que los resultados de la encuesta realizada son válidos.

3.8 Técnica de análisis de datos

En una investigación del tipo y diseño de la presente, para realizar el análisis de los datos que se obtuvieron a partir de la técnica de recolección de datos seleccionada, fue necesario la aplicación de herramientas estadísticas, en este sentido según define Tamayo y Tamayo (2005): “A través de la estadística se procesan los datos obtenidos para, organizar, analizar, e interpretar de forma adecuada los resultados” (p 47).

Para observar y analizar de una forma más clara y eficaz los resultados obtenidos, dichos resultados se graficarán en diagramas circulares.

3.9 Fases Metodológicas

Las fases metodológicas constituyen un seguimiento detallado y minucioso de los objetivos específicos planteados anteriormente, que servirán de guía en el cumplimiento del objetivo general, el cual es la meta principal de ésta investigación.

Fase I. Investigar las tecnologías actuales referente a drones

En esta fase se investigará acerca de las tecnologías que se implementan hoy en día en drones, como los elementos básicos que componen a este y el hardware y software más utilizados para otorgarle “autonomía” a este tipo de dispositivos analizando su alcance, ventajas y desventajas, también los costos que estos representan, para determinar qué es lo más factible y conveniente para este proyecto.

En esta fase también se investigarán los distintos drones que ya se encuentran en el mercado desde los más populares contruidos por compañías hasta drones hechos por aficionados analizando la proyección de dichos sistemas, inclinándose en los enfocados en el ámbito de la seguridad.

Fase II. Diseñar la estructura, hardware y software del prototipo

En esta fase se procederá a diseñar la estructura, hardware y software necesarios para el correcto funcionamiento del prototipo, para que este pueda cumplir con los objetivos planteados. Esta es la fase más importante de este trabajo de grado, ya que se desarrollará mediante exhaustivas investigaciones de los sensores, controladores y diferentes tipos de estructura a utilizar, realizando primeramente pruebas en estos para comprobar su correcto funcionamiento y para luego poder lograr una correcta calibración de los mismos. Luego de las pruebas anteriormente mencionadas, se procederá a probar los motores y controladores de velocidad y se

implementaran dichos sensores, para así poder empezar a diseñar un estabilizador y control de vuelo óptimo controlados por un Arduino. Para esto primero, se simularán distintos ambientes para recrear las perturbaciones y condiciones en las que este proyecto se verá sometido antes de ensamblar el dron final a utilizar para evitar daños innecesarios.

Fase III. Desarrollo de una interfaz de control y supervisión del Dron para los usuarios en una aplicación Android

En esta fase se diseñará y programará una aplicación Android, la cual servirá de interfaz HMI para el usuario. Dicha interfaz será diseñada con un lenguaje y entorno de programación sencilla, pero lo suficientemente potente, como lo es MIT App Inventor 2.

Con esta aplicación el usuario podrá administrar, supervisar, controlar y configurar el sistema mencionado en las fases anteriores, completando el dron de vigilancia propuesto.

Fase IV. Implementar el dron inteligente diseñado

En esta fase se procederá a unir e implementar todos los elementos y sistemas mencionados en las fases anteriores, para obtener el prototipo final e introducirlo en el medio para el que fue construido y así comenzar con pruebas de funcionamiento en condiciones más reales

CAPÍTULO IV

RESULTADOS

Las siguientes fases describen el desarrollo de los objetivos específicos establecidos al inicio de este trabajo de grado.

4.1 Fase I. Investigar las tecnologías actuales referente a drones

Se investigaron los distintos elementos que conforman un dron, como se observa en las bases teóricas del capítulo II de este trabajo de grado. Como se puede observar se investigó la teoría y el funcionamiento de dichos elementos y no solo eso, se hizo una recolección exhaustiva de datos e información acerca de las distintas marcas y tipos de componentes que actualmente utilizan los diferentes drones inteligentes en el mercado, y una comparación de eficiencia – costo de estos para que no eleven el precio del sistema, por esto, los principales componentes del prototipo son de bajo costo, pero con tecnología moderna. Ya que los recursos eran muy limitados se considera que esta es una de las fases más importante del proyecto, debido a que no cabían errores en la selección de los componentes para la realización de dicho prototipo; era vital tener los componentes correctos para obtener el funcionamiento deseado, por ende se invirtió gran parte del tiempo de la investigación en esta etapa. De acuerdo a la información recolectada de dicha investigación la estructura de un dron inteligente puede dividirse en cuatro etapas:

- CPU
- Sensorica
- Sistema de propulsión
- Sistema de comunicación

4.1.1 CPU

En la etapa del CPU se tiene el Arduino, ya que por ser un dispositivo open – source se puede conseguir una cantidad innumerable de información, uso de dispositivos y librerías en la red, aparte de ser lo suficientemente potente y cómodo de utilizar para este tipo de proyectos. otro detalle importante que ayudó a tomar la decisión de utilizar Arduino es la disposición de un bus I2C (Inter-Integrated Circuit) el cual pone en comunicación distintos circuitos integrados, esto permite incluir un sensor IMU con el que se mid la inclinación en cada eje y en definitiva la orientación del dron y así cerrar el lazo de control.

4.1.2 Sensorica

En la sensorica se tiene los componentes para la estabilización y control del dron, y estos componentes son los siguientes:

- MPU 6050
- HMC5883L
- BMP180
- GPS UBLOX NEO 6M

El MPU 6050 es un sensor IMU que está formado por 3 acelerómetros y 3 giróscopos teniendo la posibilidad de añadirle a parte un magnetómetro de tres ejes que lo complementa para medir el ángulo de giro en el eje Z (ver figura 35). Se utiliza este sensor por dos razones principalmente. La primera de ellas su precio, el más competitivo en el mercado en los últimos años y que por su extendido uso está verificada su fiabilidad. La segunda razón reside en que se pueda ampliar el sensor con la conexión de un magnetómetro. InveSense ofrece también un sensor que integra el MPU 6050 más un compás de tres ejes, el MPU9150 pero por razones de limitación de recursos se optó por este sensor y un magnetómetro aparte. Se maneja por I2C desde cualquier microcontrolador como Arduino. Dentro de una de las ventajas que cabe mencionar, El MPU-6050 posee conversores analógicos digitales

por cada uno de los ejes de cada uno de los sensores para obtener los valores en simultaneo con un rango de hasta $2000^{\circ}/s$ para el giroscopio y hasta $\pm 16g$ para el acelerómetro.

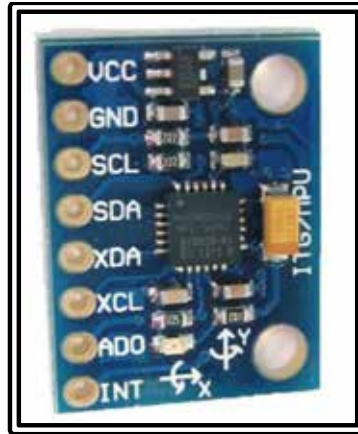


Figura 35: MPU 6050
Fuente: www.luisllamas.es

El HMC5883L es el magnetómetro escogido para terminar de complementar el MPU050 y completar el IMU con 9 DOF Y así obtener los ángulos de Euler. Se escogió el HMC5883L por las mismas razones que el MPU6050 y que aparte era compatible con este. El HMC5883L es un magnetómetro digital diseñado para medir campos magnéticos débiles (como los que hay presentes de forma natural en nuestro planeta). El rango de escala completa es de ± 8 gauss y la resolución del sensor es de hasta 5 mili-gauss. El rango de alimentación del HMC5883L es igual al del MPU 6050 de 2.16 a 3.6 volts, aunque puede alimentarse con 5 volts gracias a su regulador interno y se comunica de forma digital con el microcontrolador a través de una interfaz I2C (ver figura 36).

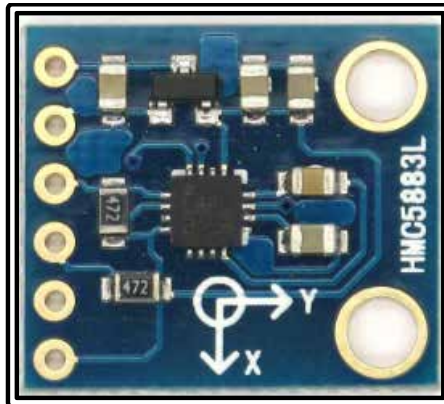


Figura 36: HMC 5883L
Fuente: www.robologs.com

El BMP 180 también es un sensor que se conecta de la forma I2C, aunque este no sea un IMU, esto es una gran ventaja por el ahorro de pines de entradas y salidas del Arduino y del fácil manejo de este tipo de protocolo de comunicación. Las razones de su elección también son las anteriormente mencionadas y que su modelo altamente compacto ahorra espacio y peso en el diseño del sistema (ver figura 37).

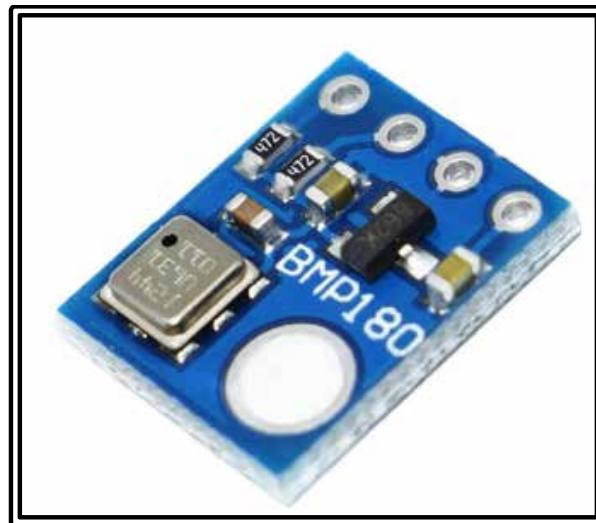


Figura 37: BMP180
Fuente: www.naylampsmechatronics.com

El BMP180 es un sensor de alta precisión y baja potencia. El rango de medición es de 300hPa a 1110 hPa, equivalente a una altitud de -500m a 9000m sobre el nivel del mar. La precisión es configurable, desde 0.06hPa (0.5 metros) en el modo de bajo consumo, a 0.02hPa (0.17 metros) en el modo de alta precisión. Este también incorpora un sensor de temperatura para compensar sus efectos en la medición de la presión barométrica.

Por último se tiene el GPS Ublox NEO 6 M (ver figura 38), se escogió este GPS por ser uno de los más utilizados en drones y por ser compatible con las librerías de uso de GPS de Arduino con el TinyGPS y TinyGPS++, haciendo mucho más fácil y cómodo su uso y la interpretación de datos que arroja este sensor.

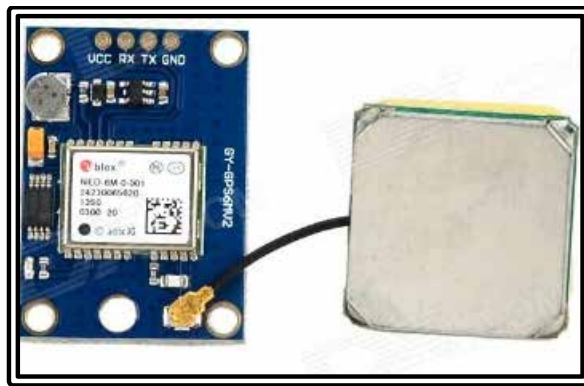


Figura 38: GPS Ublox NEO 6M
Fuente: www.naylampsmechatronics.com

El módulo incluye su antena cerámica para colocarse directamente sobre el PCB, por lo que ya viene listo para operar sin requerir más accesorios, esta fue otra de las razones atractivas por la que se escogió este sensor. Este módulo opera con un voltaje de entre 3.3 V y 5 V y en vez de utilizar I2C utiliza una interfaz UART, que es un tipo de comunicación asíncrona, también sencilla de utilizar.

4.1.3 Sistema de propulsión

Como sistema de propulsión se ha denominado todo lo que tenga que ver con el levantamiento de vuelo del dron, en este sistema se tienen los siguientes componentes:

- Motores
- Hélices
- Drivers o ESC
- Batería

En el mercado los motores más utilizados a la hora de elaborar un dron o inclusive en los drones comerciales, son los motores BLDC (Brushless Direct Courrent) o motores DC sin escobillas, debido a las altas velocidades q estos pueden alcanzar y su poco mantenimiento y desgaste debido a que no poseen escobillas; por ende los motores que se escogieron para este proyecto son motores BLDC por las razones anteriormente mencionadas. Hay una gran cantidad de motores de este tipo en el mercado, desde bajo costo pero con un buen rendimiento, hasta motores bastante caros, claro está que también todo depende de lo que se desea hacer y de las exigencias del proyecto. Para este proyecto según lo investigado se decidió utilizar motores BLDC Outrunner, ya que no se quiere ni se necesita que el dron vuele a altas velocidades, sino más bien que tenga un buen empuje.

Los motores que más se ajustaban a nuestras necesidades y que tienen un precio bastante considerable son los motores EMAX CF2822, estos son motores que nos permiten usar hélices desde 8” hasta hélices de 10” para tener un buen rendimiento del motor, otorgándonos hasta 550 gramos de empuje por motor (ver figura 39).



Figura 39: Motor EMAX CF2822
Fuente: TDG Sergio Moyano Díaz (2014).

Las características de este motor son las siguientes (ver figura 40):

Número de celdas de batería	2-3X Li-Poly
Dimensiones del estator	22x10 mm
Diámetro del eje	3 mm
Peso	39 g
Empuje	De 200 a 600 g
Tamaños de la hélice	8", 9", 10"
Constante KV	1200
Rpm Max	7700
Intensidad máxima	15 A

Figura 40: características Motor EMAX CF2822
Fuente: TDG Sergio Moyano Díaz (2014).

Como se mencionó anteriormente y como se puede observar en las características el motor EMAX para generar un buen rendimiento de él se recomienda utilizar hélices de 8, 9 o 10", mientras la hélice sea más grande habrá más empuje,

claro está reduciendo las revoluciones. Por estas razones se escogieron hélices de 9” para este proyecto, las hélices son de plástico y son un par CW y otro par CCW.

Por ultimo tenemos los ESC o drivers , obviamente que para elegir el ESC correcto primero hay q saber qué tipo de motor se va a usar , ya que dependiendo de la cantidad de corriente que consuma el motor, se adquirirá el ESC adecuado que satisfaga la demanda de corriente. Los ESC escogidos fueron ESC EMAX BLHeli de 20 A con fin de sobredimensionar con respecto a los 15 A que consume cada motor, como se puede observar en las características del motor (ver figura 41).

Este ESC permite ajustar el rango de PWM a un minino 1ms a un máximo 2 ms, por defecto detecta el 0% de empuje a 1,2ms y el 100% de empuje a 2ms.

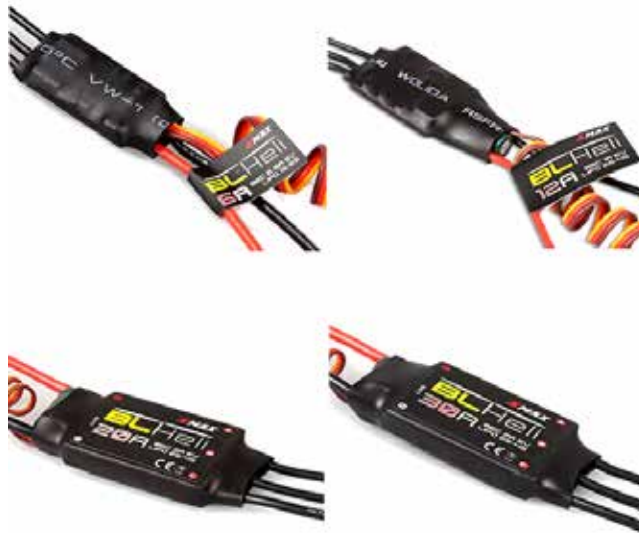


Figura 41: ESC EMAX 20 A
Fuente: www.hobbyking.com

Además incorpora varios tipos de protecciones:

- Protección de baja tensión:

Ya sea para apagar el motor de inmediato o para bajar la potencia cuando la entrada tensión cae por debajo del umbral de protección programado el motor deja de proporcionar par.

- La pérdida de la protección de señal: Se reduce la Potencia automáticamente al 20% o menos cuando se pierde la señal durante 1 segundo, y se reanuda cuando se detecta la señal de nuevo.
- La protección del sobrecalentamiento: Cuando la temperatura aumenta hasta por encima de 110 grados Celsius, la potencia es reducida hasta un 35%.

Por ultimo teniendo en cuenta lo anteriormente investigado y escogido en la etapa de sistema de propulsión, como lo son el modelo de motor y la cantidad de corriente que dichos motores consumen, el tipo de ESC y el voltaje que estos soportan, se procede entonces a escoger la batería que se adecue y satisfaga las exigencias de estos. La batería que se utilizara en el proyecto será una batería de LI-PO (Litio Ion Polímero). Esto es debido más que todo por el hecho de que la tasa de descarga de este tipo de baterías puede ser increíblemente alta, lo suficiente para satisfacer e inclusive sobrepasar las exigencias del prototipo. La batería seleccionada es entonces una Turnigy de 3S, esto quiere decir que posee tres celdas en serie de 3.7 V cada una para un Voltaje total de 11.1 V, con una capacidad de 2200 mAh y con una tasa de descarga o C de 20C (ver figura 42 y figura 43).



