



ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

UNIDAD DE GESTIÓN DE  TECNOLOGÍAS

DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA

**CARRERA DE ELECTRÓNICA MENCIÓN INSTRUMENTACIÓN
& AVIÓNICA**

**PROYECTO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL
TÍTULO DE TECNÓLOGO EN ELECTRÓNICA MENCIÓN
INSTRUMENTACIÓN & AVIÓNICA**

**TEMA: “IMPLEMENTACIÓN DE UNA INTERFAZ
TELECONTROL PARA UNA PLATAFORMA AÉREA MULTI-
ROTOR MEDIANTE MATLAB”**

AUTOR: IZA BARRIONUEVO CARLA BELEN

DIRECTOR: ING. CRISTIAN CHUCHICO

LATACUNGA

2018



**DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA
CARRERA DE ELECTRÓNICA MENCIÓN INSTRUMENTACIÓN &
AVIÓNICA**

CERTIFICACIÓN

Certifico que el trabajo de titulación **“IMPLEMENTACIÓN DE UNA INTERFAZ TELECONTROL PARA UNA PLATAFORMA AÉREA MULTI-ROTOR MEDIANTE MALTAB”** realizado por el señor **IZA BARRIONUEVO CARLA BELEN**, ha sido revisado en su totalidad y analizado por el software anti-plagio, el mismo cumple con los requisitos teóricos, científicos, técnicos, metodológicos y legales establecidos por la Universidad de Fuerzas Armadas ESPE, por lo tanto me permito acreditarlo y autorizar al señor **IZA BARRIONUEVO CARLA BELEN** para que lo sustente públicamente.

Latacunga, 11 Julio del 2018

**ING. CRISTIAN CHUCHICO
DIRECTOR**



**DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA
CARRERA DE ELECTRÓNICA MENCIÓN INSTRUMENTACIÓN &
AVIÓNICA**

AUTORÍA DE RESPONSABILIDAD

Yo, **IZA BARRIONUEVO CARLA BELEN** con cédula de identidad N° 1724128572 declaro que este trabajo de titulación “**IMPLEMENTACIÓN DE UNA INTERFAZ TELECONTROL PARA UNA PLATAFORMA AÉREA MULTI-ROTOR MEDIANTE MATLAB**” ha sido desarrollado considerando los métodos de investigación existentes, así como también se ha respetado los derechos intelectuales de terceros considerándose en las citas bibliográficas.

Consecuentemente declaro que este trabajo es de mi autoría, en virtud de ello me declaro responsable del contenido, veracidad y alcance de la investigación mencionada.

Latacunga, 11 Julio del 2018

IZA BARRIONUEVO CARLA BELEN

C.C: 1724128572



**DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA
CARRERA DE ELECTRÓNICA MENCIÓN INSTRUMENTACIÓN &
AVIÓNICA**

AUTORIZACIÓN

Yo, **IZA BARRIONUEVO CARLA BELEN**, autorizo a la Universidad de las Fuerzas Armadas ESPE publicar en la biblioteca Virtual de la institución el presente trabajo de titulación **“IMPLEMENTACIÓN DE UNA INTERFAZ TELECONTROL PARA UNA PLATAFORMA AÉREA MULTI-ROTOR MEDIANTE MATLAB”** cuyo contenido, ideas y criterios son de mi autoría y responsabilidad.

Latacunga, 11 Julio del 2018

IZA BARRIONUEVO CARLA BELEN

C.C: 1724128572

DEDICATORIA

Este trabajo y esfuerzo va dedicado a Dios quien guío mi camino día a día para poder ir a la Universidad y regresar bien a casa. A mis padres Carlos & María del Carmen quienes son mi fuente de inspiración y motivación para poder superarme cada día mas y así poder luchar para un futuro mejor.

A mis hermanas Evelyn & Valeria que han estado y estarán conmigo en las buenas y malas, brindándome su apoyo y cariño desinteresadamente. A Dayan quien con sus palabras de aliento no me dejaban decaer para no fallarles a mis padres y a mí misma.

CARLA IZA

AGRADECIMIENTO

Agradezco a Dios y a mis padres Carlos & María de Carmen por ser el pilar fundamental de toda mi vida quienes siempre han estado a mi lado apoyándome en los momentos más difíciles alentándome a que nunca me rinda, dándome sus consejos para que siga adelante, brindando su confianza y amor.

A mis ingenieros que con sus enseñanzas me permitieron llegar hasta estas estancias, en particular al Ing. Cristian Chuchico & Ing. Pablo Pilatasig y a todos los ingenieros que me han orientado de manera significativa con sus conocimientos en la elaboración de tesis.

CARLA IZA

ÍNDICE DE CONTENIDO

PORTADA	i
CERTIFICACIÓN.....	ii
AUTORÍA DE RESPONSABILIDAD	iii
AUTORIZACIÓN	iv
DEDICATORIA.....	v
AGRADECIMIENTO.....	vi
ÍNDICE DE CONTENIDO	vii
ÍNDICE DE TABLAS	ix
ÍNDICE DE FIGURAS.....	ix
RESUMEN.....	xi
ABSTRACT	xii
CAPÍTULO I.....	1
1.1 Antecedente	1
1.2 Planteamiento del problema	2
1.3 Justificación	3
1.4 Objetivos.....	3
1.4.1 Objetivo General	3
1.4.2 Objetivos Específicos	3
1.5 Alcance.....	4
CAPÍTULO II.....	5
2.1 Software Matlab.....	5
2.1.1 Características	5
2.2 Matlab GUI	6
2.3 Naza M-LITE	8
2.3.1 Características Naza M-Lite	8

2.3.2 Comportamiento Naza M-Lite.....	9
2.4 Componentes básicos de un cuadricóptero.....	10
2.4.1 Estructura o frame	10
2.4.2 BATERÍA LIPO	11
2.5 Motores brushless RCX 2219 - 920KV	13
2.6 HELICES	15
2.7 ESC (ELECTRONIC SPEED CONTROLLER)	17
2.8 SOFTWARE ARDUINO	18
2.8.1 HARDWARE ARDUINO	18
2.8.2 ARDUINO	18
2.8.3 ARDUINO UNO	19
2.9 APLICACIONES DE LOS DRONES EN AMBITO CIVIL	19
CAPÍTULO III.....	20
3.1 PRELIMINARES	20
3.2 ENSAMBLE DEL CUADRICOPTERO.....	20
3.3 CONFIGURACIÓN DEL SOFTWARE NAZA M LITE	24
3.4 DESARROLLO DE LA INTERFAZ	30
3.4.1 INTERFAZ EN EL SOFTWARE MATLAB GUI.....	30
3.4.2 DESCRIPCION DE LOS ELEMENTOS UTILIZADOS EN LA PROGRAMACION	32
3.5 COMUNICACIÓN INALAMBRICA	37
3.5.1 CONFIGURACION DE LA ANTENA 3DR	38
3.6 PROGRAMACION EN ARDUINO	40
3.6.1. RECEPCION DE DATOS.....	40
3.7 ANALISIS DE RESULTADO.....	44

CAPITULO IV	46
4.1 Conclusiones	46
4.2 Recomendaciones	46
GLOSARIO DE TERMINOS	47
REFERENCIAS BIBLIOGRÁFICAS	48

ÍNDICE DE TABLAS

Tabla 1 Características de Batería Li-Po.....	13
Tabla 2 Características motor brushless	15
Tabla 3 Características Hélices	17

ÍNDICE DE FIGURAS

Figura 1 Interfaz de Matlab.....	6
Figura 2 Interfaz de GUI.....	7
Figura 3 Formas de arranque del motor	10
Figura 4 Estructura Drone F450.....	10
Figura 5 Batería Li-Po	11
Figura 6 Cargador de Batería Li-Po.....	12
Figura 7 Motor Brushless.....	14
Figura 8 Colocación de las hélices	16
Figura 9 Controlador Electrónico de Velocidad	17
Figura 10 Acoplamiento de los motores brushless con la estructura .	21
Figura 11 Protección con termo-retráctil.....	21
Figura 12 Diagrama de conexión.....	21
Figura 13 Conexión del Control principal (MC) con los controladores (ESC).....	22
Figura 14 Indicaciones para configuración del GPS	23
Figura 15 Joystick (Extreme 3D Pro).....	23
Figura 16 Icono del software DJI NAZAM	24
Figura 17 Interfaz principal del software NAZA-M V2	24
Figura 18 Reconocimiento del hardware en el software NAZA-M V2... 	25