Figura 42: Batería de LiPo Turnigy de 2.2 A

Fuente: www.hobbyking.com(2018)



Figura 43: Batería de LiPo Turnigy de 2.2 A

Fuente: Mario Pontillo (2018)

4.1.4 Sistema de Comunicación

Hoy en día el uso masivo de teléfonos inteligentes y tabletas permite reducir costos en cuanto a dispositivos controlados por control remoto, haciendo innecesario el uso de otros tipos de sistemas de comunicación como lo es el radio control u otros sistemas en donde haya que construir un hardware o un dispositivo de control,

ahorrando tiempo y dinero, por esta razón para el sistema de comunicación, control y monitoreo se decidió crear una aplicación Android.

La idea original del proyecto era usar Wi – Fi ya que este tiene una velocidad de transmisión muy elevada, pero el protocolo Wi – Fi es más complejo de implementar y más costoso. Por tanto debido al tiempo limitado de este proyecto se optó por usar Bluetooth, ya que este protocolo es muy sencillo.

El modulo seleccionado es un Bluetooth Stick HC-06. Éste módulo funciona con UART (TX/RX) y permite establecer una comunicación inalámbrica con una velocidad de transferencia de entre 9600 y 115200 baudios. El alcance aproximado son unos 20 metros. Tiene que ser alimentado con una tensión de 3,3V – 5 V (ver figura 44).

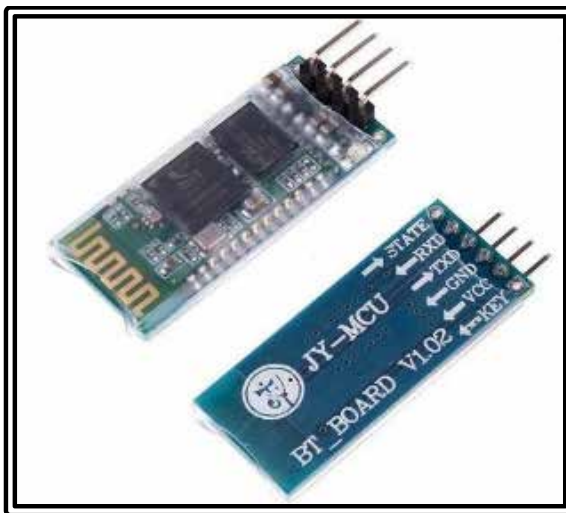


Figura 44: Bluetooth Stick HC – 06
Fuente: www.luisllamas.es(2018)

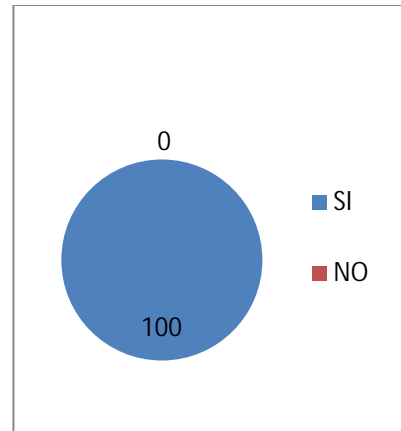
Para la aplicación Android hay muchos tipos de framework disponibles como Bootstrap, CSS, Java Script y HTML que son bastante potentes, pero por razones de tiempo y de simplicidad se optara por utilizar MIT App Inventor 2 que es un entorno de programación intuitiva y visual que permite crear aplicaciones Android totalmente funcionales para teléfonos inteligentes y tabletas en mucho menos tiempo que los entornos de programación tradicionales como los anteriormente mencionados.

4.2 Fase II. Diseñar la estructura, hardware y software del prototipo

Se analizaron los resultados de la encuesta como instrumento de recolección de datos, los mismos sirvieron para sustentar este trabajo de investigación, identificar la aceptación de posibles usuarios del sistema, así como también, los aspectos más resaltantes o importantes que se implementarán en el diseño del Dron. A continuación, se muestra el análisis por ítem de la encuesta realizada.

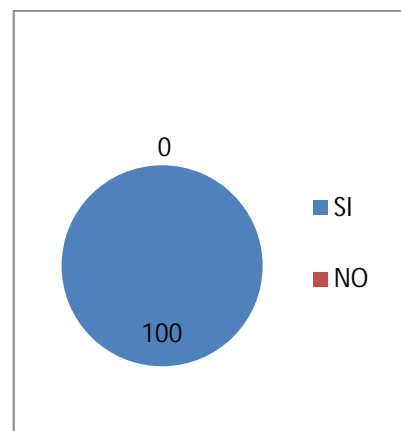
1. ¿Había oído hablar sobre los drones?

El 100% de la población tiene conocimiento de los drones (ver gráfico).



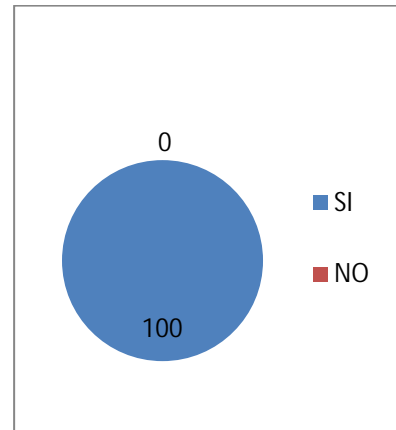
2. ¿Sabe lo que es un sistema de vigilancia?

El 100% de la población tiene conocimiento acerca los sistemas de vigilancia (ver gráfico).



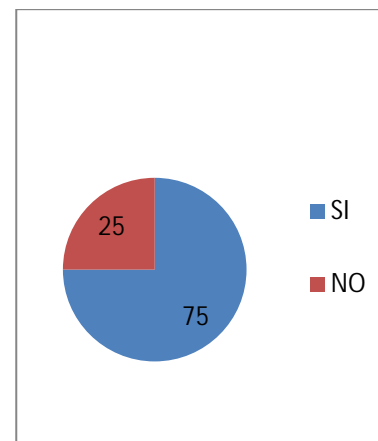
3. ¿Cree usted importante el diseño e implementación de tecnología nacional para el desarrollo del país?

El 100% de la población está de acuerdo que el desarrollo de tecnología nacional es vital para el progreso del país. (Ver gráfico).



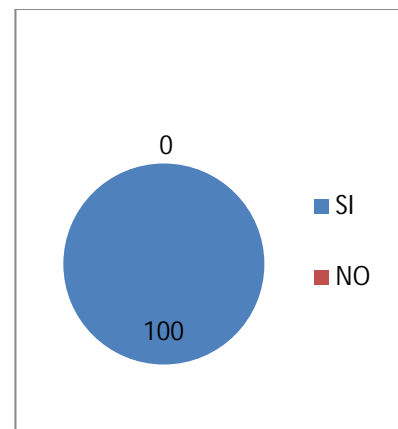
4. ¿Cree útil la implementación de un dron de vigilancia en los organismos de seguridad?

En el gráfico, se observa que un 75 % de la población cree que es de gran utilidad la implementación de estos dispositivos en los organismos de seguridad e inclusive rescate.



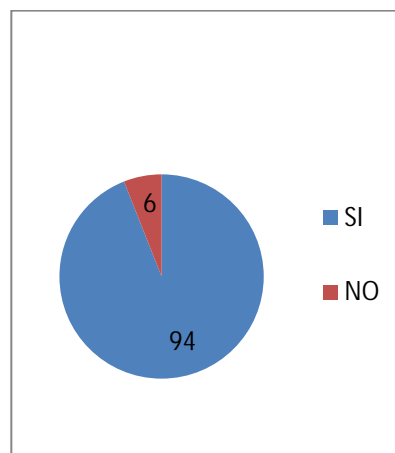
5. ¿Cree útil que un sistema de vigilancia avise a su móvil si detecto un posible intruso?

Al 100 % de los encuestados está de acuerdo y les gustaría que un sistema de vigilancia les avisara directamente en la comodidad de su móvil si detecta intrusos, lo cual afirma la necesidad de



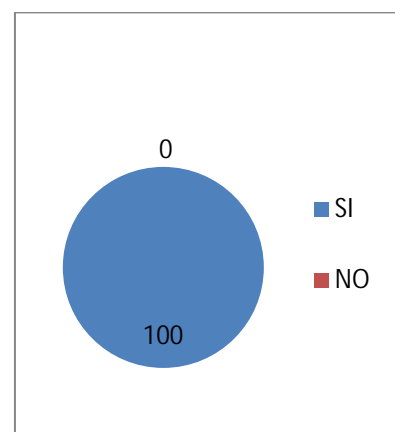
6. ¿Cree útil la existencia de Drones?

Al 100 % de los encuestados está de acuerdo y les gustaría que un sistema de vigilancia les avisara directamente en la comodidad de su móvil si detecta intrusos, lo cual afirma la necesidad de dotar el sistema con esta característica.



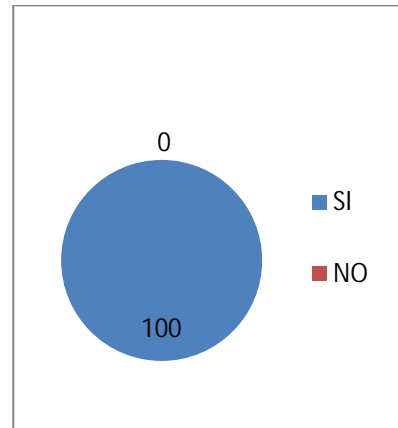
7. ¿Posee un teléfono inteligente o una tableta?

El 100% de los encuestados posee un teléfono inteligente o una tableta, lo cual garantiza un punto importante para la factibilidad de desarrollar este proyecto de investigación (ver gráfico).



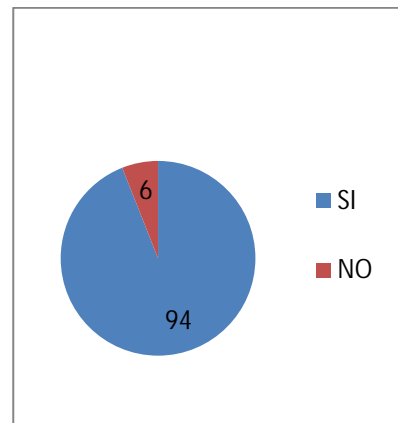
8. ¿Posee algún tipo de servicio de internet?

El 100% posee algún tipo de servicio de internet, lo cual al igual que en el ítem anterior, garantiza otro requisito importante para la factibilidad de desarrollar este proyecto de investigación (ver gráfico).



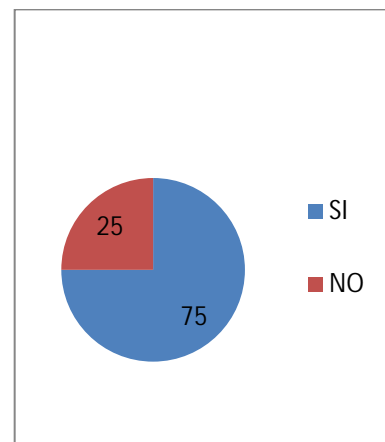
9. ¿Le gustaría poder supervisar el entorno donde reside desde un móvil?

El 94% está de acuerdo en utilizar un sistema de control y supervisión de manera remota con un dispositivo móvil (ver gráfico).



10. ¿ha pensado usted o algún conocido colocar un sistema de vigilancia en algún momento?

Como se puede observar en el grafico el 75% de la población está de acuerdo con la colocación de un sistema de vigilancia.



Al analizar los datos recolectados, se puede observar que existe el escenario propicio para implementar un dron de vigilancia, dado que hay un alto porcentaje de aceptación y también los potenciales usuarios disponen de los requisitos necesarios para ello, tales como servicio de internet y un teléfono inteligente o en su defecto una tableta. Luego a partir de esta encuesta, se procedió a diseñar el sistema pertinente, teniendo presente utilizar elementos que no eleven demasiado el costo de dicho sistema.

Al diseñar, las principales características a tener en cuenta son la resistencia y el peso. Pero desde luego, el peso es el parámetro crítico en el diseño, ya que a más peso se use más fuerza tendrán que hacer los motores, más capacidad de batería se necesitará y con lo que tendrá más peso aun.

Así pues para empezar a diseñar se debe saber cuál será el empuje máximo de los cuatros motores, para poder situar un punto de diseño a alcanzar. En este caso, que es un dron para la vigilancia, no es muy necesario que disponga de mucha aceleración, ni tampoco es de carga así que tampoco que sea extremadamente ligero, con situarse en el 50-60 % del empuje máximo es suficiente.

Obviamente cuanto menos peso mejor, pero esto puede incrementar enormemente el coste del proyecto, por tener que usar materiales extremadamente caros pero muy ligeros y resistentes como es la fibra de carbono.

Otra característica importante del diseño a tener en cuenta es la modularidad, es decir, que cada uno de los componentes sean reemplazables fácilmente y dependan lo mínimo del resto de las piezas, así se consigue optimizar el tiempo de construcción y el de reparación.

4.2.1 Diseño de la estructura

El primer aspecto que se debe decidir es la estructura que se va a utilizar para el dispositivo. En este caso se decidió utilizar una estructura en “+”, debido a que la estructura en forma “+” es más sencilla de controlar ya que para el desplazamiento solo es necesario actuar sobre 2 motores dejando los del eje perpendicular constantes.

4.2.1.1 Chasis

Para el chasis final del prototipo se optó por utilizar un chasis pre-fabricado, estos chasis son realmente económicos y de una calidad bastante aceptable, reduciendo así el tiempo y trabajo de fabricación de este. Otra de las razones por la que se escogió utilizar un chasis de este tipo es que esta fue hecha por máquinas y es mucho más ligera, precisa y simétrica en todos los aspectos, reduciendo perturbaciones y errores en el sistema (ver figura 45).

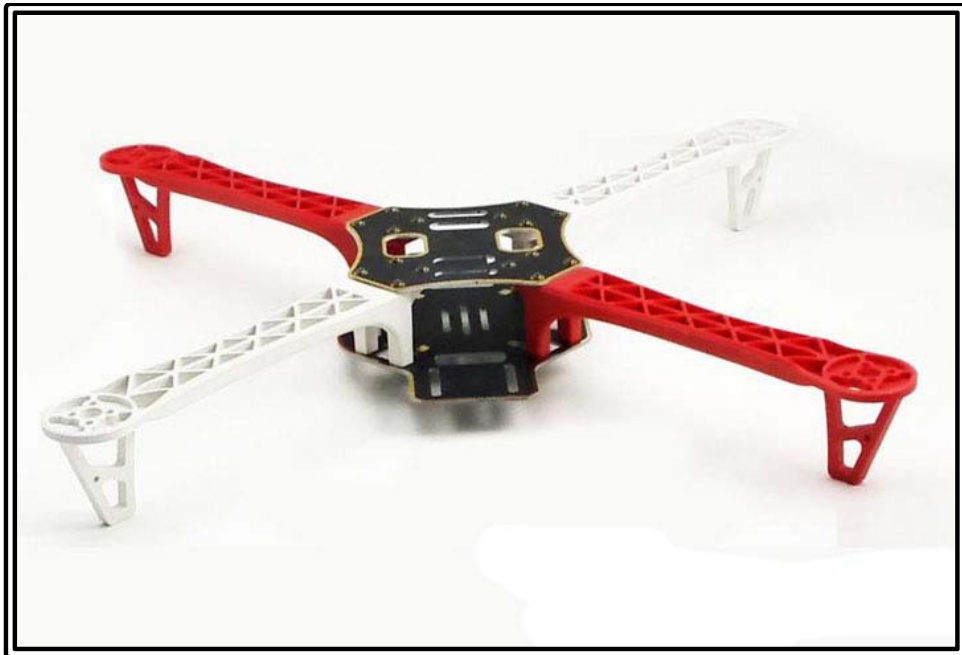


Figura 45: Chasis Frame Kit 450
Fuente: www.modeltronics.es(2018)

Este chasis esta hecho de nylon y carbono, tiene unas dimensiones de 45 cm diagonal y 32 cm de lado motor a motor y con una altura de unos 6 cm. El peso total de este chasis es de unos 282 g como se puede apreciar en la siguiente figura (ver figura 46).

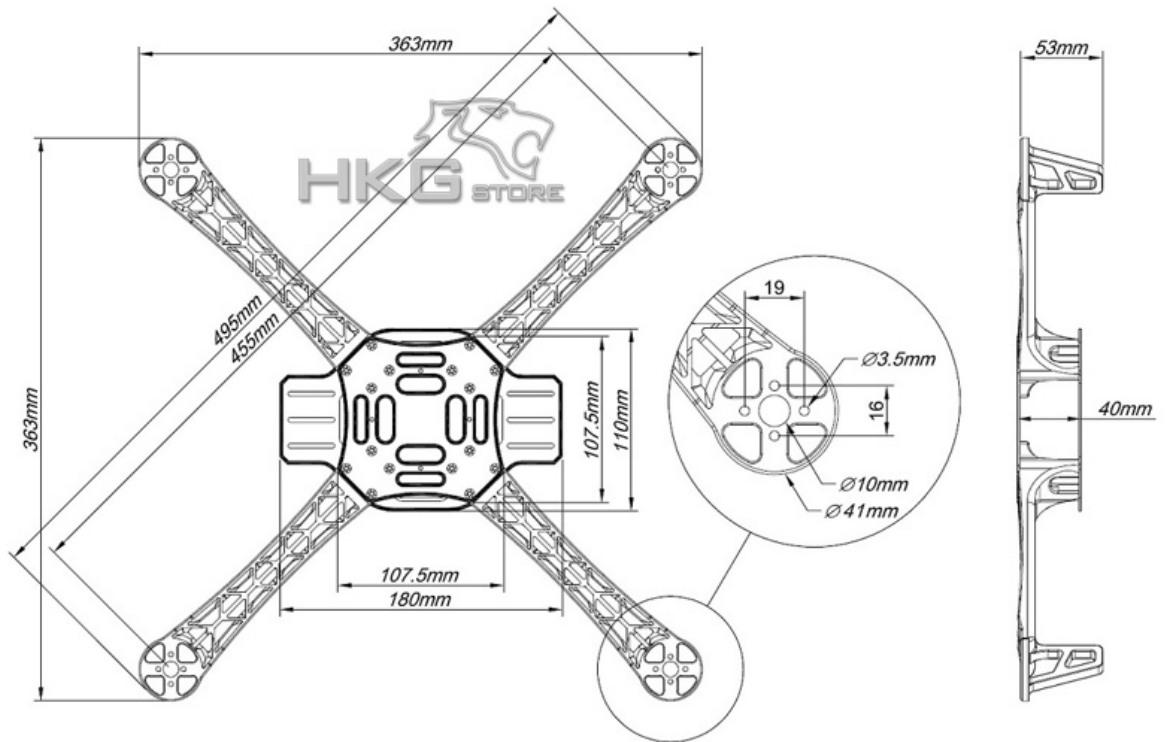


Figura 46: Dimensiones Chasis Frame Kit 450
Fuente: www.modeltronics.es(2018)

4.2.1.2 Hélices y porta hélices

Se han escogido hélices bipala especiales para Quadcopter, en concreto el modelo de 9"x4.5" (miden 9 pulgadas y tienen 4.5 pulgadas de paso) de plástico. Están unidas al motor usando un porta hélices de 3 mm, así se podrá remplazar cualquier hélice de manera sencilla, rápida, y minimizar las vibraciones del sistema.

Debido a la configuración del Quadcopter debe haber un par de hélices que giran en sentido horario (llamadas normales) y otro par que giran en sentido anti horario (llamadas inversas). Esta configuración es así con el fin de controlar el giro en eje de Yaw.

4.2.2 Diseño del Hardware

Una vez que se sabe, se diseña y se tiene el tipo de estructura y toda la parte mecánica del dispositivo, como se logró y explicó en el apartado anterior, entonces se

procede al diseño del hardware, que no es más que el diseño y conexión de toda la circuitería electrónica , tanto de sensorica como de comunicación y potencia. Como se mencionó en la fase anterior el hardware se puede dividir en:

- CPU
- Sensorica
- Sistema de propulsión
- Sistema de Comunicación

Otra etapa que no se mencionó en la fase anterior, ya que en esta se habla es de los componentes de drones en general, pero que para este proyecto es muy importante y que se mencionara a continuación, es el elemento de vigilancia que se implementará en el dron. Dicho elemento es el HC-SR501, este componente es un sensor de infrarrojo que detecta el movimiento o presencia de un cuerpo o comúnmente llamado PIR, este es un elemento muy utilizado en los sistemas de vigilancia por ser muy efectivo y de bajo costo. En principio se planeaba utilizar este componente junto a una cámara de alta definición con visión nocturna, como lo es la cámara PI NOIR para Raspberry Pi , Esta es una cámara que no posee filtro de infrarrojos haciéndola ideal para labores nocturnos y trabajar junto al HC-SR501, pero por la avería de la misma en los momentos de experimentación y que los recursos son muy limitados en este trabajo de grado , también debido a la situación que se vive en el país, se tuvo la obligación de eliminarse del prototipo e implementar solo el PIR. Aunque este componente junto con el GPS Ublox NEO 6M, que se mencionó en la fase anterior, y una alarma (como un buzzer) hacen un buen equipo también a lo que a vigilancia refiere, pero de esto se hablara más adelante. A continuación se detallaran las especificaciones técnicas del HC-SR501.

En los sensores de movimiento, el sensor PIR consta en realidad de 2 elementos detectores separados, siendo la señal diferencial entre ambos la que permite activar la alarma de movimiento. En el caso del HC-SR501, la señal generada

por el sensor ingresa al circuito integrado BISS0001, el cual contiene amplificadores operacionales e interfaces electrónicas adicionales.

Las funciones y ajustes complementarios del sensor de movimiento son:

- Ajuste de sensibilidad
- Ajuste de distancia de detección

Esto lo logra mediante 2 potenciómetros que el usuario puede ajustar para modificar tanto la sensibilidad como la distancia de detección del PIR a gusto. Como se indicó anteriormente, el rango de detección de movimiento de los PIR es ajustable y generalmente funcionan con alcances de hasta 7 metros, y con aperturas de 90° a 110° , como se muestra en la figura (ver figura 47).

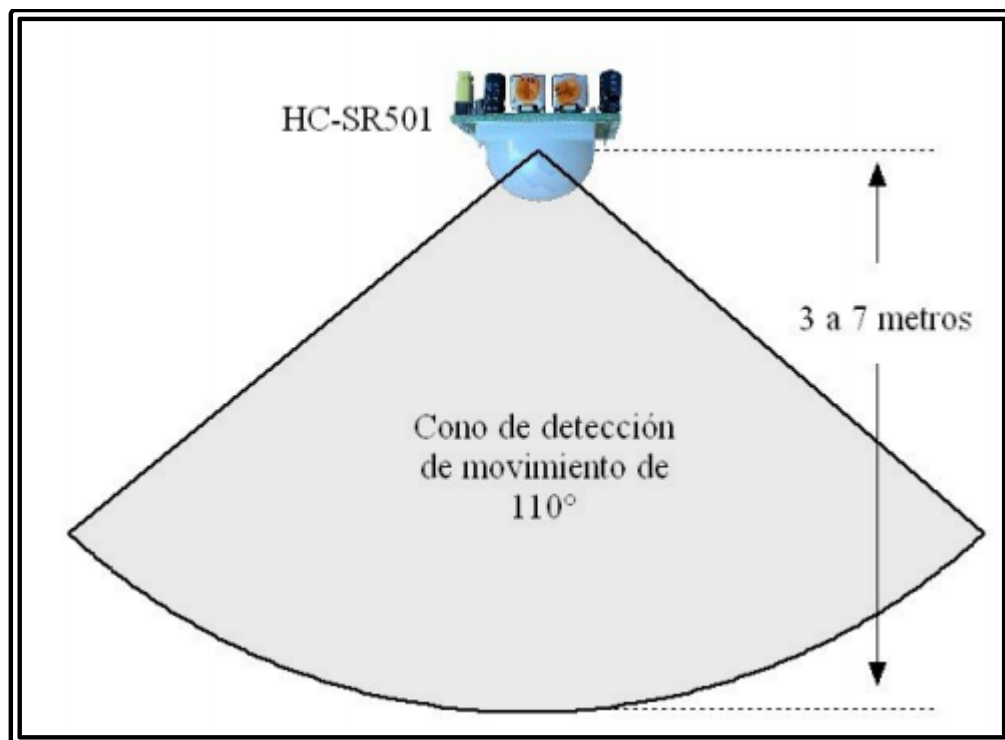


Figura 47: rango de detección HC-SR501

Fuente: www.luisllamas.com

El módulo PIR modelo HC-SR501 es de bajo costo, pequeño, e incorpora la tecnología más reciente en sensores de movimiento. El sensor utiliza 2 potenciómetros y un jumper que permiten modificar sus parámetros y adaptarlo a las necesidades de la aplicación (ver figura 48), como lo son sensibilidad de detección, tiempo de activación, y respuesta ante detecciones repetitivas.

Sus especificaciones técnicas son:

- Usa el PIR LHI778 y el controlador BISS0001
- Voltaje de alimentación: de 5 a 12 VDC
- Consumo promedio menor a 1 mA
- Rango de distancia de 3 a 7 metros ajustable.
- Angulo de detección: cono de 110°
- Ajustes: 2 potenciómetros para ajuste de rango de detección y tiempo de alarma activa.
- Jumper para configurar la salida de alarma en modo mono-disparo o disparo repetitivo.
- Salida de alarma de movimiento con ajuste de tiempo entre 3 segundos a 5 minutos.

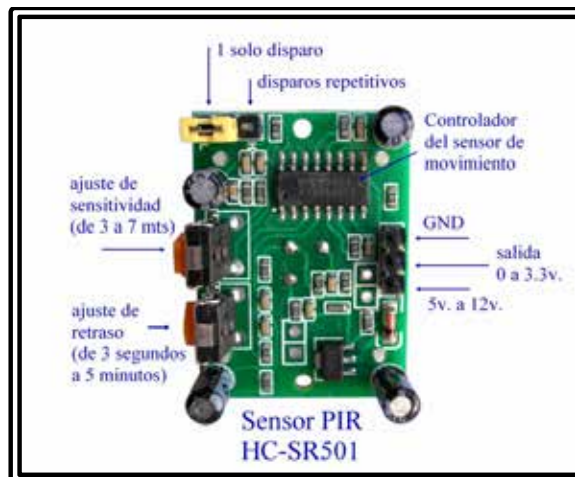


Figura 48: PIR HC-SR501

Fuente: manual de usuario Sensor infrarrojo de movimiento PIR HC-SR501 (2017)

Teniendo en cuenta lo anteriormente mencionado, se puede entonces dividir este diseño en dos partes, para así simplificar, estructurar y tener un mejor orden en el diseño. Las partes de este diseño del hardware son las siguientes:

- La digital.
- La de potencia.

Donde el sistema digital constara del CPU, Sensorica, Sistema de Comunicación y por último el sistema de vigilancia. Mientras que el sistema de potencia constara entonces del Sistema de Propulsión, como se mencionó en la fase I de este trabajo de grado.

4.2.2.1 Diseño del sistema digital

Como se sabe gracias a la fase anterior el CPU, la Sensorica y el Sistema de comunicación e inclusive los componentes de vigilancia se pueden agrupar en el diseño de la parte digital del hardware del prototipo, los componentes de cada una de estas etapas de hardware se pueden encontrar en la fase 1 de este trabajo de grado. En la siguiente figura (ver figura 49) se puede observar el diagrama de conexiones de toda la parte digital del hardware.

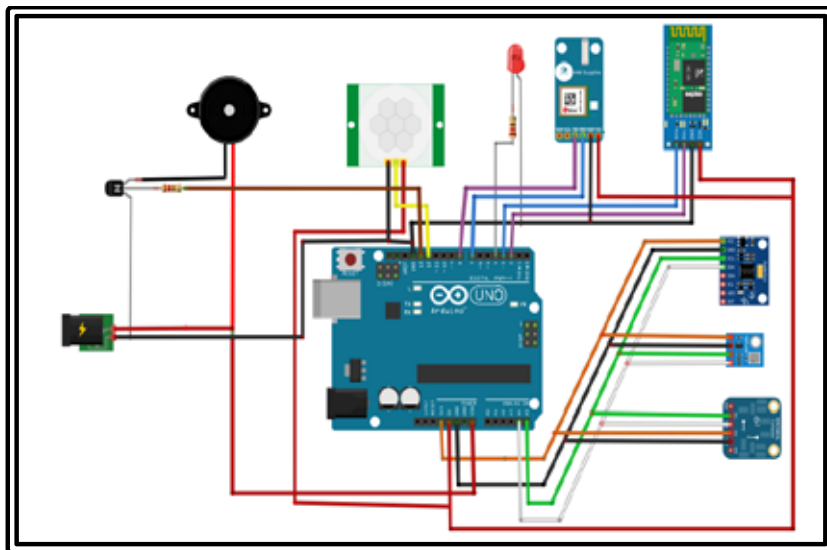


Figura 49: Diagrama de conexión digital en Fritzing
Fuente: Mario Pontillo (2018)

Como se puede apreciar en el diagrama de conexiones tenemos 3 elementos conectados en paralelo y esos elementos son:

- MPU6050
- BMP180
- HMC5883L

Estos son los sensores encargados de proporcionar la estabilidad al dron en todos sus grados de libertad. Dichos sensores utilizan el protocolo de comunicación I2C, por ellos su conexión en paralelo, esta comunicación posee 4 pines y estos son los siguientes:

- VCC
- GND
- SCL
- SDA

Los pines de VCC y GND son para la alimentación del módulo en sí, para el funcionamiento de todos los componentes que este posee. SCL y SDA son los encargados de la transmisión síncrona de los datos, donde básicamente SCL es el bus de reloj del sistema y SDA el bus de datos. Esta comunicación fue creada principalmente por esta característica de poder conectar una gran cantidad de dispositivos de en paralelo, haciendo muy sencillo su uso, solo es necesario saber las direcciones que posee cada dispositivo conectado, y entonces comenzar a transmitir con estos, aunque no simultáneamente, sino con el que se desee consultar en el momento. En el Arduino estos pines SCL Y SDA son los pines A5 y A4 respectivamente.

El GPS Ublox NEO 6M y el módulo de bluetooth HC-06 utilizan otro protocolo de comunicación, como se puede apreciar en la figura. Este protocolo es el UART (universally asynchronous receiver/transmitter) o comunicación Serie. Esto quiere decir que se necesita un pin para la transmisión de datos (TX) y un pin para la recepción de datos (RX) y obviamente al igual que los dispositivos con I2C, el VCC

y GND para la alimentación de los mismos, donde el pin TX de uno tiene que ir en el pin RX del otro dispositivo y viceversa con el de RX. Este protocolo es asíncrono y este a diferencia del I2C y la desventaja de este hardware es que se necesitan utilizar más pines en el microcontrolador por cada dispositivo que se quiera implementar que utilice este protocolo, ósea se tienen que declarar diferentes puertos UART por cada dispositivo, esto quiere decir un TX y RX para cada uno. Esta es una implementación software de un dispositivo USB para microcontroladores USB, haciendo posible construir hardware USB con casi cualquier MCU AVR, sin necesidad de un chip adicional. Se trata de un puerto USB software. En el caso de este proyecto los pines declarados en el Arduino como TX y RX para el HC-06 son el pin digital 3 y 2 y para el GPS los pines declarados son el 7 y 8 respectivamente. El Arduino en si ya posee un par de pines digitales con puerto Serie por defecto, estos pines son el 0 y 1, pero en el proyecto se decidió no utilizarlos debido a que estos pines son los mismos que los del USB del microcontrolador con los que se utilizan para cargar el programa o sketch en él. Se tiene un diodo LED conectado en serie con una resistencia de 220 ohm en el pin digital 4, cuya función es la de indicar que se ha tenido una conexión vía bluetooth exitosa con algún otro dispositivo remoto como lo es un teléfono inteligente.

Por ultimo están los elementos de seguridad, compuesto por un PIR que al detectar la presencia de un cuerpo este emite una señal digital que será leída por el Arduino en el pin 12, que fue configurado como entrada digital. El buzzer será utilizado como alarma en el caso de ser detectado un intruso y está conectado a través de un transistor NPN para liberar el Arduino de la carga del mismo, evitar averías en el microcontrolador y obtener más potencia en el Buzzer para un mayor sonido. La base de dicho transistor va conectada en serie a una resistencia de 1 Kohm para evitar una corriente excesiva e innecesaria en la base del transistor y va conectada en el pin 13 del Arduino que fue configurado como salida digital.

4.2.2.2 Diseño del sistema de potencia

La parte de potencia lo conforman los siguientes componentes:

- Los motores
- Los ESC
- Las baterías de LIPO
- Circuito medidor de carga

En la siguiente figura (ver figura 50) se aprecia el diagrama de conexiones de todo el sistema de potencia del hardware.