Figura 19 Verificación de los movimientos de los canales	25
Figura 20 Selección del Quad-rotor Tipo X.....	26
Figura 21 Valores de la ubicación exacta del GPS	26
Figura 22 Selección de Tipo de Receptor y Modo de Vuelo.....	27
Figura 23 Valores de ganancia del UAV.....	28
Figura 24 Selección de velocidad de arranque de los motores.....	28
Figura 25 Valores para protección de descarga de LiPo 3S	29
Figura 26 Exportación de parámetros hacia el controlador DJI NAZA	29
Figura 27 Interfaz Grafica	30
Figura 28 Archivos M-file (.m) y Fig-file (.fig)	31
Figura 29 Propiedades de cada elemento.....	31
Figura 30 Interfaz Principal del Telecontrol.....	32
Figura 31 Elementos “AXES” de la interfaz de Navegación.....	32
Figura 32 Botones de indicación.....	33
Figura 33 Elemento Pop-Up Menu	33
Figura 34 Slider de control.....	34
Figura 35 Inicio de la programación.....	35
Figura 36 Programación de la interfaz GUI.....	35
Figura 37 Programación de configuración del puerto USB	36
Figura 38 Programación de control de los slider.....	37
Figura 39 Identificación de cada elemento de la interfaz	38
Figura 40 Alimentación de receptor y el transmisor.....	38
Figura 41 Parámetro de Tx y Rx	39
Figura 42 Verificación de Comunicación entre Tx y Rx	39
Figura 43 Pines que generan PWM en Arduino Uno.....	40
Figura 44 Establece la velocidad de datos en bits por segundo (baudios) para la transmisión de datos en serie	40
Figura 45 “serialEvent” permite recibir datos seriales de una forma parecida a una interrupción o de forma asíncrona.....	41
Figura 46 Desbloqueo de motores	42
Figura 47 Compilación y verificación de error en el sketch.....	43
Figura 48 Drone ensamblado	44
Figura 49 Drone en vuelo	44
Figura 50 Elementos para realizar el control.....	45

RESUMEN

El proyecto técnico hace referencia a la implementación de una interfaz telecontrol para una plataforma aérea multi-rotor mediante MATLAB que por medio de un controlador Naza M-Lite permite la estabilización del vehículo aéreo no tripulado (UAV).

Mediante el software Matlab GUI se realizó la programación para generar una trama datos generados por un joystick EXTREME 3D PRO, estos se transmiten mediante una antena transmisora 3DR de 915 Mhz a una antena receptora que cuentan con un protocolo MAVLink siendo un sistema utilizado para aeronaves no tripuladas con un estándar IEEE802.11, la misma que está conectada a la comunicación serial de Arduino el cual es encargado de generar los PWMs necesarios para el control del UAV.

Las PWMs controlan la velocidad de los motores activándolos o desactivándolos para su funcionamiento, estos están configurados para diferentes movimientos tales como son: Ailerons, Elevator, Rudder, Throttle que ayudan al despegue y al movimiento del multi-rotor.

Terminado este proceso se verifica el correcto funcionamiento de la implementación teniendo un resultado satisfactorio en condiciones de vuelo favorables en las pruebas prácticas, sin embargo, al exponer al vehículo aéreo no tripulado a perturbaciones generadas por el viento se perderá comunicación.

Palabras claves:

Matlab

UAV

PWMs

Multi-rotor

ABSTRACT

This technical project refers to the implementation of a telecontrol interface for a multi-rotor aerial platform using MATLAB, which means of a Naza M-Lite controller wich allows the stabilization of the unmanned aerial vehicle (UAV). Through Matlab GUI software, the programming was performed to generate a frame to do data generated by an EXTREME 3D PRO joystick, which is transmitted through a 3DR transmitting antenna of 915 Mhz to a receiving antenna that has a MAVLink protocol and is a system used for unmanned aircraft with an IEEE802.11 standard, which is connected to Arduino's serial communication which is responsible for generating the PWMs needed to control the UAV.

The PWMs control the speed of the motors by activating or deactivating them for operation, they are configured for different movements such as these: Ailerons, Elevator, Rudder, Throttle that help them launch and the movement of the multi-rotor.

Once this process is finished, the correct functioning of the implementation is verified, with a satisfactory result under favorable flight conditions in the practical tests, however, when the unmanned aerial vehicle is exposed to disturbances generated by the wind, the communication will be lost.

KEYWORDS:

MATLAB

UAV

PWMS

MULTI-ROTOR

CHECKED BY:

Lcda. María Elisa Coque

DOCENTE UGT

CAPÍTULO I

1.1 Antecedente

El interés por la seguridad ciudadana se ha incrementado en los últimos años, muestra de ello son (Roman Nuñez & Ruiz Veintimilla, 2012) quienes en su trabajo de investigación de la Universidad Central Del Ecuador, realizan un análisis de la prevalencia de delitos corresponde al porcentaje o proporción de la población que ha sido víctima de la inseguridad, contados y registrados una sola vez independientemente del número de veces que hayan sido víctimas de este problema. Se tiene el Índice de prevalencia de robo a personas con uso de la fuerza, sin uso de la fuerza y amenazas; en este sentido tomando en cuenta los resultados de la encuesta, tenemos que en la provincia de Pichincha existe un 7,8% de casos, en los que personas de 16 años y más, han sido víctimas de las diferentes formas del delito.

El interés por la utilización de Drones hace referencia a las investigaciones realizadas por diferentes autores, uno de los trabajos es el de (Barrientos , y otros, 2007) quienes desarrollaron el informe “Vehículos aéreos no tripulados para uso civil. Tecnología y aplicaciones”, donde se da a conocer datos investigativos acerca del avance de esta aeronave no tripulada, utilidades de estos equipos tecnológicos entre otras novedades que han servido de gran ayuda en las tareas agrarias.

En la Unidad De Gestión De Tecnologías “ESPE” no se ha realizado trabajos de titulación basados en la seguridad ciudadana, siendo este trabajo un proyecto innovador, en el cual se utilizará tecnología sofisticada para el control de una plataforma multi-rotor que servirá como ayuda para el monitoreo de varios escenarios mediante imágenes aéreas tomadas desde lugares donde el ojo humano no pueda visualizar.

1.2 Planteamiento del problema

Existe gran diversidad de situaciones que afectan la seguridad ciudadana. Sin duda el aumento de la violencia y de la delincuencia en todas sus expresiones es el principal elemento catalizador de la sensación de inseguridad que viven el mundo entero. No obstante, la mayor dificultad encontrada para una evaluación del fenómeno reside en la forma de conceptualizarlo y medirlo puesto que está configurado y se manifiesta de manera multidimensional asociado estrechamente a la subjetividad de las personas. (Arriagada & Godoy)

En los últimos años el país de Ecuador ha transformado sus ciudades en grandes centros urbanos. El tema de la inseguridad y la violencia se encuentra presente en la vida social, se intensifica en zonas donde los ciudadanos no velan por su integridad moral y física. El aumento de la inseguridad se ha producido principalmente por el crecimiento de delitos comunes como robos atracos, pero que derivan de otras cuestiones sociales afectando los espacios de convivencia ciudadana. En la actualidad estos espacios son tratados, desde el enfoque de la seguridad ciudadana, a partir de los modelos de prevención del delito que busca generar un proceso de organización en las zonas afectadas por la inseguridad, (Fernanda, 2017) registra evidencia que para el año 2011 los centros de las ciudades son la segunda zona que presenta mayor delincuencia a personas, con un porcentaje de 16,6% por el lugar donde vive, mientras que por lugar de concurrencia ocupa el mismo sitio con un 22,9%, y el robo a personas con 8,5% según la Seguridad Ciudadana.

En el último año, en la Unidad de Gestión de Tecnologías se han registrado alteraciones en la seguridad que ha afectado a docentes y alumnos de la Unidad, el principal problema presente en las inmediaciones de la universidad radica en permitir ingresar a personas ajenas a la institución pasándose como alumnos causando así la pérdida de objetos propios de los profesores y de los estudiantes en general. El descuido de los docentes y estudiantes al tener que cubrir otras ocupaciones hace mucho más fácil el trabajo para las personas dueñas de lo ajeno.

1.3 Justificación

Los problemas de inseguridad se han convertido en el centro de las preocupaciones públicas, afectando la integridad de las personas por lo que es de suma importancia el estudio e investigación para tratar y garantizar la seguridad ciudadana con la mayor eficacia.

Varios sistemas de seguridad ciudadana se han implementado en distintas zonas en un entorno donde se fomente una cultura de paz y de seguridad integral, se ha debido fundamentalmente a aspectos de orden estructural en el Ministerio del Interior que, con su ayuda, han designado policías que trabajan con estrategias necesarias para combatir la inseguridad ciudadana.

El presente proyecto tiene como finalidad brindar ayuda a la seguridad de la Unidad de Gestión de Tecnologías que ha presentado delincuencia en sus instalaciones implementando una interfaz telecontrol para una plataforma aérea multi-rotor con el fin de disminuir la inseguridad ciudadana.

1.4 Objetivos

1.4.1 Objetivo General

“Implementar una interfaz telecontrol para una plataforma aérea multi-rotor mediante MATLAB”

1.4.2 Objetivos Específicos

- Desarrollar el algoritmo de programación en el software MATLAB para la interfaz de telecontrol de la plataforma multi-rotor.
- Implementar un sistema que permita establecer la comunicación entre la plataforma multi-rotor y la interfaz en Matlab.
- Realizar pruebas de funcionamiento del sistema tanto del software como del hardware que controla la plataforma multi-rotor.

1.5 Alcance

La implementación de una interfaz de telecontrol para el control de una plataforma multi-rotor ayudará a reducir la inseguridad en todas las instalaciones de la Unidad de Gestión de Tecnologías permitiendo así llevar un registro para formular antecedentes de personas que cometan actos que atenten con la integridad de las personas y sobre todo también para disminuir los actos delictivos especialmente de robo dentro de la universidad.

CAPÍTULO II

MARCO TEORICO

Software de Procesamiento

2.1 Software Matlab

Matlab viene de la palabra **Matrix Laboratory** que significa laboratorio de matrices el cual presenta diversas aplicaciones como manipulación de matrices, presentación de datos y funciones, implementación de algoritmos, comunicación con programas de otros lenguajes y dispositivos hardware y la creación de interfaces de usuario (GUI).

Matlab es un software que al paso de los años se ha destacado en el campo investigación en universidades y centros de estudio que realizan diversas aplicaciones en búsqueda de la solución a problemas o facilitar en la ejecución de tareas.

2.1.1 Características

Las características más destacadas de Matlab son las siguientes:

- Lenguaje permite operaciones de vectores, matrices, funciones y programación orientada a objetos que puede ejecutarse en un entorno interactivo o archivos script.
- Gráficos: suministra funciones y herramientas para la visualización de datos en 2D Y 3D y herramientas para crear diagramas personalizados, también dispone de recursos para mostrar vectores y matrices en gráficos e imprimirlos.

En la construcción de gráficos se realiza una interfaz del programa que permite colocar los elementos donde se desee. Tras ello se exporta la información a un fichero.m.

- La función básica de creación de gráficos es plot
- Si y es un vector, plot (y) dibuja un gráfico de los valores de los elementos de y frente a sus índices.

- Si x e y son dos vectores de igual tamaño, $\text{plot}(x,y)$ dibuja un gráfico de los valores de los elementos de y frente a los de x .
- Programación de funciones: Cada elemento del gráfico tiene asociado en el fichero `.m` una función donde el programador escribe las instrucciones de cada elemento.
- Interacción con otros lenguajes de programación: Matlab puede realizar interfaces al llamar funciones o subrutinas escritas en lenguaje C o Fortran al crear una función envoltorio que permite que sean pasados y devueltos datos de Matlab.

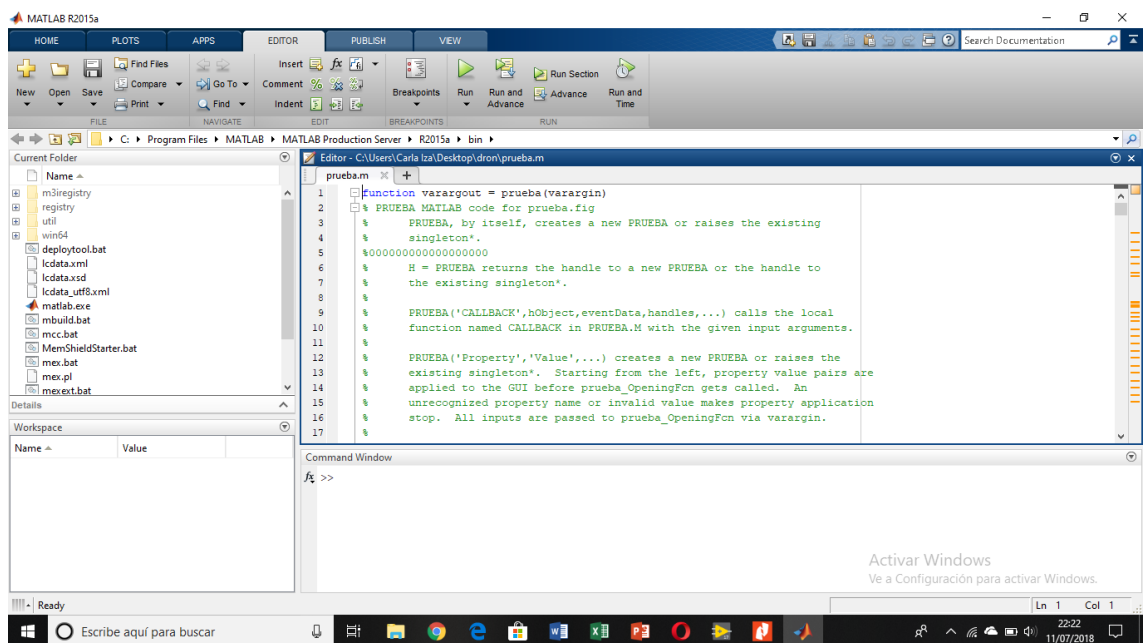


Figura 1 Interfaz de Matlab

2.2 Matlab GUI

Matlab cuenta con un editor de diseño GUIDE que permite diseñar gráficamente la interfaz de usuario al mismo tiempo puede agregar cuadros de dialogo y controles (botones y controles deslizantes).

Entorno gráfico: El comando GUIDE abre una pantalla con la que se puede dibujar el esquema gráfico de la interfaz.

- Cada elemento añadido es un objeto con un nombre y propiedades que se pueden modificar en la pantalla.
- Los elementos se pueden alinear, formar bloques y otra serie de operaciones para conseguir una gráfica bonita.
- Cuando la figura se haya terminado se procede a exportar la información a un fichero.m.

Los componentes habituales de un GUI son los siguientes:

- Sliders es un recurso que se utiliza para presentar y cambiar información de manera gráfica y dinámica.
- Pushbuttons es un botón con un mecanismo de conmutación simple para controlar aspectos en un proceso.
- List boxes es un elemento de control gráfico que permite al usuario seleccionar una o más elementos de un cuadro de lista dentro de un cuadro de texto.
- Frames es un elemento que permite dividir una pantalla en varias áreas independientes unas de otras, y con contenidos distintos, aunque puedan estar relacionados.

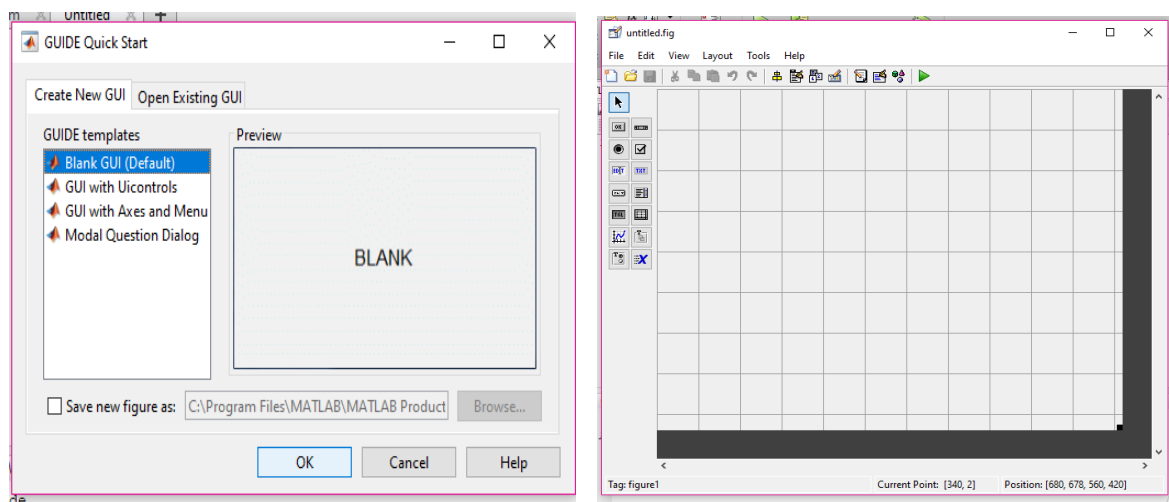


Figura 2 Interfaz de GUI

2.3 Naza M-LITE

El kit NAZA M-LITE es un sistema autopiloto, ofreciendo características de vuelo para el trabajo de multi-rotors que alcanza bajas alturas.

Este controlador de estabilización de vuelo provee un sistema de auto nivelado y una excelente estabilización de altura, proporcionando aplicaciones profesionales o de entrenamiento, también ofrece la seguridad del operador en el proceso de calibración gracias a la disposición de la deshabilitación de los motores mientras existe la conexión con la PC y su protección de la descarga de las baterías.

2.3.1 Características Naza M-Lite

- Diseño todo en uno: Integra dentro de la misma caja, amortiguación de aire, PC, giros, acelerómetro y barómetro.
- Nuevo algoritmo avanzado de actitud y estabilización.
- BEC (Battery Eliminator Circuit) y LED (Light Emitting Diode) todo en uno.
- Protección de batería baja: Mediante LED alta luminosidad y aterrizaje automático.
- D-BUS: Soporta receptores con salida D-Bus/S-Bus.
- PPM (Pulse Position Modulation): Soporta receptores con salida PPM.
- Estabilización cámara 2 ejes: El sistema de estabilización NAZA es compatible con la mayoría de Gimbal (articulación empleada en mecánica para transmitir un movimiento de rotación en distintas direcciones) del mercado, ofreciendo una estabilización perfecta.
- Firmware y software específico Naza Lite
- IOC (Intelligent Orientation Control)
- Aviso batería baja.

2.3.2 Comportamiento Naza M-Lite

- **Algoritmo de estabilidad avanzado**

El algoritmo de estabilidad avanzado provee estabilidad en vuelo de los productos DJI, también una excelente maniobrabilidad. Además, los operadores del multirrotor tiene una experiencia de vuelo excelente.

- **Múltiples modos de control de vuelo**

El NAZA M-LITE ofrece tres tipos de control de vuelo:

- Modo de actitud GPS
- Modo de actitud
- Modo Manual

Estos modos de control de vuelo ofrecen al piloto tener distintas características de vuelo en los cuales implica su ambiente asegurando un vuelo seguro y sano.