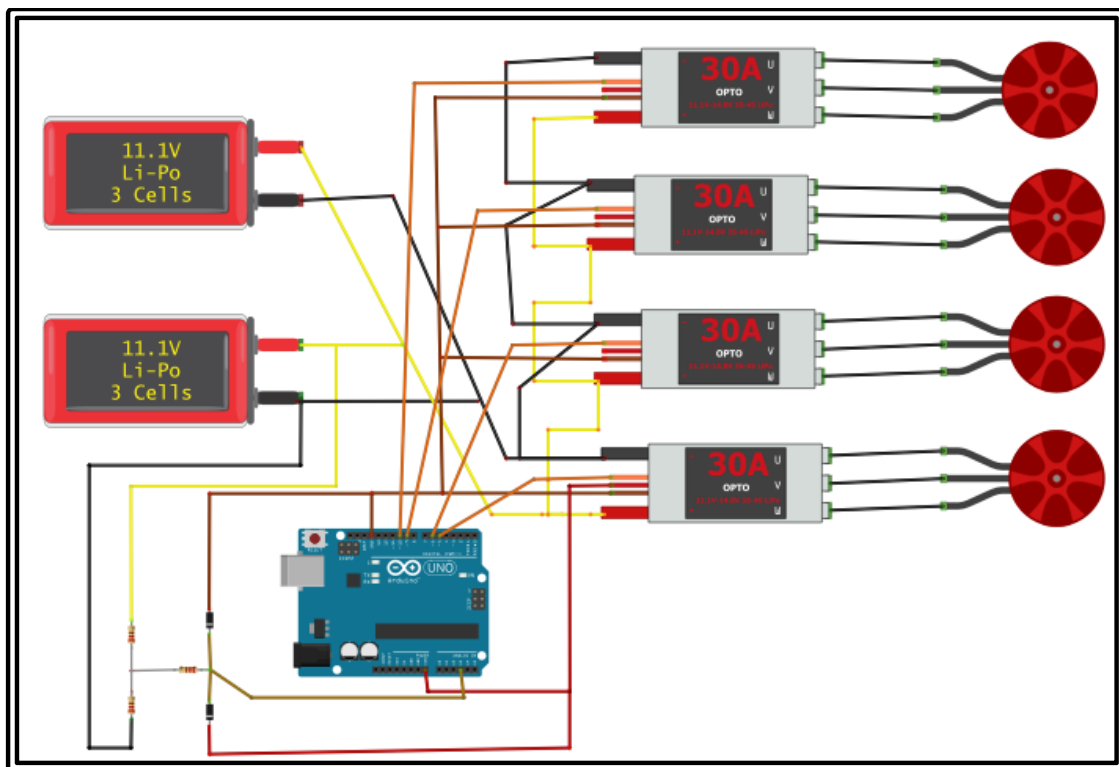


Figura 50: Diagrama de conexión de potencia en Fritzing
Fuente: Mario Pontillo (2018)

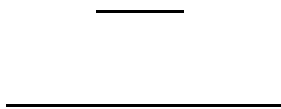
En el diagrama de conexiones se logra observar que los ESC que aparecen son de 30 Ampere, pero esto no es cierto, los ESC reales que se escogieron para el proyecto son de 20 Ampere, se colocaron esos en el diagrama ya que el programa Fritzing no poseía ningún ESC con la tecnología de los reales que fuesen de 20 A, entonces solo con el fin de ilustrar las conexiones, se obvió ese hecho.

Los ESC Emax BLHeli de 20 A del proyecto poseen una tecnología llamada BEC (Battery Eliminator Circuit) esta tecnología nos brinda una salida de 5 V y 2 A en su pin de cable color rojo, esto es bastante útil para el proyecto porque no es necesario el diseño, ni la implementación de una fuente de poder en el dron ahorrando tiempo, costo, trabajo, peso y espacio en el proyecto. Como se observa en la figura solo se utiliza la salida BEC de uno de los ESC debido a que la corriente de 2 A que nos proporciona es más que suficiente para alimentar todo el sistema digital del hardware. Se conecta la salida BEC (el cable rojo del ESC) al pin de Vin del Arduino para energizarlo y así no necesitar mantenerlo conectado con el USB a la computadora. Con el Arduino energizado y con su fuente interna se alimenta el resto de la sensorica y componentes. El cable marrón del ESC es el GND y el naranja es la entrada de la señal PWM proveniente del Arduino. Las entradas de señal PWM de los ESC se conectaron a los pines 5, 6, 9 y 10 del Arduino respectivamente, ya que estos son pines compatibles con salida PWM. Sin mucho tecnicismo se puede decir que las baterías alimentan los ESC, a su vez los ESC regulan el voltaje de dichas baterías y se alimenta toda la parte digital y a su vez estos invierten el voltaje de la batería para arrancar los motores.

Como se muestra en el diagrama de conexiones de potencia, hay dos baterías conectadas en paralelo, se decidió llevar a cabo el diseño de esta forma debido a que, se sabe de la fase I que cada motor consume un aproximado de 15 A cuando estos trabajan a su empuje máximo, por lo que al ser 4 motores se tiene un total de consumo aproximado de 60 A. la batería que más se ajustaba a las necesidades del proyecto y que a su vez era la económicamente más accesible es la Turnigy de 3S, 2200 mAh con una tasa de descarga de 20C. Al realizar los cálculos con una sola batería se obtiene que:

Esto quiere decir que una batería con estas características solamente puede entregar un máximo de 44 A. Es allí donde reside la batería extra y la conexión en paralelo de ambas, debido a que los 4 motores pueden llegar a consumir un total de 60 A, claro está que la idea no es llegar a ese límite, ya que ese escenario ocurriría si los motores trabajasen a toda máquina, pero siempre se debe diseñar en base al máximo consumo.

Al colocar 2 baterías de este tipo en paralelo se obtiene una cantidad de corriente de 88 A, estando muy por encima de las exigencias máximas de los motores, dando así gran holgura aumentando la vida de las baterías. Otra de las ventajas que se obtiene de esto, es que se suman las capacidades de las baterías, motivo por lo que ahora se cuenta con una capacidad de 4400 mAh, aumentando el tiempo de vuelo del dron. Para tener una referencia del tiempo de vuelo del dron se llevó a cabo un cálculo, asumiendo el escenario en que los motores estuvieran trabajando al máximo:



Esto quiere decir que con el máximo empuje aplicado, el dron tendrá un tiempo de vuelo aproximado de 4, 4 minutos. Los ESC al ser de 20 A cada uno también se sobredimensionan con respecto al consumo de los motores, así se evitan los recalentamientos y se alarga la vida de los componentes.

Por ultimo se encuentra el circuito medidor de batería que no es más que un divisor de tensión que permite medir la tensión de la batería. Según las indicaciones de su carcasa su tensión habitual de funcionamiento es de 11.1V y según las curvas de carga habituales de las baterías LiPo de esas características, 9V es un

voltaje representativo de batería descargada. Se añade un divisor de tensión 1:4 para que se obtenga un valor máximo de 3V y un valor mínimo de 2.25 estando así el valor centrado entre los límites de la entrada analógica del Arduino. Se monitoriza la medida a través del ADC concretamente mediante la entrada A3 del Arduino. Además del divisor de tensión se incluye una resistencia serie y dos diodos que aseguran el rango de voltaje entre 0V y 5V y limitan la entrada de corriente.

Los valores elegidos para el divisor de tensión, además de establecer una proporción 1:4, son de resistencias de 100 Kohm y otra de 33 Kohm, estos valores están pensados para que no suponga un consumo extra de potencia, para ello el nivel de impedancia es relativamente elevado (ver figura 51).

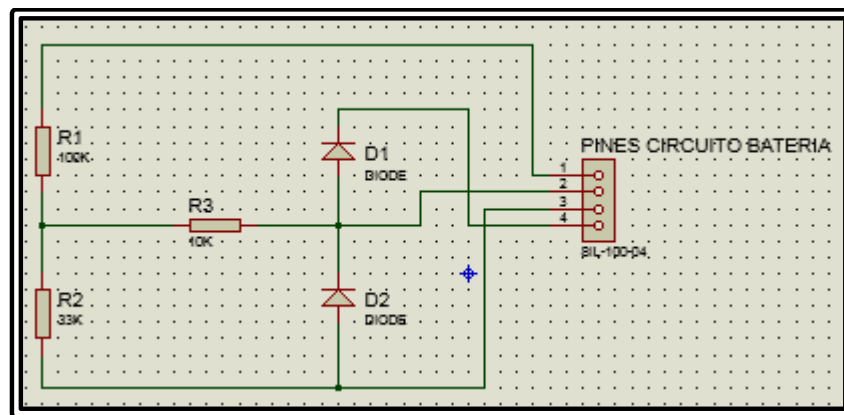


Figura 51: esquema circuito medidor de batería en Proteus 8
Fuente: Mario Pontillo (2018)

Teniendo en cuenta que el máximo valor en el punto intermedio va a ser 3V a causa del divisor de tensión, se estima la corriente máxima que puede llegar a entrar por el terminal analógico, aunque en la práctica será bastante menor.

del procesador del que dispone Arduino, el ATmega 328P, en el Datasheet se indica que las entradas analógicas están optimizadas para una resistencia de salida de

4.2.2.3 Configuración Final del Hardware

Se muestra el esquema hecho en Proteus 8 donde se observa a grandes rasgos como es el conexionado del hardware diseñado (ver figura 52).

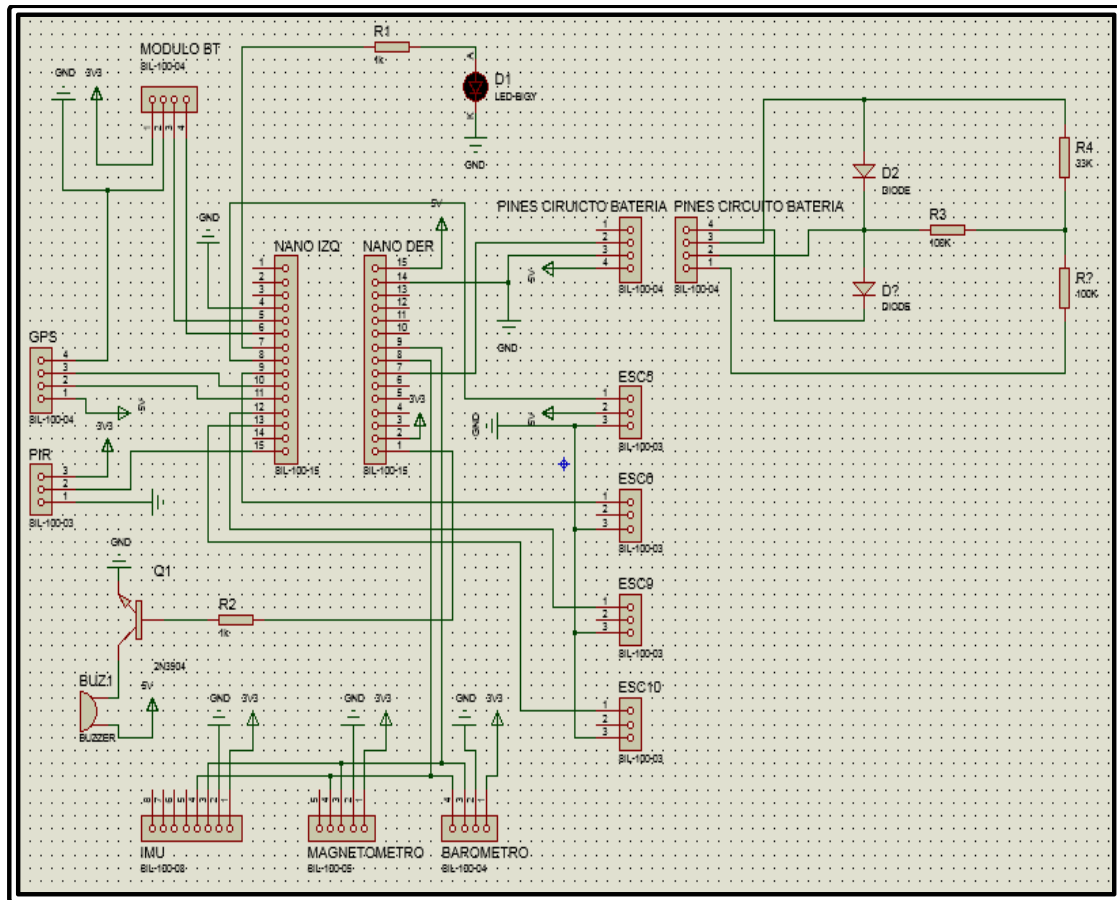


Figura 52: esquema electrónico del Hardware completo en Proteus 8
Fuente: Mario Pontillo (2018)

También se ha hizo un diagrama de bloques de este para tener una visión global y conjunta del sistema (ver figura 53).

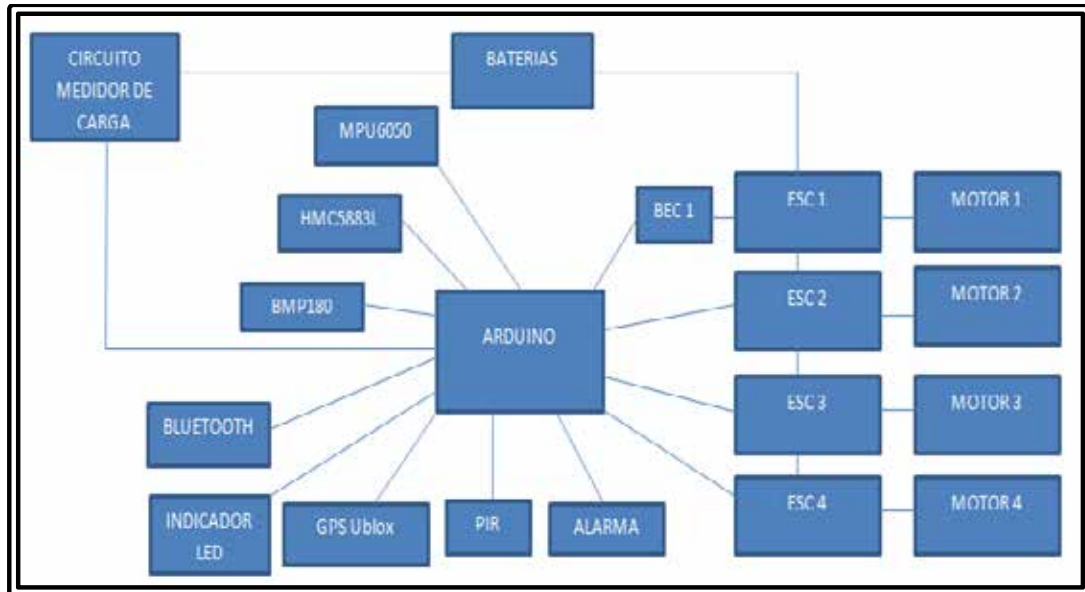


Figura 53: Diagrama de bloques generalizado del Hardware
Fuente: Mario Pontillo (2018)

4.2.3 Diseño del Software

4.2.3.1 Entorno de programación.

La propia plataforma Arduino proporciona un IDE liviano y básico, pero muy funcional brindando las herramientas necesarias para subir, depurar y comunicarse con la propia placa. No obstante, existen otros entornos de programación, algunos también livianos y otros bastante robustos, complejos y más completos. Algunos de estos entornos de programación son los siguientes:

- Scratch para Arduino (S4A)
- Eclipse
- Visual Studio

- Codebender
- Python

El entorno que se escogió para el desarrollo del proyecto fue el IDE proporcionado por el equipo Arduino, que posee un lenguaje de programación basado en Wiring y un entorno de desarrollo basado en Processing. La interfaz es muy similar a la del Processing e incorpora todas las librerías que la propia Wiring posee. La ventaja de utilizar este IDE y por la que se escogió, es que este tiene un gran abanico de librerías que simplifican enormemente cualquier tarea a realizar, gracias a la gran comunidad Arduino que existe y debido a que es un entorno de desarrollo libre, como por ejemplo lectura y escritura de los puertos digitales y analógicos, implementación de los protocolos de comunicación como SPI, I2C, UART y permitir el control de Timers y PWMs. Estas son de gran utilidad ya que permite el desarrollo de un proyecto de una forma rápida y sencilla.

Al abrir el programa de Arduino se puede apreciar que tiene dos funciones o elementos (ver figura 54).



```
sketch_jun20a Arduino 1.6.3
Archivo Editar Programa Herramientas Ayuda
sketch_jun20a
1 void setup() {
2   // put your setup code here, to run
3
4 }
5
6 void loop() {
7   // put your main code here, to run r
8
9 }
```

Figura 54: Funciones principales en IDE Arduino
Fuente: Mario Pontillo (2018)

Estas funciones son las siguientes:

- **Setup:** es una función que solo se ejecuta una vez al inicio del programa, y es aquí donde se deben hacer las tareas de inicialización de variables, abrir canales de comunicación (puerto serie, ethernet, etc...). Es decir, es donde se debe cargar toda la configuración y estado inicial del proyecto a realizar.
- **Loop:** es ejecutar un bucle infinito. Este ejecuta lo que este en dicha función. Es decir, es el cuerpo principal del programa del proyecto, pero esto se ejecutará infinitamente.

La estructura básica de un programa en Arduino, son estas dos funciones, todo lo demás, son clases y funciones extras, librerías, entre otros... que se añadirán en el transcurso de realizar este proyecto.