- **Modulo GPS (Global Positioning System)**

El módulo GPS plug and play mejora la posición ya que cuenta con retención de posición, retorno al lugar de despegue y funciones de control de orientación, también su altitud con exactitud que ayuda en condiciones de viento.

- **Modo Failsafe**

Si el transmisor tiene una falla se puede activar el modo de falla a través del puerto-U. El controlador de vuelo tiene incorporado una función de falla automático, permitiendo las salidas de todos los comandos de control ubicándose la posición central cuando se pierde la comunicación entre el controlador y el transmisor protegiendo la integridad del multirrotor hasta reconectar la comunicación con el radio control.

- **Armado y desarmado de motores**

Al momento de armar los motores para su arranque tienen cuatro formas.

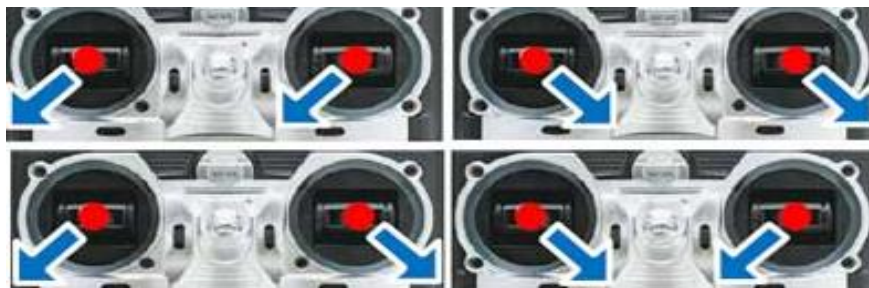


Figura 3 Formas de arranque del motor

Fuente: (Gizmojo, 2014-2015)

Hay dos modos de apagar los motores:

- Modo Inteligente
- Modo Inmediato

2.4 Componentes básicos de un cuadricóptero

2.4.1 Estructura o frame

Corresponde al cuerpo del cuadricóptero F450, cuya función es servir de esqueleto para componentes eléctricos y mecánicos del Drone. Su elaboración se basa de un material denominado PA66+30GF, constituido por Poliamida 66 (nylon 66) y relleno en un 30% de refuerzo de fibra de vidrio para estos componentes dada su alta resistencia y bajo peso.

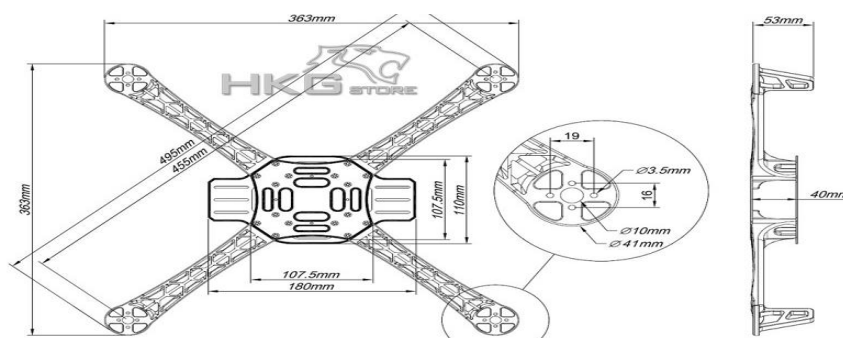


Figura 4 Estructura Drone F450

Fuente: (Modeltronic, 2015)

2.4.2 BATERÍA LIPO

Las baterías de polímero de iones de Litio, más conocidas por la abreviación Li-Po, siendo una batería recargable, compuesta generalmente de varias células secundarias idénticas en paralelo para aumentar la capacidad de la corriente de descarga.



Figura 5 Batería Li-Po

A. Celdas

Las baterías Li-Po están formadas por celdas, depende de cómo se conecte las celdas que componen la batería se obtendrá más o menos nivel de tensión, estas celdas pueden estar conectadas de dos formas en serie o paralelo.

- **Conexión en serie**

La conexión en serie se conecta un N de celdas, que por todas ellas circulan la misma intensidad, que es igual a multiplicar por N el valor de la tensión de una celda.

- 1S= 1 celda = 3,7 voltios
- 2S= 2 celda = 7,4 voltios
- 3S= 3 celda = 11,1 voltios

- **Conexión en paralelo**

La conexión en paralelo de varias celdas, cada celda conectada sus extremos eléctricamente a un mismo punto eléctrico, quiere decir todos los extremos positivos de las celdas son como un mismo punto eléctrico y los negativos también, mientras que la tensión entre ambos extremos del conjunto de celdas en paralelo es una sola celda.

Con esto se logra duplicar su capacidad de corriente obteniendo algo más importante que es duplicar el amperaje que podemos obtener del conjunto.



Figura 6 Cargador de Batería Li-Po

Fuente: (Turnigy, s.f.)

B. Especificaciones

Las baterías Li-Po tienen una vida útil de 2 a 3 años o 500 cargas completas, son fabricadas de diferentes formas y tamaños, su tasa de descarga va desde 1 A hasta 25 A. En las baterías Li-Po cada celda tiene 3.7 V y puede tener baterías de 1 a 6 celdas.

Tabla 1**Características de Batería Li-Po**

Características	Dimensiones
Química	Li-Polymer
Número de células	3S
Voltaje	Voltaje:
Capacidad	Capacidad:
Descarga	55C
Enchufe de carga	JST-XH
Tapón de descarga	Conector / conector T Dean-Style
Dimensión de la batería	133 * 43 * 30 mm
Peso de la batería	460g

Fuente: (Fluoreon, 2017)

C. Cargador de batería Li-Po

Los cargadores de las baterías Li-Po tienen entradas donde se inserta los conectores que se utiliza para conseguir que todos los elementos que forman el pack o la batería carguen por igual y consigan tener el mismo amperaje y sobre todo voltaje, teniendo un correcto cargado de la batería. (Turbo Hobby , 2013)

2.5 Motores brushless RCX 2219 - 920KV

El motor brushless es un motor eléctrico que no emplea escobilla, las cuales son elementos que hacen contacto en el colector de un motor común, los bobinados de cobre (Cu) se encuentra en la parte interior central y los imanes están situados en una campana exterior que rodea al bobinado y a la conecta el eje al tiempo que gira la campana gira el eje produciendo un menor número de revoluciones haciendo que la campana actúe como un volante de inercia que dota de un mayor par generando más potencia.

Los primeros motores sin escobillas (Brushless) fueron motores de corriente alterna asíncronos que tenían un colector de delgas o un par de anillos rozantes provocando un rozamiento que disminuía su rendimiento.



Figura 7 Motor Brushless

A. Especificaciones

Los motores sin escobillas, de mejor rendimiento y escaso o nulo mantenimiento. Basados en los fundamentos de los de corriente alterna donde un sólo campo magnético provoca el giro, por tanto, los variadores tienen que ser distintos a los habituales ya que tienen que simular dicha corriente alterna.

Los motores Brushless son muy utilizados para el aeromodelismo ya que presentan ventajas importantes: piezas sin rozamientos, sin escobillas que no genera chispas perturbadoras, volumen y peso menor con mayor aprovechamiento de la energía (Aeromodelismo, 2016).

Tabla 2

Características motor brushless

Características	Dimensiones
KV	920
Configuración	12N14P
Diámetro del estator	22 mm
Longitud del estator	22 mm
Diámetro del eje	3.0 mm
Dimensiones del motor (Dia. * Len)	Φ28 × 24mm
Peso (g)	28g
Peso	aprox. 54 g (tuerca de puntas incluida)
No. of Cells (Lipo)	3-5S
Máx. Corriente continua (A)	17.2 ^a
Max Fuente Continua (W)	318W
Resistencia interna	Ω

Fuente: (Myrcmart, 2008-2017)

B. FACTOR KV

El factor “kV” es un parámetro muy importante que se debe considerar para que nos indique el número de revoluciones por minuto a la que es capaz de girar el motor por cada voltio de electricidad a la que se aplique.

2.6 HELICES

Las hélices se fabrican a partir de materiales compuestos de polímeros, estos tienen la característica anisotropías debido a que resisten fuerzas muy altas. Esto es útil para los multirotores debido al tipo de fuerzas centrífugas y de roce viscoso con el aire.

Las hélices transforman la energía mecánica en fuerza propulsora, están constituidas por un número variable de aspas o palas, en la que cada pala está formada por un conjunto de perfiles aerodinámicos.