4.2.3.2 Diseño del programa en Arduino del Dron

En cuanto al diseño del programa se decidió usar un diseño modular o estructurado, basándose en la POO (Programación Orientada a Objetos), ya que esta simplifica y optimiza el diseño del programa. Las características más importantes que ofrece este tipo de programación y las que se ponen en práctica en el presente trabajo de grado son las siguientes:

- Abstracción
- Encapsulamiento
- Polimorfismo
- Herencia
- Modularidad

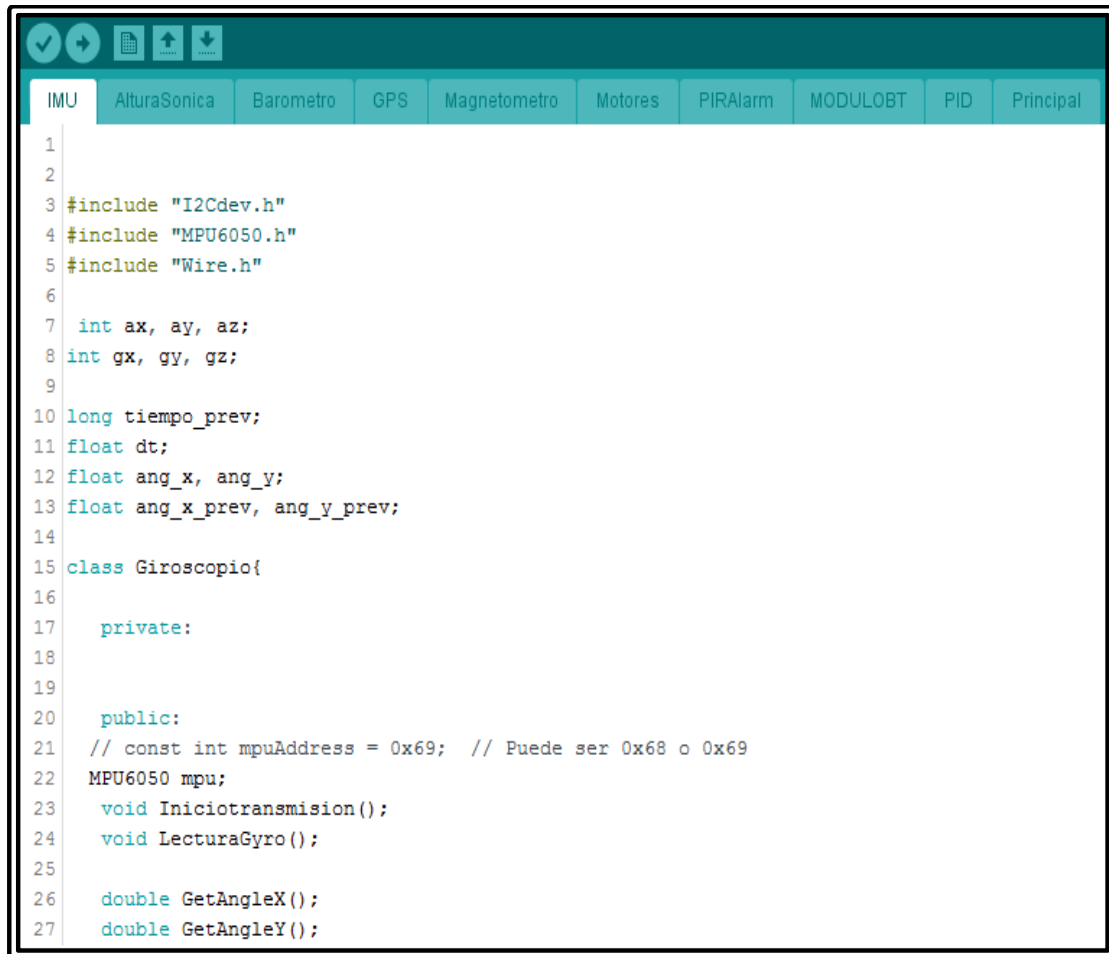
Al programar de esta forma se recicla una enorme cantidad de código, sin mencionar que vuelve mucho más ordenado y comprensible el programa a la hora de revisiones. En este programa se subdividió el Sketch en partes más pequeñas, cada una de las cuales son independientes del Sketch en sí y de las partes o módulos restantes. La ventaja que proporciona, y una de las razones por la que se optó por

diseñar de esta forma, es que estos módulos se pueden compilar por separado, pero tienen conexión con el resto de los módulos, muy parecido a la encapsulación, para así ahorrar memoria dinámica del microcontrolador. En pocas palabras cada clase es independiente una de la otra y la hora de necesitarse o de consultar algo de dicha clase, se instancia el objeto de esta y se llama la función deseada en cualquier punto en la ejecución del código o programa, liberando al Arduino de trabajo extra.

En total el programa posee 10 módulos:

- IMU
- AlturaSonica
- Barómetro
- GPS
- MODULOBT
- Magnetómetro
- Motores
- PIRAlarm
- PID
- Principal

En la siguiente figura (Ver figura 55) se pueden observar los módulos anteriormente mencionados, la estructura e independencia de estos en el IDE de Arduino.



```
1
2
3 #include "I2Cdev.h"
4 #include "MPU6050.h"
5 #include "Wire.h"
6
7 int ax, ay, az;
8 int gx, gy, gz;
9
10 long tiempo_prev;
11 float dt;
12 float ang_x, ang_y;
13 float ang_x_prev, ang_y_prev;
14
15 class Giroscopio{
16
17     private:
18
19
20     public:
21     // const int mpuAddress = 0x69; // Puede ser 0x68 o 0x69
22     MPU6050 mpu;
23     void Iniciotransmision();
24     void LecturaGyro();
25
26     double GetAngleX();
27     double GetAngleY();
```

Figura 55: Módulos del programa en IDE Arduino
Fuente: Mario Pontillo (2018)

El módulo de AlturaSonica se programó debido a que en un principio se planeaba utilizar el sensor ultrasónico HC-SR04 para la detección de obstáculos y paredes para que el dron pudiera evitar accidentes autónomamente. El modulo se mantuvo debido a fines de ampliaciones y mejoras futuras, pero este no es utilizado, por ende no será mencionado en el resto de esta fase.

En la siguiente figura (ver figura 56) se muestra el árbol de clases y la jerarquía o herencia de estas:

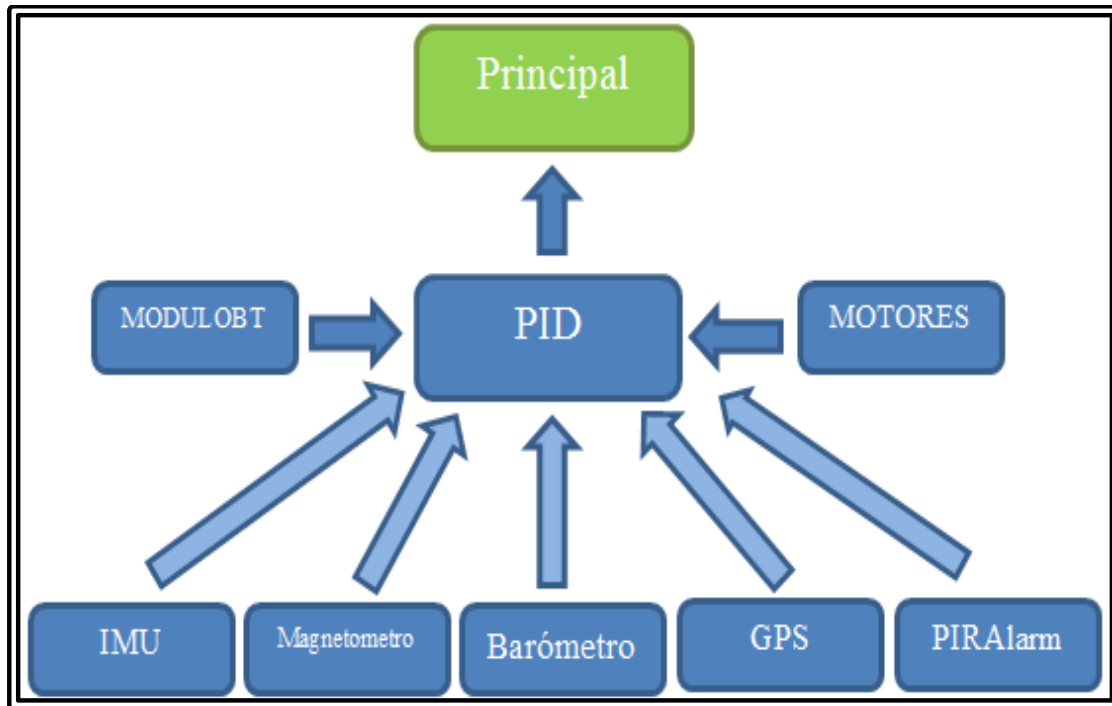


Figura 56: Árbol y jerarquía de clases del programa
 Fuente: Mario Pontillo (2018)

4.2.3.3 Modulo IMU

Este módulo se encarga de importar las librerías necesarias, configurar e inicializar el sensor MPU6050 por medio del protocolo I2C. En este también se hacen las lecturas del Acelerometro y Giroscopio que posee el MPU6050, también realiza los cálculos necesarios para obtener las velocidades angulares escaladas, los respectivos ángulos de inclinación y por último se acondicionan dichos valores mediante un filtro complementario. El modulo posee también otras funciones que utiliza para retornar los resultados obtenidos. Todo el proceso anteriormente mencionado se dividió en diferentes funciones asociadas a una clase, para que luego se pueda instanciar dicha clase y llamar la función deseada. La respectiva sección de código del módulo se observa en la figura (ver figura 57).



```
34 };
35
36 void Giroscopio::Iniciotransmision()
37 {
38     {
39
40     Wire.begin();
41     mpu.initialize();
42     //Serial.println(mpu.testConnection() ? F("IMU iniciado correct
43     }
44
45     void updateFiltered()
46 {
47     dt = (millis() - tiempo_prev) / 1000.0;
48     tiempo_prev = millis();
49
50     //Calcular los ángulos con acelerometro
51     float accel_ang_x = atan(ay / sqrt(pow(ax, 2) + pow(az, 2)))*(1
52     float accel_ang_y = atan(-ax / sqrt(pow(ay, 2) + pow(az, 2)))*(
53
54     //Calcular angulo de rotación con giroscopio y filtro complemen
55     ang_x = 0.98*(ang_x_prev + (gx / 131)*dt) + 0.02*accel_ang_x;
56     ang_y = 0.98*(ang_y_prev + (gy / 131)*dt) + 0.02*accel_ang_y;
57
58     ang_x_prev = ang_x;
59     ang_y_prev = ang_y;
60 }
```

Figura 57: Código Modulo IMU en IDE Arduino
Fuente: Mario Pontillo (2018)

4.2.3.4 Modulo Barómetro

En este módulo se hace básicamente lo mismo que el de IMU, la única diferencia es que el sensor es un BMP180 y este es un sensor que lee la temperatura y presión, luego con esos valores se hace el cálculo pertinente y se estima la altitud. Para este caso se acondiciona la señal, no por medio de un filtro con en el caso del módulo IMU, sino que básicamente se toman una cantidad de lecturas y se hace un

promedio de ellas, para que el resultado sea un poco más preciso. La respectiva sección de código del módulo se observa en la figura (ver figura 58).



```
7 };
8
9
10
11 void Barometrico::SetupBaro()
12 {
13     if (bmp180.begin())
14     {
15         Serial.println("BMP180 iniciado correctamente Tomando medidas del punto de referencia");
16         status = bmp180.startTemperature(); //Inicio de lectura de temperatura
17         if (status != 0)
18         {
19             delay(status); //Pausa para que finalice la lectura
20             status = bmp180.getTemperature(Temperatura); //Obtener la temperatura
21             if (status != 0)
22             {
23                 status = bmp180.startPressure(3); //Inicio lectura de presión
24                 if (status != 0)
25                 {
26                     delay(status); //Pausa para que finalice la lectura
27                     status = bmp180.getPressure(Presion, Temperatura); //Obtenemos la presión
28                     if (status != 0)
29                     {
30                         Po=Presion; //Asignamos el valor de presión como punto de referencia
31                         Serial.println("Punto de referencia establecido: h=0");
32                     }
33                 }
34             }
35         }
36     }
37 }
38
39 guardado.
```

Figura 58: Código Modulo Barómetro en IDE Arduino
Fuente: Mario Pontillo (2018)

4.2.3.5 Modulo GPS

Este módulo es el encargado de configurar un Software USB por medio de la librería SoftwareSerial, este declara un par de pines del Arduino como TX y RX para ser utilizados como puerto UART, para luego el mismo poder iniciar la comunicación con el GPS Ublox NEO 6M que utiliza este protocolo de comunicación.

El modulo GPS se le programo una cantidad de funciones cuya tarea es la de recibir la señal de los satélites y decodificarla, obtener la posición en coordenadas de longitud y latitud, la velocidad en m/s y km/h y por ultimo también la capacidad de poder guardar una posición determinada que se llamó Home y entregar la distancia entre la posición actual y el Home y el del curso para llegar a dicha posición. Todas las funciones retornan los respectivos valores anteriormente mencionados cada vez que se los solicitemos. La respectiva sección de código del módulo se observa en la figura (ver figura 59).



```
void GPSplus::SetupGPS() {
  SerialGps.begin(9600);
}

void GPSplus::LecturaGPS() {
  while (SerialGps.available() > 0)
    GPS.encode(SerialGps.read());
}

double GPSplus::Location() {
  if (GPS.location.isValid()) {
    Serial.print(F(" Lat="));
    Serial.print(GPS.location.lat(), 6);
    Serial.print(F(" Long="));
    Serial.println(GPS.location.lng(), 6);
    return(GPS.location.lat());
    return(GPS.location.lng());
  }

  if (GPS.location.isUpdated())
  {
    Serial.print(F(" Lat="));
    Serial.print(GPS.location.lat(), 6);
  }
}
```

Figura 59: Código Modulo GPS en IDE Arduino
Fuente: Mario Pontillo (2018)

4.2.3.6 Modulo Magnetómetro

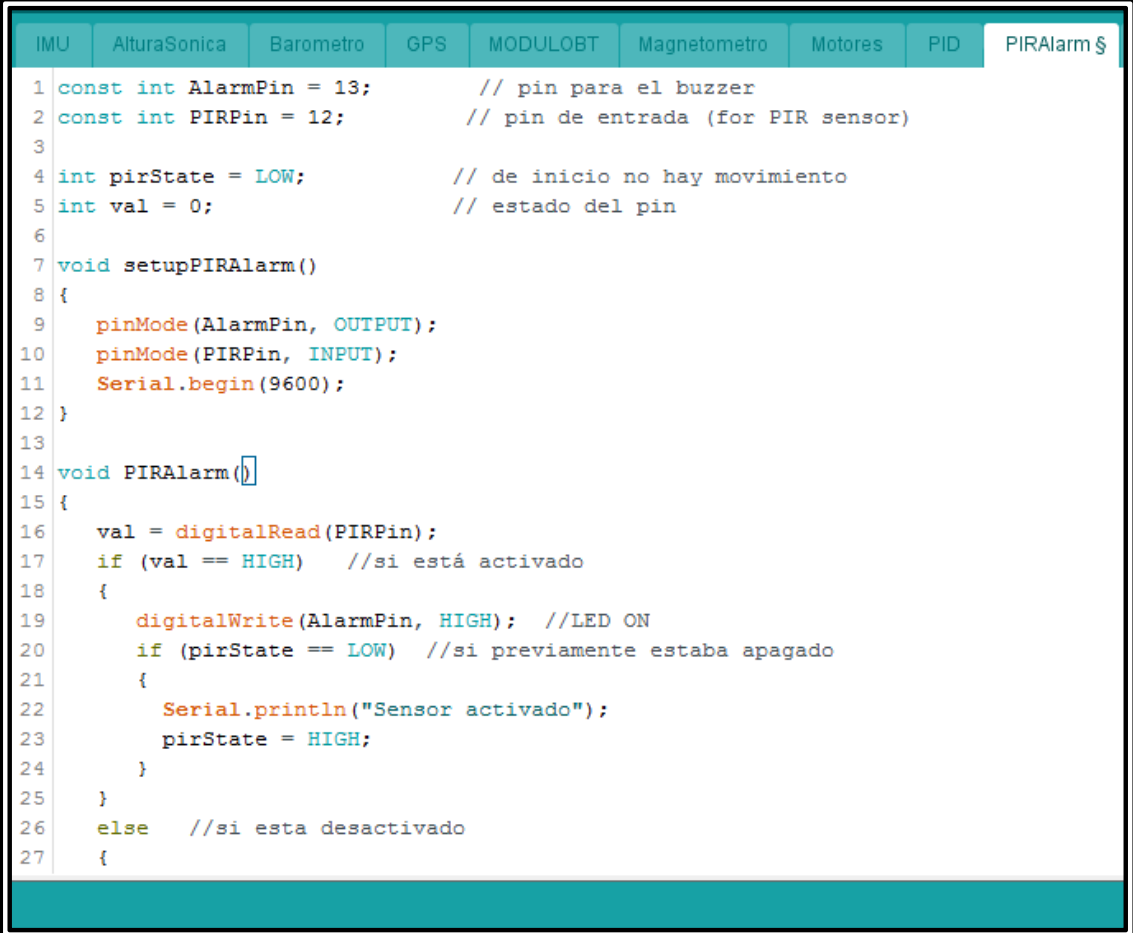
Como en los módulos IMU y Barómetro este se encarga de configurar el protocolo de comunicación I2C e inicia la transmisión del sensor HMC5883L. Básicamente el modulo se encarga de calibrar el HMC5883L, de obtener las lecturas de los distintos vectores de campo magnético y luego calcular el ángulo del Yaw con respecto al norte geográfico y al igual que el módulo IMU, este aplica un filtro complementario entre el ángulo obtenido por medio del HMC6883L y el ángulo obtenido por medio del MPU6050 para una medición más precisa. Este módulo también posee funciones que devuelven el valor del ángulo del Yaw acondicionado y la velocidad angular en ese eje cuando sean necesarias. La respectiva sección de código del módulo se observa en la figura (ver figura 60).

```
1
2 #include "HMC5883L.h"
3
4
5
6 class Magnetometro{
7
8   private:
9
10  int16_t mx, my, mz;
11  const float declinacion = 1;
12  long tiempo_prev;
13  float anguloprev;
14  float angulo;
15  float Zspeed;
16
17  public:
18  HMC5883L compass;
19  void SetupMag();
20  void Brujula();
21  float LecturaZ();
22  float LecturaSpeedZ();
23
24 };
25
26
27 void Magnetometro::SetupMag()
```

Figura 60: Código Modulo Magnetómetro en IDE Arduino
Fuente: Mario Pontillo (2018)