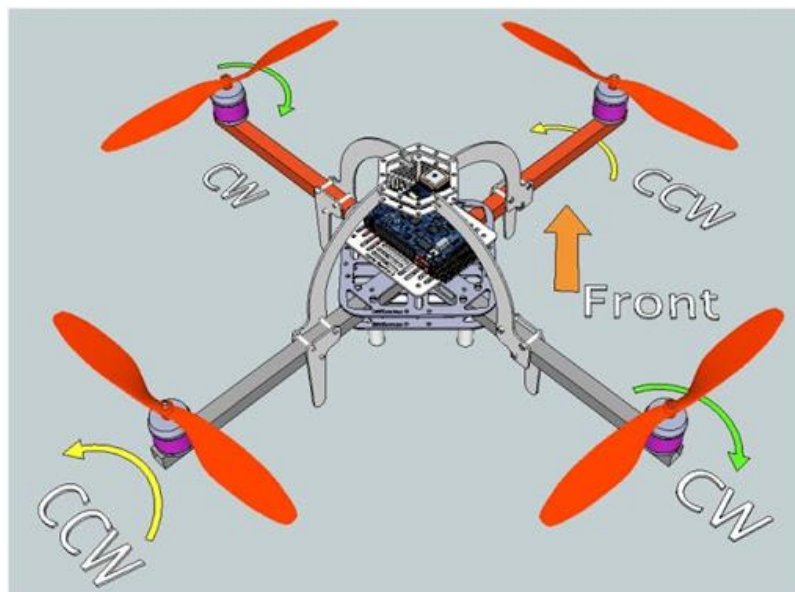


Figura 8 Colocación de las hélices

Fuente: (Dronecenter, 2016)

A. VELOCIDAD DE GIRO Y ACELERACION

Las hélices al tener mayor ángulo de ataque tienen mayor empuje en términos generales. Esto es así asumiendo una velocidad de giro del motor constante. Un mayor empuje siempre resulta en un mayor consumo de amperios. Este factor se debe tener en cuenta para el ensamble del vehículo aéreo no tripulado, teniendo como objetivo importante. (Guiadrone, 2017).

El tamaño de la pala afectará la aceleración del dron. Las palas de tamaño pequeño aceleran más rápido que las grandes. Las palas grandes tienen la ventaja de conseguir una mayor sustentación en el aire. Su lado negativo es que tardan más en acelerar y conseguir las RPM deseadas.

B. FORMAS

Existen diferentes tipos de hélices:

- Hélices bipala que serán las que usaremos al principio para aprender a volar.
- Hélices tripalas que son las que usan normalmente los expertos. Estas hélices ofrecen mayor empuje y “flotabilidad”.

Tabla 3

Características Hélices

Características	Dimensiones
Diámetro de la hélice	10 "/ 254 mm
Paso propulsor	4.5 "/ 114.3 mm
Diámetro del eje	8 mm (ajustable a través de anillos adaptadores *)
Espesor del cubo	8 mm (ranura de tuerca hexagonal de 2,2 mm)
Peso de la hélice	0.44oz / 12.6g
peso del paquete	3.7oz / 105g
Dirección de rotación	2 CW y 2 CCW en el conjunto
Material	Compuesto de fibra de vidrio R
Color	Negro azabache

Fuente: (Masterairscrew, 2018)

2.7 ESC (ELECTRONIC SPEED CONTROLLER)

El controlador electrónico de velocidad es un circuito que tiene como objetivo controlar la velocidad y el sentido de giro de un motor electrónico, además, algunos ESC de mayor calidad tienen una frecuencia de actualización mucho más rápida para comprobar el número de veces de nuevas instrucciones de la tarjeta de control, logrando ajustar la velocidad del motor y brindando estabilidad para su óptimo funcionamiento.

Los ESC se clasifican principalmente para la cantidad de corriente que puede suministrar constantemente al motor.

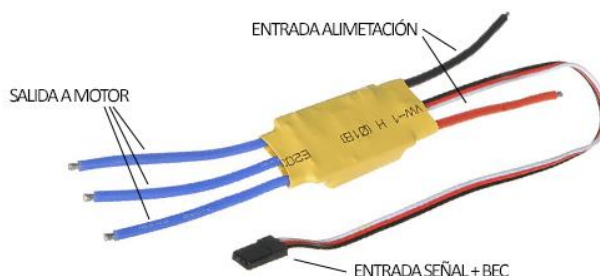


Figura 9 Controlador Electrónico de Velocidad

Fuente: (Taloselectronics, 2018)

Los controles de velocidad electrónicos son categorizados según la corriente a la que se puede someter, la corriente pico que genera que genera el arranque de un motor eléctrico es un factor importante para considerar a la hora de elegir el controlador que se va a utilizar, entre más corriente soporte el ESC más elevado será su costo.

2.8 SOFTWARE ARDUINO

El software de Arduino cuenta con un Entorno de Desarrollo Integrado para casi todas las plataformas (Windows, Linux, Mac), donde podemos escribir las aplicaciones, descargarlas al Arduino y ejecutarlas o depurarlas desde allí.

El microcontrolador en la placa Arduino se programa mediante el lenguaje de programación Arduino basado en Wiring y en el entorno de desarrollo Arduino basado en Processing.(FERNANDO, 2017)

2.8.1 HARDWARE ARDUINO

El hardware es la placa que cualquiera puede hacerla es por esa razón que hay distintos modelos de Arduino, el más sencillo consiste en una placa con un microcontrolador y una serie de puertos de entrada y salida.

Los microcontroladores más utilizados son Atmega168, Atmega328, Atmega1280 y Atmega8 por su sencillez y bajo costo que permiten desarrollar múltiples diseños.

2.8.2 ARDUINO

Arduino es una plataforma de electrónica de código abierto (Open-Source) basada en hardware y software flexibles y fáciles de usar, diseñada para facilitar el uso de la electrónica en proyectos de innovación.

El Arduino se puede utilizar para desarrollar elementos autónomos, también puede conectarse a otros dispositivos o interactuar con otros programas para interactuar en su hardware y software (FERNANDO, 2017).

2.8.3 ARDUINO UNO

El Arduino Uno es una placa con un microcontrolador Atmega 328 que tiene 14 entradas/salidas digitales, 6 salidas que se puede utilizar como PWM, una conexión USB, un Jack de poder, un cabezal ICSP y un botón de reinicio.

2.9 APLICACIONES DE LOS DRONES EN AMBITO CIVIL

Al paso de los años las aplicaciones y usos de los drones en el ámbito civil crecen generalmente en la obtención de imágenes aéreas las cuales dan privilegiados puntos de vista, según (International, 2016) menciona varias aplicaciones tales como en:

- Logística. - Los drones pueden realizar diferentes tareas que se encuentran involucradas con el transporte de productos entre empresas facilitando el medio de comunicación entre estas.
- Mantenimiento. - Los drones pueden realizar tareas de monitoreo de varios sectores que necesiten el registro constante de sus plantas de producción, de esto surge varias aplicaciones tales como revisión del estado de líneas eléctricas, generadores entre otros.
- Agricultura: Los drones en este campo sirven de gran ayuda ya que llevan registro de lo que pasa en tiempo real diagnosticando diferentes anomalías que ocurren en los cultivos.
- Seguridad: Los drones son empleados para el reconocimiento de personas debido a que vuelan a bajas alturas, hoy en día son utilizados por la policía para el control de los sectores vulnerables de delincuencia.
- Periodismo y fotografía: los drones son perfectos para almacenar mediante cámara y video diferentes eventos sociales, aquí surge una nueva aplicación para estos UAVs.

CAPÍTULO III

IMPLEMENTACIÓN DE UNA INTERFAZ TELECONTROL PARA UNA PLATAFORMA AEREA MULTI-ROTOR MEDIANTE MATLAB

3.1 PRELIMINARES

En este capítulo se detalla el desarrollo del control del Drone, con énfasis en la metodología y los procedimientos de montaje de la plataforma.

Los elementos utilizados para el drone se en listan a continuación:

- Frame o Estructura: F450 455mm. Brazos de nylon poliamida, centro de fibra de vidrio.
- 4 motores Brushless: D2212-920KV
- 4 controladores (ESC): 40A
- 1 tren de aterrizaje
- Sistema de estabilización de vuelo: DJI NAZA M LITE
- 4 hélices: Diámetro 254mm.
- Antena 3dr: 915Mhz
- Batería LiPo: 3S 5000mAh
- Arduino
- Joystick

3.2 ENSAMBLE DEL CUADRICOPTERO

Para realizar el ensamblaje del cuadricóptero se contempló varios pasos considerando diversos factores como el cableado y el equilibrio del drone como se indica a continuación:

- El primer paso es el acoplamiento de los motores brushless hacia la estructura donde se necesitó tornillos con el diámetro del motor y las ranuras del frame como se muestra en la figura 10.

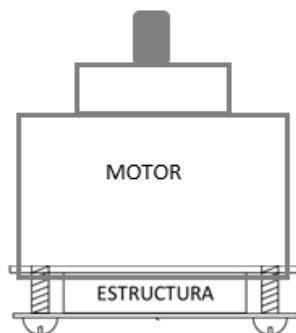


Figura 10 Acoplamiento de los motores brushless con la estructura

- Una vez ensamblado los motores en la estructura se conectó con los controladores de velocidad (ESC), procurando protegerlos para evitar cortocircuitos como se muestra en la figura 11. Para elegir los ESC hay que tomar en cuenta la corriente pico generada por el motor en el arranque a velocidad nominal.