4.2.3.7 Modulo PIRAlarm.

La programación de este módulo fue bastante sencilla, debido a que el PIR HC-SR501 es un sensor muy básico y sencillo de usar, pues solo posee un pin donde arroja una señal de salida, donde es un HIGH si se detecta presencia y LOW si el sensor no ha detectado nada. Este módulo solo se encarga de declarar la entrada digital en el Arduino para recibir dicha señal, de generar la alarma y retornar un valor booleano si se detecta el movimiento. Este módulo también realiza la tarea de disparar la posición actual del dron y la distancia desde el HOME al usuario en el caso de detectar un intruso. La respectiva sección de código del módulo se observa en la figura (ver figura 61).



```
IMU | AlturaSonica | Barometro | GPS | MODULOBT | Magnetometro | Motores | PID | PIRAlarm $
1 const int AlarmPin = 13;          // pin para el buzzer
2 const int PIRPin = 12;           // pin de entrada (for PIR sensor)
3
4 int pirState = LOW;              // de inicio no hay movimiento
5 int val = 0;                     // estado del pin
6
7 void setupPIRAlarm()
8 {
9     pinMode(AlarmPin, OUTPUT);
10    pinMode(PIRPin, INPUT);
11    Serial.begin(9600);
12 }
13
14 void PIRAlarm()
15 {
16     val = digitalRead(PIRPin);
17     if (val == HIGH) //si está activado
18     {
19         digitalWrite(AlarmPin, HIGH); //LED ON
20         if (pirState == LOW) //si previamente estaba apagado
21         {
22             Serial.println("Sensor activado");
23             pirState = HIGH;
24         }
25     }
26     else //si esta desactivado
27     {
```

Figura 61: Código Modulo PIRAlarm en IDE Arduino
Fuente: Mario Pontillo (2018)

4.2.3.8 Modulo MODULOBT.

Este módulo es el que se encarga de gestionar todo el tráfico de datos entre la aplicación y el dron. Al igual que el GPS Ublox NEO 6M el Stick de bluetooth HC-06 utiliza protocolo de comunicación UART o Serie, por ende este módulo al igual que el modulo GPS configura un puerto UART en el Arduino e inicializa la comunicación entre el HC-06 y el mismo Arduino. El MODULOBT posee funciones que se encargan básicamente de recibir los Bytes desde la aplicación al dron, guardar dichos Bytes de datos y luego retornar esos valores para que puedan ser utilizados por las estructuras de control del programa para que finalmente el dron realice lo que el usuario demanda. Este también escribe los valores sensados por el dron y los envía a la aplicación, así el usuario siempre puede estar al tanto del estatus del dron y de lo que ocurre en el entorno. La respectiva sección de código del módulo se observa en la figura (ver figura 62).

```
1 #include <SoftwareSerial.h>
2 SoftwareSerial BT(2,3); //rx,tx
3
4 class Comunicacion {
5     private:
6     char Bytes;
7
8     public:
9
10    void SetupBT();
11    char ReceivedBytes();
12 };
13
14 void Comunicacion::SetupBT(){
15     BT.begin(9600);
16 }
17
18 char Comunicacion::ReceivedBytes(){
19
20     if ( BT.available() ){
21         Bytes = BT.read();
22     }else{
23         Bytes = 0;
24     }
25     return(Bytes);
26
27 }
```

Guardado

Figura 62: Código Modulo MODULOBT en IDE Arduino

Fuente: Mario Pontillo (2018)

4.2.3.9 Modulo Motores

Para poder usar los motores, hay que tener claro el protocolo y firmware que usan los ESC para el inicio y calibrado, por ende primero se procedió a entender y explicar dicho protocolo y luego explicar la tarea del módulo y de cómo se programó, para que se tenga un mejor entendimiento del mismo.

Para el inicio normal se conecta el ESC a la alimentación del sistema (normalmente mediante la conexión de las baterías). El ESC detecta el valor mínimo del PWM, y hace un largo sonido en forma de "beep". Entonces el sistema detecta el voltaje de la batería y hace varios sonidos cortos en forma de "beep", lo que denota el número de células de la batería y finalmente el ESC lleva a cabo la auto-

Para el procedimiento de calibrado se conecta el ESC a la alimentación del sistema, el ESC detecta el PWM máximo y hace dos sonidos en forma de "beep", lo que denota que el Max PWM ha sido confirmado y se guarda. (Si se mantiene pulsado durante más de 2 segundos después de este paso, se lleva al modo de programación). Después de 2 segundos el sistema detecta la mínima señal del PWM y hace un largo sonido en forma de "beep". A continuación el ESC detecta el voltaje de la batería y hace varios sonidos cortos en forma de "beep", lo que denota el número de células de la batería. Finalmente el ESC lleva a cabo la auto-comprobación. Si es

constantemente sonidos en forma de " beep". Cualquier fallo en la auto-comprobación, hará que suene 20 sonidos cortos en forma de "beep".

Una vez que se conoce este protocolo, se procede a programar el modulo, en el cual consiste solamente de 3 funciones. La primera función se encarga de la calibración de los ESC como se explicó anteriormente, se les aplica una señal de PWM máximo a todos los ESC y se aguardan 2 segundos a que los ESC verifiquen y guarden el valor, luego se les aplica la señal de PWM mínima igualmente a todos los ESC y se aguardan nuevamente 2 segundos. Las otras dos funciones son las encargadas de escribir el valor de PWM o de escribir a los ESC la cantidad de

potencia que recibirán los motores. Se hacen dos funciones ya que cada funciones escribe para cada par de motores, esto quiere decir que una función se encarga de los motores del eje X mientras que la otra función de los motores del eje Y. La respectiva sección de código del módulo se observa en la figura (ver figura 63).

```
22
23 void Drivers::Calibracion() {
24
25 ESCXL.attach(MOTORXL);
26
27 ESCXL.write(pwmhigh);
28 delay(2500);
29 ESCXL.write(pwmlow);
30 delay(2500);
31
32 ESCXR.attach(MOTORXR);
33
34 ESCXR.write(pwmhigh);
35 delay(2500);
36 ESCXR.write(pwmlow);
37 delay(2500);
38
39 ESCYL.attach(MOTORYL);
40
41 ESCYL.write(pwmhigh);
42 delay(2500);
43 ESCYL.write(pwmlow);
44 delay(2500);
45
46 ESCYR.attach(MOTORYR);
47
48 ESCYR.write(pwmhigh);
49 delay(2500);
```

Figura 63: Código Modulo Motores en IDE Arduino
Fuente: Mario Pontillo (2018)

4.2.3.10 Modulo PID

Para el control del dron se decidió utilizar un control PID (Proporcional - Integral -Derivativo) y esto se debe a que en un mundo ideal el dron podría volar estable sin tanto problema y para manejar o controlar los distintos grados de libertad (Yaw, Pitch, Roll y Throttle) simplemente se usaría una matriz de comandos para obtener la consigna a aplicar a cada motor.

Pero existen algunas cosas que hacen que sea imposible o poco probable realizar esto en la vida real. Las perturbaciones tales como el viento, como las diferencias en los motores, hélices, desbalance en la estructura misma, provocan que el comportamiento del Quadcopter en la vida real sea ruidoso y variable. La Matriz de comandos no toma esto en cuenta, y la mente, ojos y manos podrían no ser lo suficientemente rápidas para reaccionar a estos cambios en el momento.

Se necesita que el prototipo tenga un cierto grado de autonomía. En particular, que este pueda auto nivelarse y para que pueda lograrse se utilizan técnicas de control. Por ende para este proyecto se optó por diseñar un una estructura común para el control de realimentación como lo son los PID, ya que es la estructura de control más utilizada y de la que se tiene más conocimiento y práctica.

Este módulo se encarga de crear el objeto PID, para el control de cada uno de los ángulos de navegación del dron (Yaw, Pitch y Roll). Como se planea que el prototipo se ubique a una altitud ingresada en la aplicación por el usuario, también se crea un cuarto objeto PID que controle el Throttle. En este módulo, también se configuran dichos PID, esto quiere decir que el modulo coloca el valor de K_p , K_i y K_d , el valor de SetPoint, el modo de operación del controlador, el del tiempo de muestreo y por ultimo de aplicar un rango limite a la salida u Output del PID, esto con el fin de eliminar el problema del Wind Up. El modulo realiza este proceso para cada uno de los PID existentes en el sistema. El modulo se encarga de todo el proceso de control, básicamente de leer toda la información obtenida por los sensores BMP180, MPU6050 y HMC5883L, que son los sensores que otorgan la orientación, para que luego cada objeto PID existente pueda realizar el computo pertinente. Este

por ultimo procede a distribuir los Output, salidas o resultados de los computo de cada PID a los motores correspondientes para mantener estable y orientado el Quadcopter. Esto quiere decir que leerá los ángulos y altitud del Dron en todo momento y si hay un cambio en la lectura, que no sea la de los Setpoint establecidos en el sistema de control o por el usuario mediante la aplicación, como lo es una perturbación externa, el PID procederá a corregirla automáticamente manteniendo este horizontal y estable. Este módulo se conecta con el de MODULOBT, llamado ciertas funciones de este para poder leer los Bytes enviados por la aplicación. Esto se hace debido a que, todo lo que proviene desde la aplicación y llega al dron, son Bytes o datos que refieren al movimiento y ubicación del prototipo, por ende se leen directamente en el módulo PID para que el sistema de control realice los ajustes para que el dron haga los movimientos solicitados.

El diagrama de bloques generalizado del controlador para un eje de navegación programado se muestra en la figura (ver figura 64).

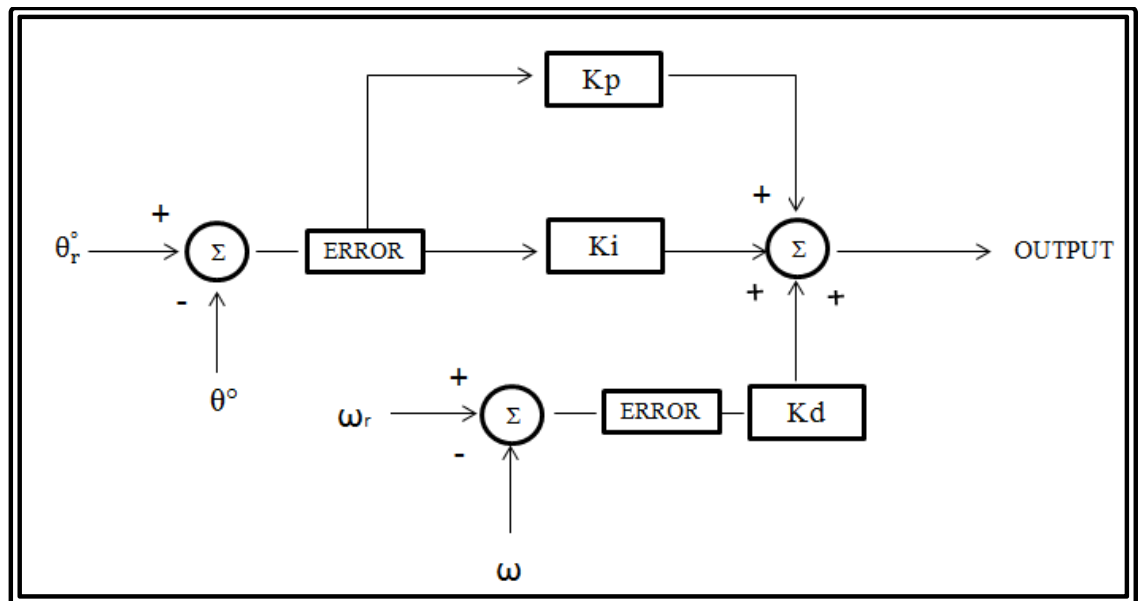


Figura 64: Diagrama de bloques generalizado del controlador en un solo eje de navegación
Fuente: Mario Pontillo (2018)

El diagrama de bloques de la figura, es el del controlador PID de un eje de navegación, por ende se programó un controlador igual al mostrado en la figura para cada eje, esto quiere decir para Yaw, Pitch y Roll.

El diagrama de bloques generalizado del sistema de control completo es el mostrado en la figura siguiente (Ver figura 65).

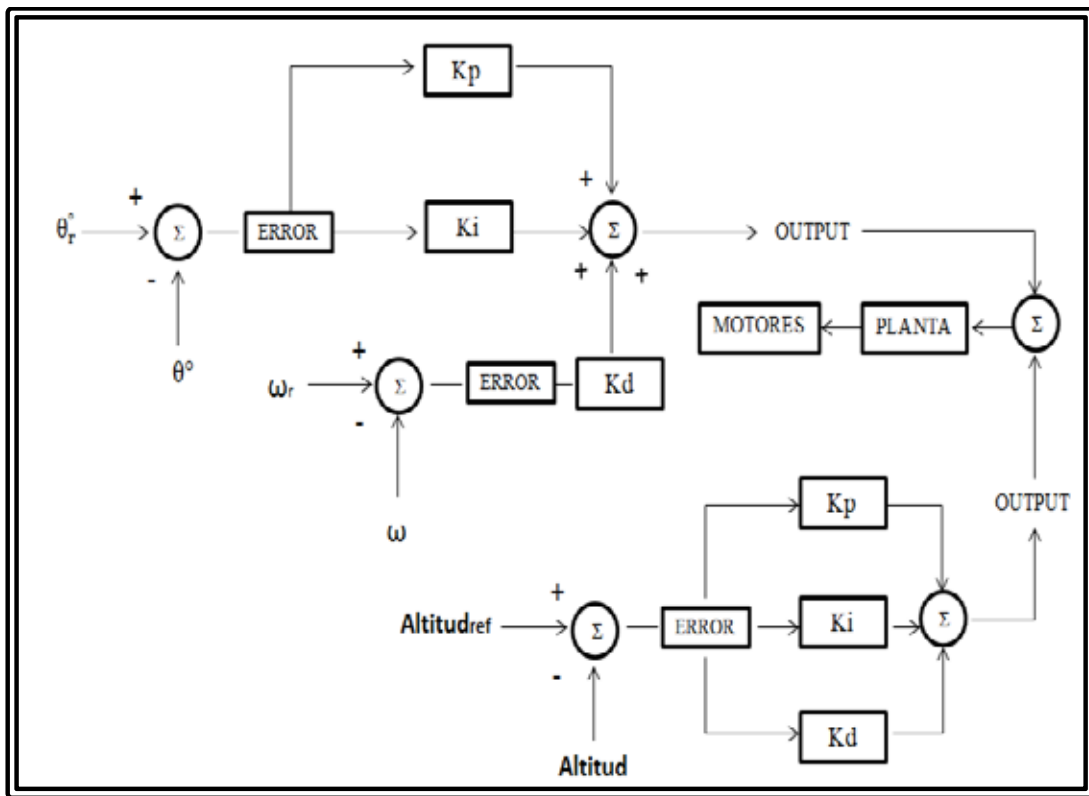


Figura 65: Diagrama de bloques generalizado del controlador
Fuente: Mario Pontillo (2018)

Como se puede observar en la figura solo se muestra el control de un eje, pero esto es debido a motivos de generalización e ilustración. Se sabe que en la realidad es uno para cada ángulo de navegación y las salidas de dichos controladores se suman con el sistema de control de altitud y la salida se envía a los motores a través de la matriz de comandos (ver figura 66).

	Norte	Sud	Este	Oeste
Throttle	+	+	+	+
Pitch	-	0	+	0
Roll	0	-	0	+
Yaw	-	+	-	+

Figura 66: Matriz de comandos

Fuente: TDG Sergio Moyano Díaz (2014, p.65).

Un + indica que el motor debe ser acelerado en respuesta a un comando de entrada positiva. Un - indica que el motor debe ser frenado en respuesta a un comando de entrada positiva. Un 0 indica que la velocidad del motor no se ve afectada por un comando de entrada positiva. La respectiva sección de código del módulo se observa en la figura (Ver figura 67).

```

40 double Kp=0.1, Ki=0, Kd=0.01;
41 double constKp=1.1, constKi=1.4, constKd=0.5;
42 double SetpointX = 2.5, SetpointY = -1.3, SetpointZ = 0, SetpointI = 0, SetpointWX = 0.08, SetpointWY = 0.06, SetpointWZ =
43 double OutputX, OutputY, OutputZ, OutputI, OutputWX, OutputWY, OutputWZ;
44 double InputX, InputY, InputZ, InputI, InputWX, InputWY, InputWZ;
45
46 PID myPIDZ(<InputZ, <OutputZ, <SetpointZ, KpZ, KiZ, KdZ, DIRECT);
47 PID myPIDY(<InputY, <OutputY, <SetpointY, KpY, KiY, KdY, DIRECT);
48 PID myPIDX(<InputX, <OutputX, <SetpointX, KpX, KiX, KdX, DIRECT);
49 PID myPIDWX(<InputWX, <OutputWX, <SetpointWX, KpW, KiW, KdW, DIRECT);
50 PID myPIDWY(<InputWY, <OutputWY, <SetpointWY, KpW, KiW, KdW, DIRECT);
51 PID myPIDI(<InputI, <OutputI, <SetpointI, constKp, constKi, constKd, DIRECT);
52
53
54 void Control::reconfiguracionX(){
55
56   myPIDX.SetMode(AUTOMATIC);
57   myPIDX.SetSampleTime(5);
58   myPIDX.SetOutputLimits(-1000, 1000);
59   myPIDWX.SetMode(AUTOMATIC);
60   myPIDWX.SetSampleTime(5);
61   myPIDWX.SetOutputLimits(-1000, 1000);
62   int i;
63   for ( i = 0; i <= 20; i++) {
64     GYRO.LecturaGyro();
65     SetpointY = GYRO.GetAngleX();
66     delay(100);

```

Figura 67: Código Modulo PID en IDE Arduino

Fuente: Mario Pontillo (2018)

4.2.3.11 Modulo Principal

Este módulo es simplemente el modulo principal que posee la función Setup y la función Loop que son las funciones principales que ejecutan todo el programa, como se explicó anteriormente en esta misma fase. Este módulo se encarga es de instanciar las clases de los diferentes módulos y llamar las funciones pertinentes para que estas se ejecuten y se obtenga el funcionamiento completo deseado del sistema, aparte también se decidió programar en este mismo modulo el medidor de la batería, ya que es un elemento que siempre se va a estar consultando y es una programación sencilla con pocas líneas de código, se vio innecesario crearle un módulo y clase aparte. En esta sección se lee el valor de la batería por medio de la entrada analógica y se imprime dicho valor a través del MODULOBT y se envía a la aplicación para su monitorización. La respectiva sección de código del módulo se observa en la figura (ver figura 68).

```
3 void setup() {
4   Serial.begin(9600);
5   BlueT.SetupBT();
6   GYRO.Iniciotransmision();
7   Compass.SetupMag();
8   BAROPressure.SetupBaro();
9   pinMode(ledconnect, OUTPUT);
10 }
11 void loop() {
12
13 BlueT.ReceivedBytes();
14
15 char STATUS = BlueT.ReceivedBytes();
16 if (STATUS == 'c') {
17   digitalWrite(ledconnect, HIGH);
18 }
19
20 while ( STATUS == 'c' ) {
21
22
23 // CONTROL.computeALL();
24 //CONTROL.computeZ();
25 // BAROPressure.PTA();
26 // Serial.println(BAROPressure.GetAltitude());
27 /*GYRO.LecturaGyro();
28 Serial.println( GYRO.GetOMEGAY());
29 /*Compass.Brujula();
```

Figura 68: Código Modulo Principal en IDE Arduino
Fuente: Mario Pontillo (2018)

4.3 Fase III. Desarrollar una interfaz de control y supervisión del Dron para los usuarios en una aplicación Android

En esta fase se desarrolló una aplicación Android, basada en el Framework MIT App Inventor 2 que es una estructura de desarrollo e implementación de aplicaciones elaborada por un equipo de estudiantes y profesores del MIT (Instituto

tecnológico de Massachusetts) y Google. Este es usado por una gran cantidad de organizaciones para el desarrollo de aplicaciones. Esta herramienta contiene plantillas de diseño con tipografía, formularios, botones, cuadros, menús de navegación y otros elementos de diseño. Este posee un compilador que traduce el lenguaje visual de los bloques para la aplicación en Android utiliza Kawa como lenguaje de programación, distribuido como parte del sistema operativo GNU de la Free Software Foundation. Uno de los beneficios más importante de usar App Inventor en el diseño de la aplicación Android, es que con él se pueden desarrollar aplicaciones adaptables, que se ajusten a cualquier dispositivo que se utilice para visualizarla. Otro beneficio importante es que sus plantillas están diseñadas tomando en cuenta que los colores, tamaños de objetos, fuente, etc., sean ergonómicos y agradables a la vista y que su lenguaje de programación basada en bloques Permite crear una aplicación en menos tiempo que otros, y se pueden programar aplicaciones más complejas en mucho menos tiempo que con los lenguajes más tradicionales, basados en texto.

Se diseñó un sistema de acceso a la página web, en donde se debe ingresar el usuario y la contraseña, esto para evitar que otra persona externa a los residentes de la vivienda pueda acceder al sistema domótico, en la figura 69 se muestra el diseño del formulario utilizado para acceder a las funciones de la aplicación web. En el caso de que se ingrese un usuario o contraseña incorrectos aparece una alerta de error que indica que la validación del usuario no se pudo realizar.



Figura 69: pantalla inicio de sesión
Fuente: Mario Pontillo (2018)

Se desarrolló una interfaz o pantalla principal donde se puede observar todo el estatus del Dron mientras este es controlado o se esté moviendo autónomamente en su recorrido de vigilancia, como lo son la carga de la batería, altitud y orientación de este, entre otros. En esta pantalla principal se dejó el centro libre y todos los elementos que muestran el estatus, botones y barras, se colocaron a los lados, debido a que en un futuro se planea implementar la cámara que se planeó implementar desde un inicio en el proyecto, que no se pudo debido a la avería de la misma en los momentos de experimentación, para que en todo ese espacio se muestre el video en tiempo real de lo que capta dicha cámara. Cabe destacar que esta aplicación es totalmente adaptable a cualquier dispositivo, lo cual se puede apreciar en la figura (ver figura 70) donde se observa que esta, se ajusta a la resolución de la pantalla de cada dispositivo.



Figura 70: Interfaz Principal
Fuente: Mario Pontillo (2018)

En la interfaz principal anteriormente mencionada también se tienen los controles manuales del Quadcopter y se programó para que la distribución fuese bastante cómoda de utilizar, parecida a la de un control con imagen, como la de una aplicación de juegos de autos.

En el diseño del sistema de configuración de los movimientos del prototipo, se tomaron en cuenta algunas consideraciones, pues como se mencionó anteriormente, el movimiento del dron puede lograrse de forma manual o autónoma, esto quiere decir escribiendo una posición en la que se desea que este vaya o recorra.

Al elegir entre manual y automático hay algunas opciones que se deshabilitan, por ejemplo, si el control es automático no hace falta usar los botones de mando de la aplicación, por lo tanto, se realizó un script con el App Inventor de validación para habilitar o deshabilitar campos del formulario dependiendo de las opciones que el usuario vaya eligiendo, dicho script se muestra en la figura (ver figura 71).

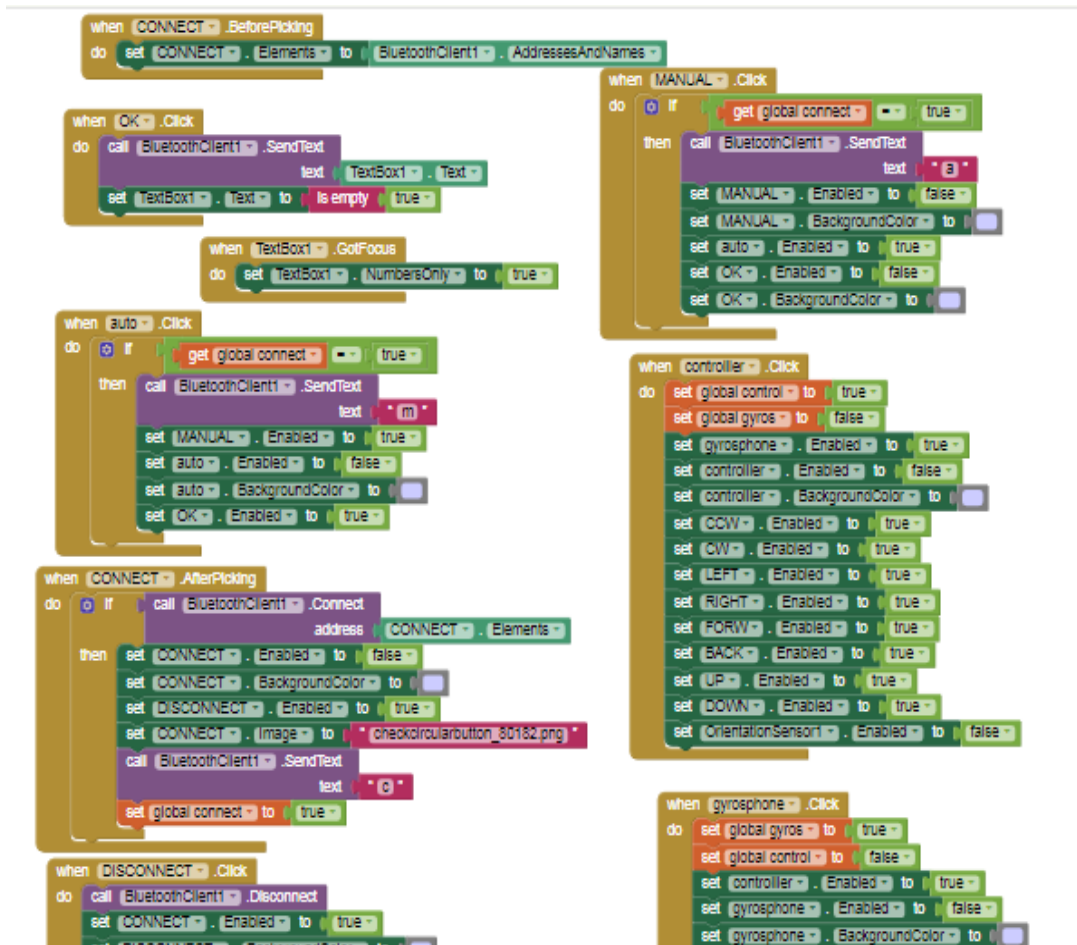


Figura 71: Código de programación en MIT App Inventor
Fuente: Mario Pontillo (2018)

Al elegir la forma automática se desarrolló un código que hace visible y habilita un cuadro con un par de cajas de texto, parecida a la de inicio de sesión, donde este solicita la Altura y la posición en coordenadas de latitud y longitud a ingresar, para luego ser leídos por el programa realizado en Arduino, el cual toma las acciones de control programadas para adaptarse a lo establecido por el usuario, en este caso, mover el Quadcopter a la posición deseada (ver figura 72).

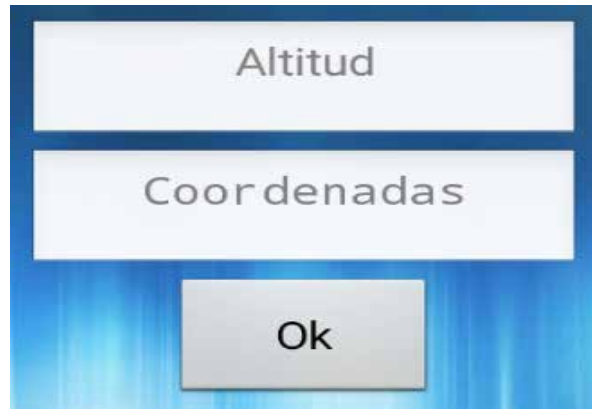


Figura 72: Cuadro
Fuente: Mario Pontillo (2018)

Por último se desarrolló una paleta de opciones desplegadas a la aplicación, donde posee los elementos que no son necesarios que sean visibles en todo momento en la interfaz principal por el usuario, como lo son los elementos que muestran el estatus del dron y el entorno, que si es necesario que permanezcan en una parte donde este visibles y puedan ser monitoreados siempre que se pueda. Algunas de las opciones de dicha paleta son la conexión y desconexión de la comunicación Bluetooth del dispositivo y la opción de solicitud de posición o coordenadas del dron (ver figura 73).



Figura 73: paleta Abierta
Fuente: Mario Pontillo (2018)

4.4 Fase IV. Implementar el dron inteligente diseñado

En esta fase se implementó el prototipo final de Dron diseñado en la Fase II. Se realizaron distintas pruebas y prototipos iniciales para la evolución de este, dichas pruebas fueron necesarias para saber que se iba por un camino correcto en el diseño y para obtener resultados y datos que sirvieran para seguir avanzando y mejorando el diseño final sin arriesgar y averiar los recursos que se poseían, ya que eran limitados, para la construcción del mismo e inclusive ocasionar daños colaterales, como a otras personas u objetos. Por ende la implementación fue evolucionando poco a poco desde prototipos iniciales seguros y no en una prueba con un modelo final que incluyese todos los recursos obtenidos, ya que así era mucho más arriesgado e irresponsable.

Sabiendo las principales variables que se necesitan controlar para un vuelo adecuado, se decidió diseñar e implementar por etapas. Para lograr esto se pensaron y realizaron distintos ensayos en este trabajo de investigación, y dichos ensayos son los siguientes:

- Ensayo con 1 Grado de Libertad
- Ensayo con 3 Grados de Libertad
- Ensayo con 4 Grados de Libertad

4.4.1 Ensayo con 1 Grado de Libertad.

En la siguiente figura (ver figura 74) se aprecia el prototipo diseñado que se utilizó para experimentar en este ensayo y obtener los datos suficientes para avanzar.

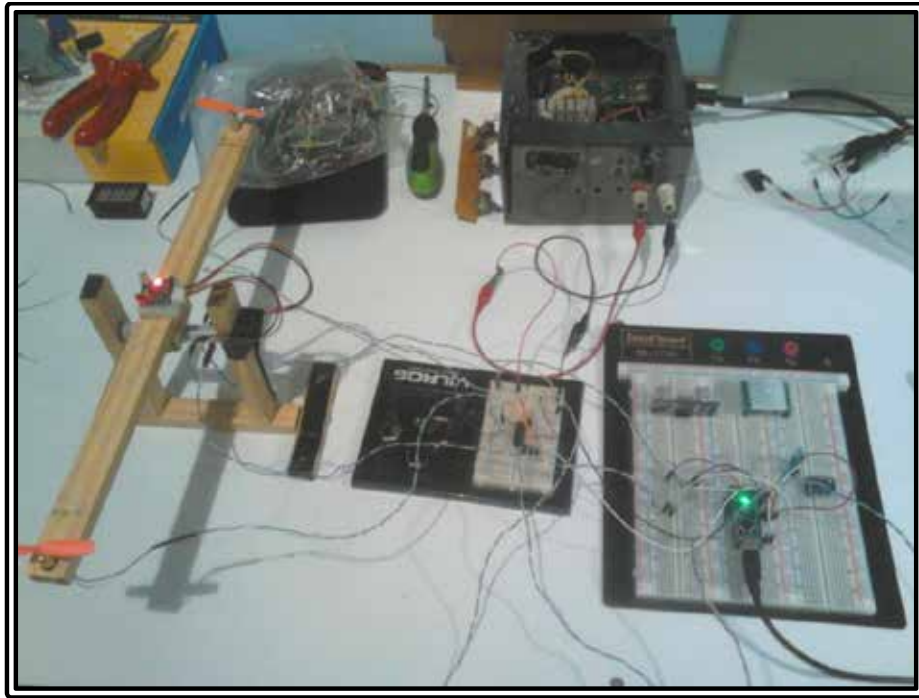


Figura 74: Prototipo diseñado para el primer ensayo
Fuente: Mario Pontillo (2018)

El prototipo de la figura 74 se diseñó de esa manera ya que lo que se quería probar en este ensayo era primero el control con un solo eje, ósea el control de un

solo grado de libertad. Se diseñó un pequeño prototipo cuya mecánica es parecida a la de un balancín o sube y baja. Como los ejes de navegación del Pitch y Roll son totalmente iguales pero ortogonales, con experimentar el control de uno solo fue suficiente y aplicar el mismo control para el otro eje o grado de libertad para poder seguir avanzando con la investigación. Este ensayo fue crucial también para el conocimiento del uso del IMU MPU6050 y el de también reconocer que el control PID es un control adecuado para este proyecto, teniendo así ideas más claras de cómo debía ser el diseño del mismo a futuro.

Para este ensayo se diseñó un control PID sencillo con el control de un solo eje para determinar el comportamiento de este. Los resultados de dicho ensayo fueron totalmente concluyentes y abrió paso para proceder con el siguiente diseño del prototipo pertinente que funcionaria para experimentar ahora con la unión de Pitch y Roll.

4.4.2 Ensayo con 3 Grados de Libertad

Teniendo en cuenta los datos adquiridos debido a la experimentación con un eje, como se mencionó anteriormente, en este ensayo se procedió a diseñar un prototipo y un algoritmo en el control PID para unir dichos ejes y además agregar el Yaw. El prototipo implementado se observa en la figura (ver figura 75 y figura 76).



Figura 75: Prototipo diseñado para el segundo ensayo
Fuente: Mario Pontillo (2018)



Figura 76: Prototipo diseñado para el segundo ensayo
Fuente: Mario Pontillo (2018)

El prototipo de la figura está construido con MDF y se hizo con la estructura y chasis igual a la de un Dron con topología en “+”, ya que se experimentaría con los 3 grados de libertad de este. Para lograr esto se acoplo este chasis en una rotula para que pudiera moverse libremente y dicha rotula se fijó en una base. Se implementaron los motores BLDC y los ESC, Ya que la estructura diseñada para este ensayo se hizo lo más parecida a la que se quería para el prototipo final, por ende se necesitó utilizar los motores BLDC del proyecto para poder moverla. Cabe destacar que esto también se hizo de esta manera para poder experimentar con la fuerza de los motores y familiarizarse con los ESC y poder probar algoritmos básicos para el uso de estos. Por último se procedió a experimentar y a analizar los resultados al igual que en el ensayo anterior y los resultados fueron suficientemente buenos y concluyentes para proceder a agregar el último grado de libertad al Dron. En este ensayo también se implementó el HMC5883L por primera vez, para la orientación del Yaw.

4.4.3 Ensayo con 4 Grados de Libertad

Para este ensayo el prototipo es básicamente el mismo que el del pasado, esto quiere decir que la estructura utilizada es la misma, a diferencia que este se desacoplo de la rótula y se le abrió un agujero en uno de sus lados y en vez de la rótula esta vez se le paso un bastón largo en el agujero y dicho bastón se fijó en la base. Esto se hizo con el propósito de poder agregar el último grado de libertad que el dron necesita.