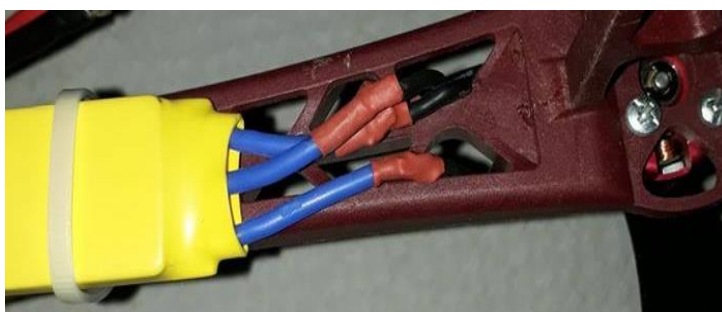


Figura 11 Protección con termo-retráctil

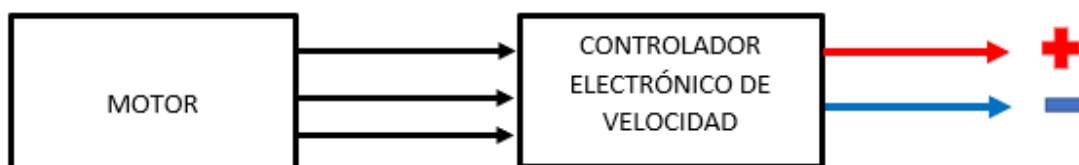


Figura 12 Diagrama de conexión

- Los terminales de la alimentación de los controladores de velocidad van conectados en paralelo a la placa de fibra de vidrio del frame ayudando a disminuir la cantidad de cableado, de una terminal de alimentación se extiende el cable positivo y negativo para poder conectarse con la batería LiPo, teniendo en cuenta la utilización de un conector Bullet, también se procede a alimentar el controlador principal (MC) DJI NAZA M LITE que está conectado a un regulador de voltaje de 5V que da alimentación al mismo y este a su vez esta alimentado de la batería LiPo.
- Al terminar la conexión de alimentación de los controladores de velocidad (ESC), vamos a los conectores de la señal que van conectados al controlador principal (MC) DJI NAZA M LITE como se muestra en la figura 13.

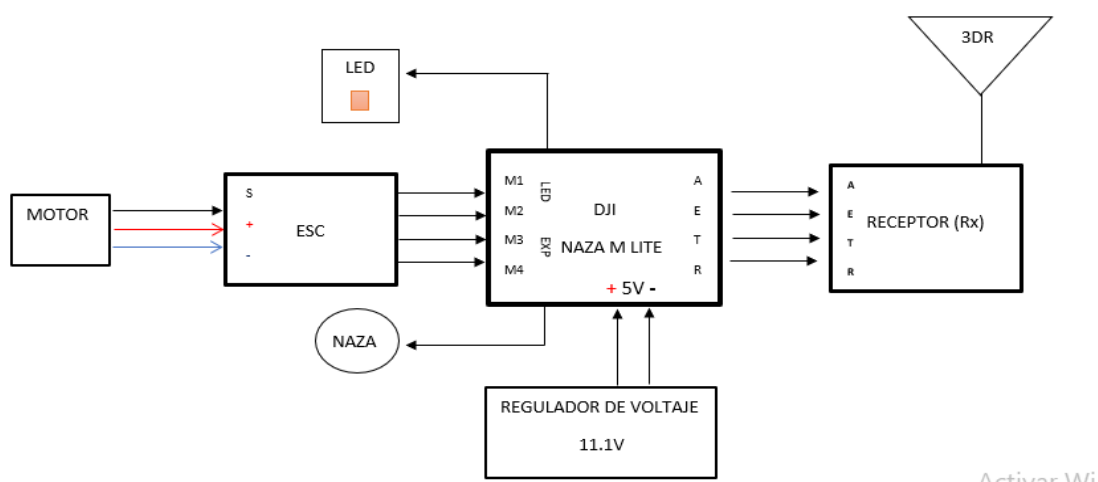




























Figura 13 Conexión del Control principal (MC) con los controladores (ESC)

- Una vez conectadas las señales de los ESC al controlador principal (MC) DJI NAZA M LITE, se identifica las señales de salida del mismo en las que se tiene las señales para el RUDD, ELEV, AILE, THRO, también se conecta el módulo GPS y un LED de indicación que indicara el modo de control en el que se encuentra el GPS como se indica la figura 14, también tiene una interfaz USB para configurar Unidad NAZA-M LITE y la actualización de firmware.

System Status	LED Flashing
System start and self-check	
IMU abnormal data	
Warm up after power on	
Bias of Sensors too Big	
Compass Error too Big	
Tx signal lost	
Low Voltage Alert	
Record forward direction or home point	
Control Mode Indicator	Manual Mode: None ATTI Mode:  (stick(s) not at center ) GPS Mode:  (stick(s) not at center ) IOC Mode:  (stick(s) not at center   )
GPS Signal State Indicator (GPS/Compass Module is necessary)	GPS Signal is Best(GPS Satellite number > 6):  None GPS Signal is Well(GPS Satellite number = 6):    GPS Signal is Bad (GPS Satellite number = 5):   GPS Signal is Worst (GPS Satellite number < 5):    




Compass Calibration	LED Flashing
Begin horizontal calibration	
Begin vertical calibration	
Calibration or others error	

Figura 14 Indicaciones para configuración del GPS

Fuente: (DJI NAZA M LITE, 2014)

- Al identificar las señales de salida del controlador principal (MC) DJI NAZA M LITE se procede a conectar con el Arduino Uno, este al mismo tiempo se encuentra conectado con una antena 3Dr que sirve como receptor para recibir una señal enviada desde la PC mediante Matlab controlada por un Joystick (EXTREME 3D PRO).



Figura 15 Joystick (Extreme 3D Pro)

3.3 CONFIGURACIÓN DEL SOFTWARE NAZA M LITE

Al recibir una señal al controlador principal (MC) DJI NAZA M LITE se procede a realizar la configuración mediante el software NAZA M LITE



Figura 16 Icono del software DJI NAZAM

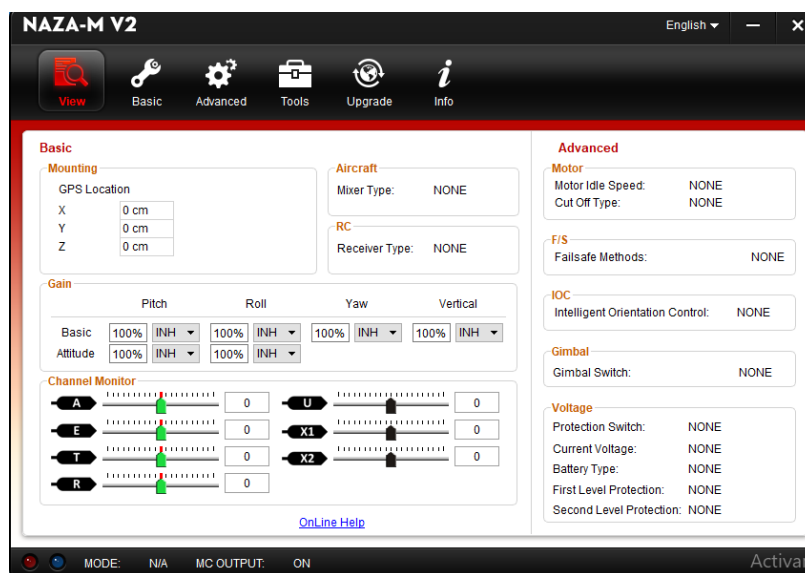


Figura 17 Interfaz principal del software NAZA-M V2

Se conecta el DJI NAZA hacia la PC para reconocer el sistema en el software de configuración, en la pantalla el led color azul (MODE) titilará e indicará que ha sido reconocido el sistema estabilizador de vuelo DJI NAZA en el software. A demás se indica los valores de voltaje que se mostrará y cambiarán en tiempo real.

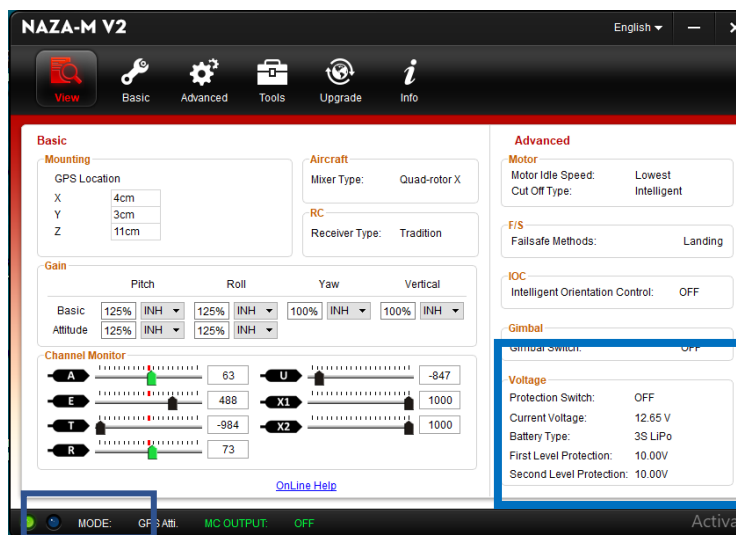


Figura 18 Reconocimiento del hardware en el software NAZA-M V2

Una vez reconocido el hardware se puede comprobar el movimiento de los diferentes ejes (canales) del UAV mediante el joystick.

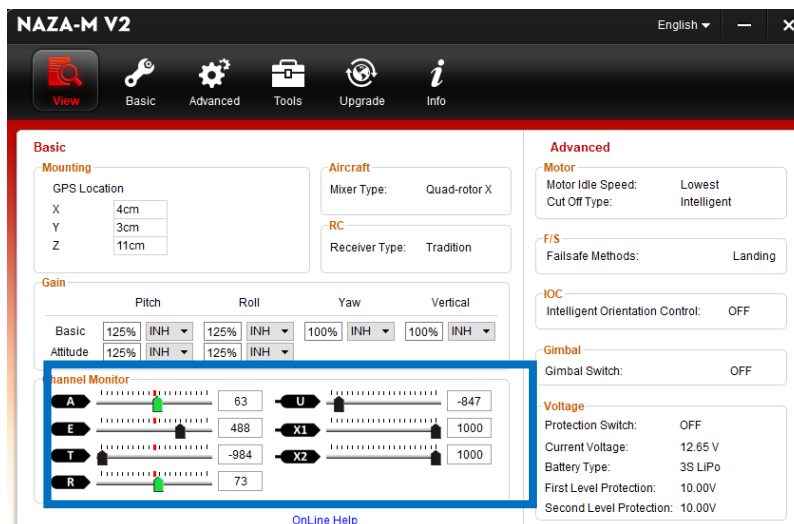


Figura 19 Verificación de los movimientos de los canales

Ingresar en la opción Basic/Aircraft, seleccionar el tipo de mezclador (MIXER TYPE) según sea la conveniencia, en este proyecto técnico se seleccionó un Quad-rotor Tipo X por sus características de funcionamiento, el software nos indica el sentido de giro de cada uno de los motores del UAV que están etiquetados como M1-M2-M3-M4 respectivamente.

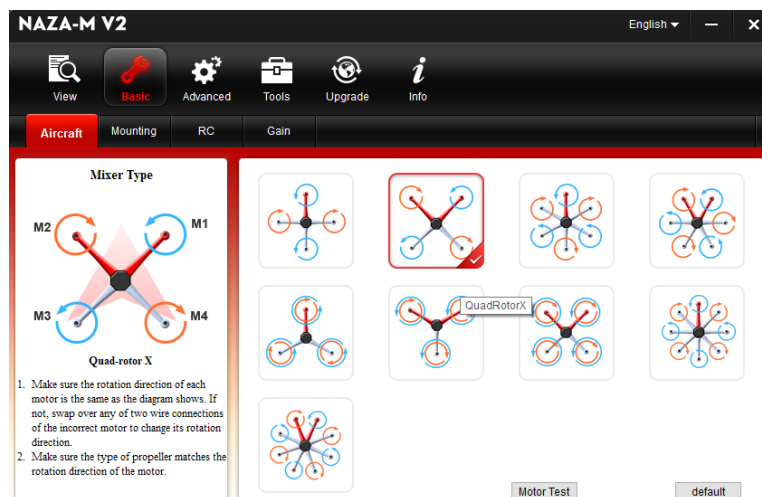


Figura 20 Selección del Quad-rotor Tipo X

Al ingresar en la opción Basic/Mounting, aparecen los valores de la posición donde estará ubicado el GPS en la estructura del dron; $X=4\text{cm}$, $Y=3\text{cm}$, $Z=11\text{cm}$. Además, el software de configuración indica como debe ser montado el sistema de estabilización de vuelo en la estructura del dron, éste ofrece una referencia para la instalación del kit de DJI NAZA.

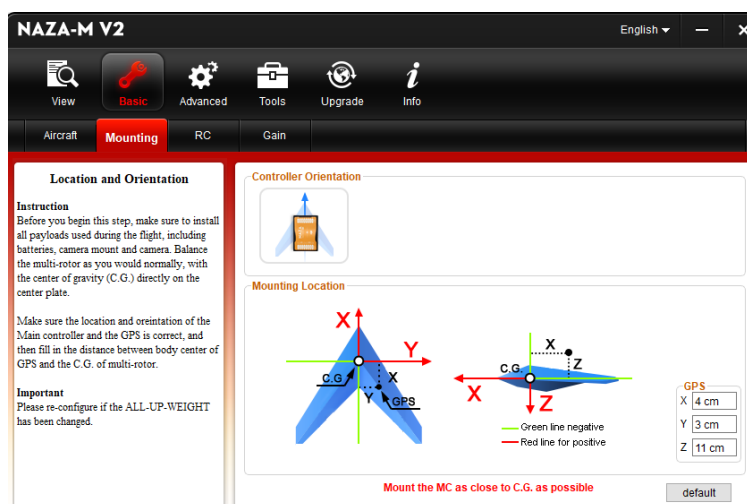


Figura 21 Valores de la ubicación exacta del GPS

Ingresar en la opción Basic/RC, Seleccionar el tipo de receptor "TRADITION", Comprobar el movimiento de los Command Sticks y calibrarlos, escogiendo el modo de control de switch en los que tenemos:

- FAILSAFE

El modo de control failsafe no es aconsejable utilizar ya que perderá el control del vehículo aéreo no tripulado (UAV), salvo en casos de emergencia se lo puede utilizar ya que precede al aterrizaje de este.

- ATTI

El control en modo atti ayuda a mantener una actitud estable pero no es capaz de controlar el desplazamiento a causa de factores como en este caso es el viento.

- MANUAL

El modo manual es un vuelo acrobático que es utilizado o recomendado para personas con experiencia en el manejo de este.

Finalmente seleccionar el “Modo de vuelo GPS” que es el que se utilizó, ya que este modo cuenta con un control de orientación inteligente y es capaz de controlar el desplazamiento que es causado por el viento.

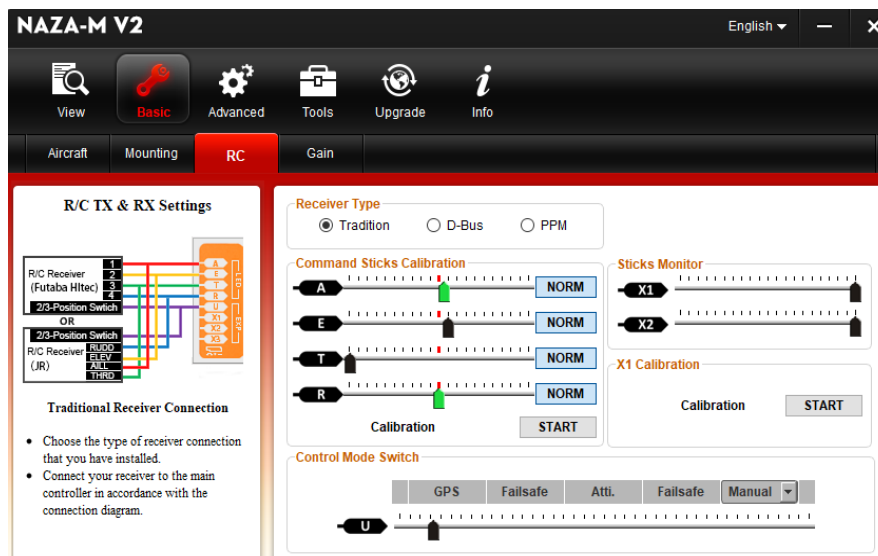


Figura 22 Selección de Tipo de Receptor y Modo de Vuelo

Ingresar en la opción Basic/Gain, se puede observar el valor de las ganancias para el control óptimo del vehículo aéreo no tripulado.

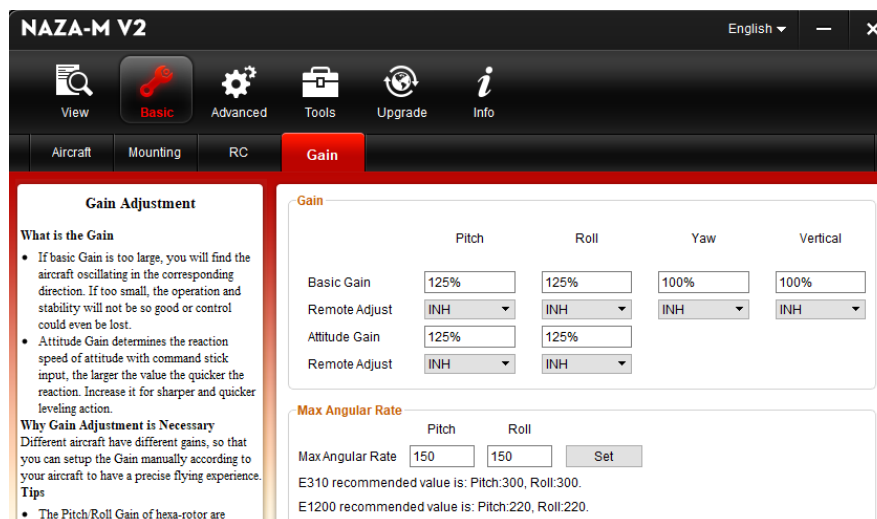


Figura 23 Valores de ganancia del UAV

Ingresar en la opción Advanced/Motor, se selecciona LOW para que los motores arranquen con velocidad baja, al arrancar con baja velocidad se evita corrientes picos que puedan afectar el buen funcionamiento de los motores y también se selecciona el corte tipo “INTELLIGENT”.