El grado de libertad experimentado es el del Throttle, ósea el desplazamiento vertical, y es a lo que este ensayo fue dirigido, en experimentar con el Throttle mediante la implementación y uso del BMP180, ya que para este proyecto se deseaba un movimiento más autónomo del prototipo. En la siguiente figura se puede observar el montaje que se utilizó para este ensayo (ver figura 77).

Básicamente los algoritmos y componentes fueron los mismos que los del ensayo anterior, la única diferencia que en este caso se conoció, manipulo e implemento el BMP180, que es el sensor de altitud, e íntegro el código para la

manipulación de este en el algoritmo y para que así el control PID tomara la variable de la altitud en cuenta y así poder controlar más autónomamente la altitud también.



Figura 77: Prototipo diseñado para el tercer ensayo

Fuente: Mario Pontillo (2018)

Cabe destacar que para cada uno de los ensayos mencionado siempre se observaron y analizaron los datos que el sistema arrojaba durante y después de los experimentos. Todo eso se logró gracias a la comunicación Serial de Arduino, ya que en todo momento de las pruebas se dejó conectado el prototipo pertinente a una laptop para ir monitoreando dichos datos, analizarlos y obtener resultados concluyentes para seguir avanzando (Ver figura 78). Al igual que todos los ensayos anteriores, al obtener los datos deseados, se procedió a continuar avanzando en el trabajo de investigación. En este caso con los resultados suficientemente concluyentes se procedió directamente con el montaje final del proyecto, también debido a la falta de tiempo que se tuvo para este, no se pudo seguir haciendo más ensayos para asegurar el perfecto funcionamiento del prototipo.



Figura 78: Monitoreo de datos del tercer ensayo
Fuente: Mario Pontillo (2018)

4.4.4 Montaje final del trabajo de investigación

Se implementó el prototipo final del proyecto con todo el hardware y software diseñado en la fase II. Con todos los ensayos que se hicieron anteriormente mencionados satisfactorios, se procedió a construir un nuevo prototipo con una nueva estructura. Dicha estructura es una pre-fabricada, esta es más simétrica, ligera y de mejor calidad que las anteriores, aunque con la misma topología “+”. Este chasis viene completamente en partes, esto quiere decir desarmada, lo cual en esta fase se procedió al ensamble de este (ver figura 79 y figura 80) y la fabricación de los circuitos impresos con todo el hardware mencionado en la fase II de este trabajo de investigación.



Figura 79: Estructura ensamblada con ESCs Soldados y conectados para el nuevo prototipo.

Fuente: Mario Pontillo (2018)



Figura 80: Estructura ensamblada con ESCs Soldados y conectados para el nuevo prototipo.

Fuente: Mario Pontillo (2018)

Hasta ahora solo se había implementado la sensorica básica y necesaria para el funcionamiento de un Quadcopter, pero ya en esta fase se implementó el resto de esta, esto quiere decir la sensorica “extra” o los accesorios que proporcionan el objetivo específico del dispositivo, el cual es la vigilancia.

Se fabricó el circuito impreso principal en donde se encuentra y se conecta toda la electrónica del dispositivo. Dicho circuito impreso se diseñó en Proteus 8 y en la siguiente figura se puede apreciar el PCB (ver figura 81).

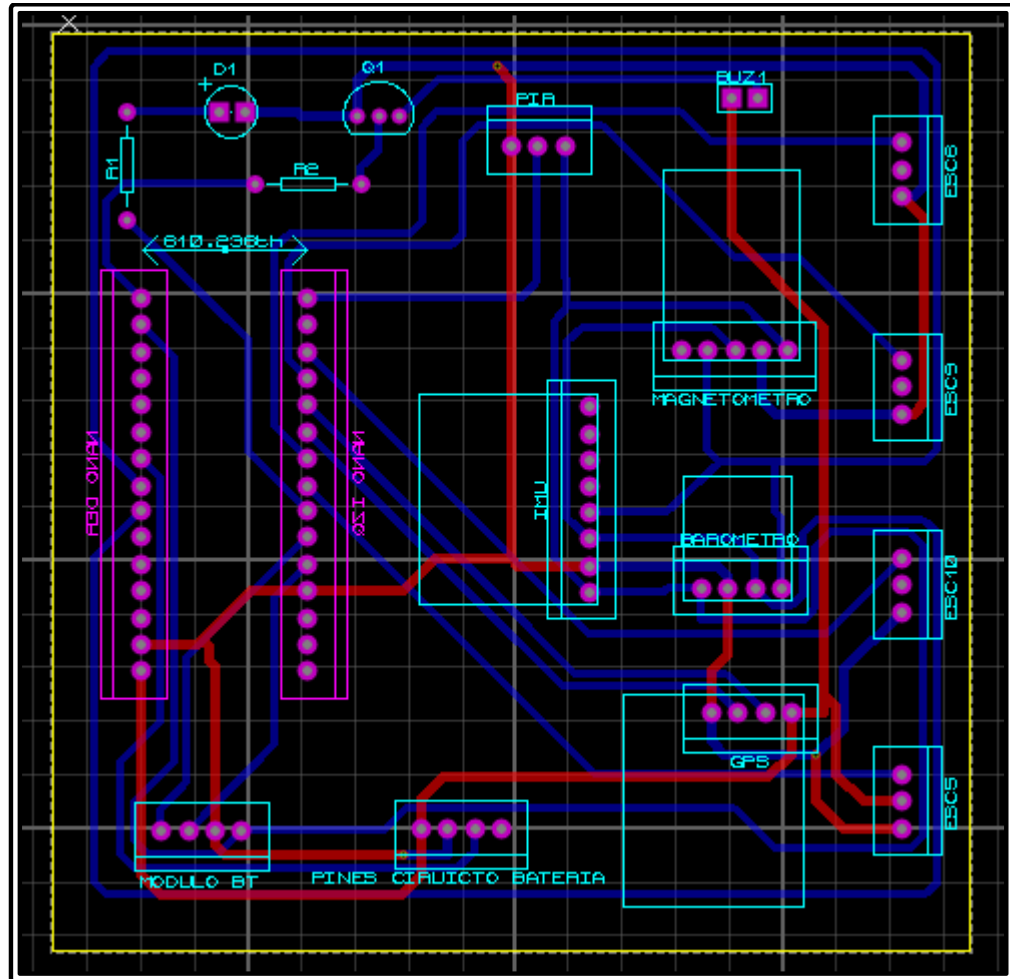


Figura 81: PCB Tarjeta principal del prototipo diseñado en Proteus 8.

Fuente: Mario Pontillo (2018)

El circuito medidor de batería se decidió diseñar en una tarjeta aparte. Se fabricó dicha tarjeta y se conectó la salida de este a la tarjeta principal por medio de cableado. El circuito impreso se diseñó también en Proteus 8 y se muestra el PCB en la figura (ver figura 82).

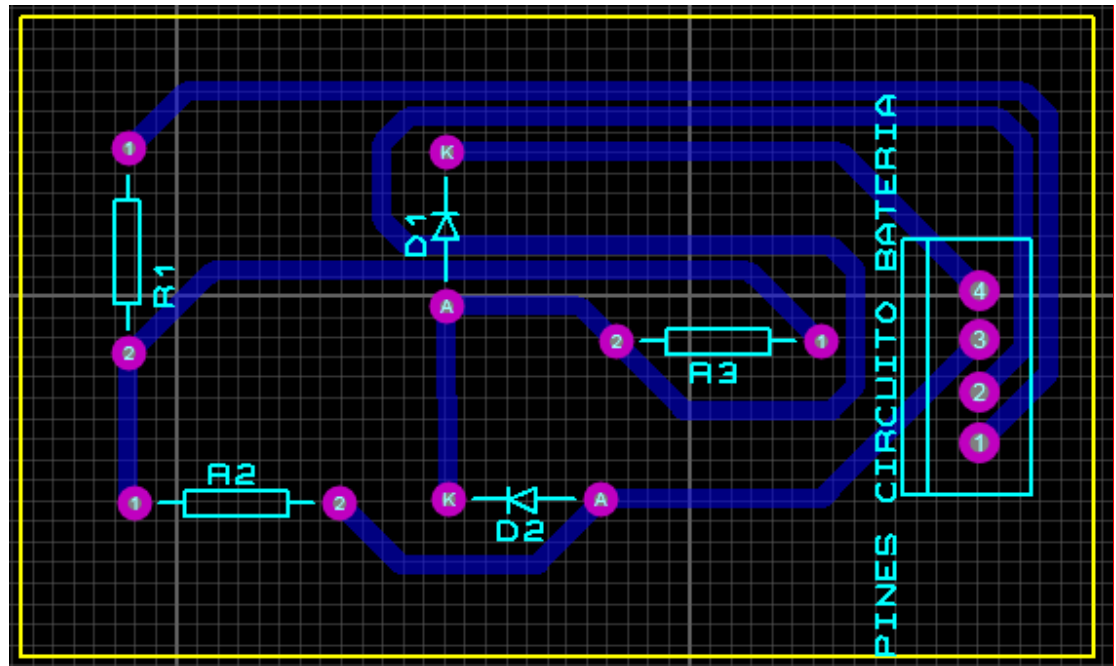


Figura 82: PCB Tarjeta circuito medidor de carga del prototipo diseñado en Proteus 8.

Fuente: Mario Pontillo (2018)

Por ultimo con los circuitos impresos listos, probados y la estructura completamente ensamblada con los motores y ESC instalados y soldados adecuadamente en ella, se realizó el montaje y conexión de las tarjetas principal y medidor de batería, para así concluir con la implementación del prototipo final del proyecto. En la figura se observa el dron inteligente diseñado (Ver figura 83).



Figura 83: Dron Completamente Armado
Fuente: Mario Pontillo (2018)



Figura 84: Dron Completamente Armado
Fuente: Mario Pontillo (2018)

CONCLUSIONES

Como conclusión final hay que destacar que el proyecto ha sido muy ambicioso y de una gran dificultad técnica, el objetivo principal era poder desarrollar un proyecto multidisciplinar donde intervinieran el diseño de hardware y software, como es el diseño y la construcción de un Quadcopter. También se encontró la existencia de la Asociación Venezolana de Robótica y Domótica, la cual está representada por la Universidades UCAB, USB, UNEXPO, UNET, ULA y UNEG.

El valor de lo aprendido en este proyecto es incalculable, tener la experiencia de crear un sistema desde cero. Ya que nos ha servido para poner en practica la capacidad para resolver casi cualquier tipo de problema. Siempre pensado la mejor solución en cuanto a calidad/precio y tiempo.

En cuanto al diseño de hardware ha sido un éxito, se consiguió integrar todos los componentes: Microcontrolador, Bluetooth, Sensores, Drivers y Motores. El diseño se realizó lo bastante ordenado para que en caso de avería de algunos de los componentes sea fácilmente reemplazable.

Para el software se logró diseñar una librería para cada dispositivo, otorgando alta independencia entre clases, capaz de tomar decisiones ante eventos específicos. En cuanto al control hay que destacar los parámetros encontrados son muy experimentales y varían en función del tipo de vuelo que se le quiera otorgar, y requieren de ciertos ajustes dependiendo del sitio donde se quiera volar. Lo ideal sería usar control adaptivo donde estos parámetros se ajustaran automáticamente, pero es un tema muy complejo y desde un principio se descartó por cuestión de tiempo. La comunicación con el móvil mediante el Bluetooth funciona perfectamente, y permite saber el estado de la batería, los cuatros grados de libertad del Quadcopter, la posición, entre otros.

Posteriormente para la interacción del prototipo con el usuario se diseñó una aplicación Android satisfactoriamente. Es importante destacar que la aplicación es

compatible con todos los dispositivos actuales como teléfonos inteligentes, tabletas, computadora, entre otros.

Las horas de ingeniería invertidas en el prototipo fueron de 20 a 25 horas semanales aproximadamente, para un total aproximado de entre 640 y 800 horas de trabajo invertidas. La rentabilidad de este prototipo en comparación con otros drones cuadricóptero comerciales, es mayor, esto se debe a que el prototipo fue diseñado con tecnología de software y hardware libre (open source), esto quiere decir principalmente que se encuentra mucha información al respecto y que el sistema puede ser optimizado y escalado por cualquier entusiasta. Otra ventaja es que este es más económico que cualquier otro dron comercial, por lo anteriormente mencionado ya que estos dispositivos son normalmente de bajo costo, pero con una tecnología igual de actualizada y de calidad.

RECOMENDACIONES

- Implementar un sistema de comunicación entre la aplicación y el Dron vía internet por medio de un módulo GSM.
- Lograr la implementación del mini ordenador Raspberry Pi para mayor procesamiento de datos y una cámara de alta definición con visión nocturna.
- Realizar una programación en dicho mini ordenador por medio de OPEN CV que logre la identificación y captura de rostro para la cámara mencionada.
- Diseñar un sistema de seguridad, capaz de enviar imágenes al usuario por medio de la aplicación Telegram cada vez que la cámara detecte un intruso en el área.
- Terminar el diseño de software que logra que el prototipo se mueva en una ruta definida por el usuario.
- Mejorar el diseño de la aplicación (Aspecto visual, seguridad, entre otros.)
- Desarrollar la aplicación de modo que sea compatible con diferentes sistemas operativos, por ejemplo IOS.
- Crear un control PID adaptativo, para que los parámetros se cambien automáticamente dependiendo del entorno y la estructura utilizada evitando la constante reprogramación del Arduino.
- Por último, se sugiere a la Universidad José Antonio Páez a afiliarse a la Asociación Venezolana de Robótica y Domótica.

REFERENCIAS BIBLIOGRÁFICAS

Honeywell, Datasheet **Magnetometro HMC 5883L**. (2016)

(Disponible en:

<http://dlnmh9ip6v2uc.cloudfront.net/datasheets/Sensors/Magneto/HMC5883L-FDS.pdf>).

B. Beauregard. (2016) **Arduino PID-Guía de uso librería PID**.

(Disponible

en:

<http://brettbeauregard.com/blog/wpcontent/uploads/2012/07/Gu%C3%ADa-de-uso-PID-para-Arduino.pdf>)

K. Shirriff y P. (2010) **Badger, Secrets of Arduino PWM**

(Disponible

en:

<http://arduino.cc/en/Tutorial/SecretsOfArduinoPWM>)

Oscar Torrente Artero. (2006) **Arduino curso básico de formación**. Alfaomega.

Hernández, R., Fernández, C. y Baptista P. (2006). **Metodología de Investigación**.

México: McGraw-Hill.

Ballestrini M. (2006). **Como elaborar el proyecto de investigación** Caracas:

Editorial Consultores Asociados.

Sergio Moyano Díaz (2014). **Diseño y construcción de un Quadcopter**.

Juan Carmona Fernández (2013). **Diseño de un sistema de control para un cuadricóptero**. Universidad Carlos III de Madrid.

Arias, F. (2004). **El proyecto de investigación: Introducción a la metodología científica**, Caracas: Editorial Episteme.

Hurtado, J. (2000). **Metodología de Investigación Holística (3ª. Ed.)**. Caracas: Editorial Sypal.

Tamayo y Tamayo, M. (2005). **El proceso de la investigación Científica**, México: Editorial Limusa.

Palella, S. y Martins, F. (2006). **Metodología de la investigación cuantitativa**. (2ª ed.), Caracas: Editorial Fedupel.

Asociación venezolana de robótica y domótica (2011). **http://www.ing.ula.ve/averod/?page_id=14**, Última visita: 20 de Marzo del 2018 a las 6:37 PM.

Real Academia Española (2018). **<http://dle.rae.es/?id=E7W0v9b>**, Última visita: 17 de Marzo del 2018 a las 8:00 PM.

Luis Llamas (2016). **<https://www.luisllamas.es/como-usar-un-acelerometro-arduino/>**, Última visita: 4 de Marzo del 2018 a las 10:30 PM.

Luis Llamas (2016). **<https://www.luisllamas.es/como-usar-un-giroscopio-arduino/>**, Última visita: 4 de Marzo del 2018 a las 10:30 PM.

MIT App Inventor (2018). <http://appinventor.mit.edu/explore/>, Última visita: 10 de Marzo del 2018 a las 6:30 PM.

Naylamp Mechatronics (2016).http://www.naylampmechatronics.com/blog/43_Tutorial-sensor-de-presi%C3%B3n-barom%C3%A9trica-BMP180.html, Última visita: 10 de Marzo del 2018 a las 7:30 PM.

ANEXO A
ENCUESTA

ITEM		I	O
1	¿Había oído hablar sobre los drones?		
2	¿Sabe lo que es un sistema de vigilancia?		
3	¿Cree usted importante el diseño e implementación de tecnología nacional para el desarrollo del país?		
4	¿Cree útil la implementación de un dron de vigilancia en los organismos de seguridad?		
5	¿Cree útil que un sistema de vigilancia avise a su móvil si detecto un posible intruso?		
6	¿Cree útil la existencia de Drones?		
7	¿Posee un teléfono inteligente o una tableta?		
8	¿Posee algún tipo de servicio de internet?		
9	¿Le gustaría poder supervisar el entorno donde reside desde un móvil?		
10	¿Ha pensado usted o algún conocido colocar un sistema de vigilancia en algún momento?		