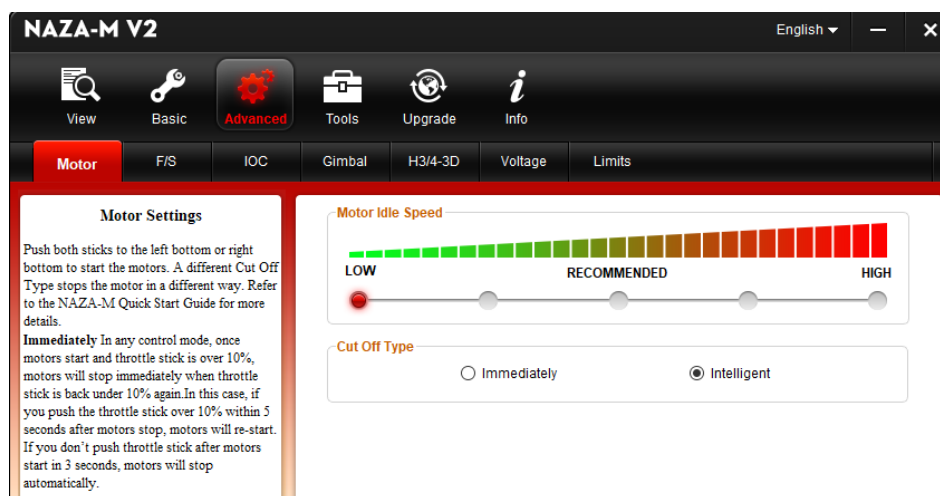


Figura 24 Selección de velocidad de arranque de los motores

Ingresar en la opción Advanced/Motor, colocar en el primer y el segundo nivel el valor de protección para descarga de batería, lo más recomendable es no permitir que la tensión de la LiPo 3S descienda menos de los 10 v.

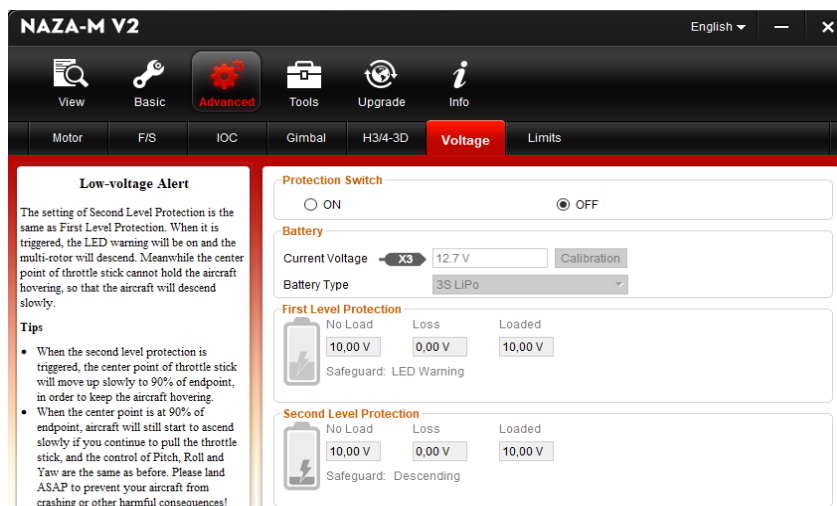


Figura 25 Valores para protección de descarga de LiPo 3S

Ingresar en Tools y verificar que todos los parámetros antes mencionados estén correctos para exportar los datos hacia el controlador. Finalmente, chequear el estado del IMU y exportar los parámetros.

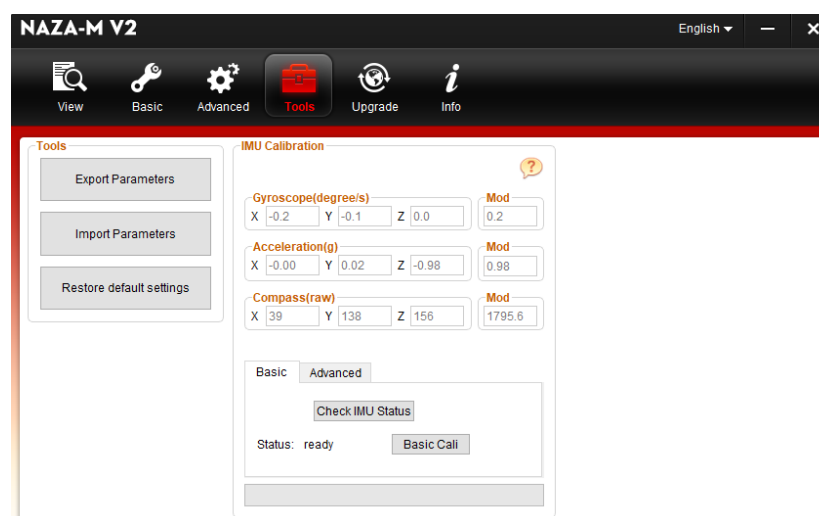


Figura 26 Exportación de parámetros hacia el controlador DJI NAZA

3.4 DESARROLLO DE LA INTERFAZ

3.4.1 INTERFAZ EN EL SOFTWARE MATLAB GUI

Matlab GUI es un entorno que permite realizar interfaces graficas que hace fácil la interacción entre el usuario y el programa en ejecución. Los diferentes elementos que se puede encontrar en el entorno GUI factibles para la programación son: botones, deslizadores, textos de descripción, menú de selección entre otros como se muestra en la figura 2.26.

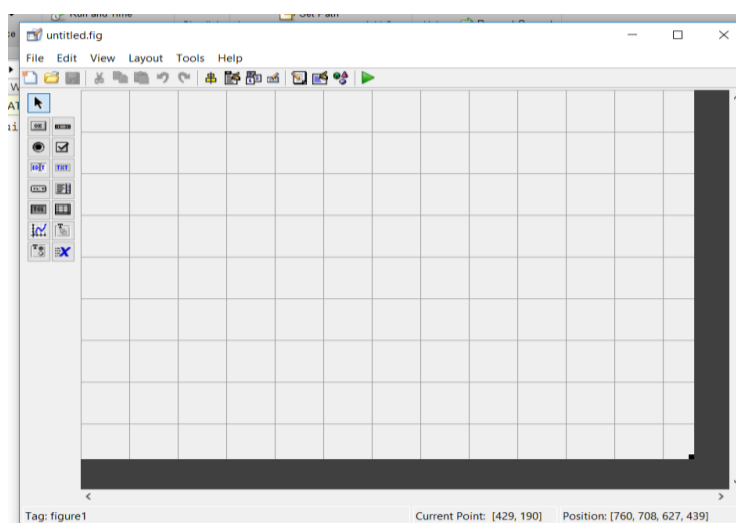


Figura 27 Interfaz Grafica

Al crear una nueva interfaz gráfica se abre una ventana del panel principal que es el entorno de desarrollo donde van los elementos de acuerdo a nuestras necesidades, al realizar la compilación del entorno GUI automáticamente se generan dos archivos el M-file (.m) el cual almacena las funciones asociadas con cada elemento que contiene toda la programación de la interfaz gráfica y el Fig-file (.fig) que contiene la descripción completa del GUI siendo así que los dos archivos funcionan de manera acoplada, es decir, cada elemento que consta en el archivo .fig es un elemento programable a través de sus propiedades o Callbacks ya que tiene una sentencia de ejecución en el archivo .m la cual se lleva a cabo cada vez que el usuario modifique las propiedades de cada elemento.

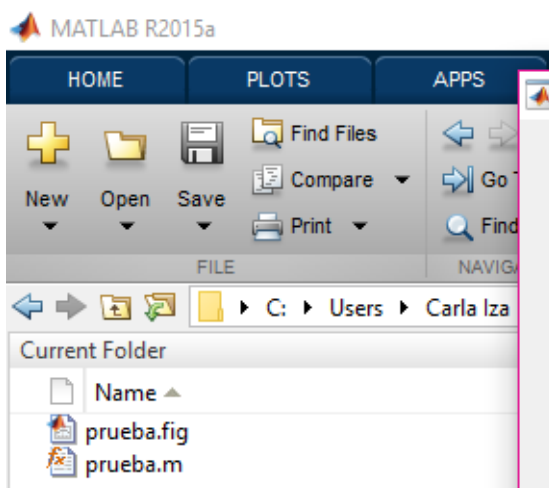


Figura 28 Archivos M-file (.m) y Fig-file (.fig)

Cada elemento que se utilice en la programación cuenta con propiedades para modificar su color, el string entre otras modificaciones

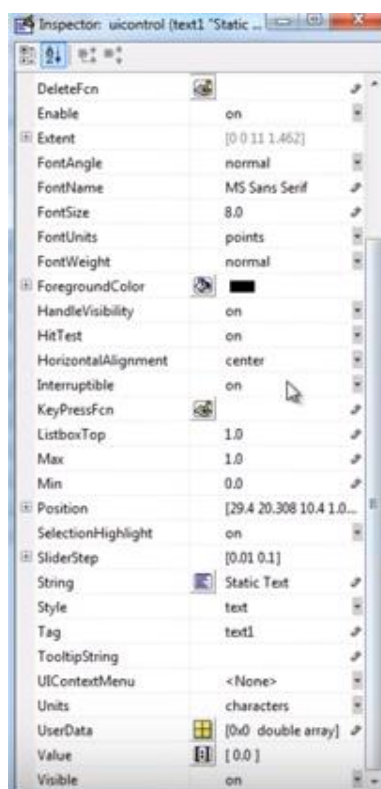


Figura 29 Propiedades de cada elemento

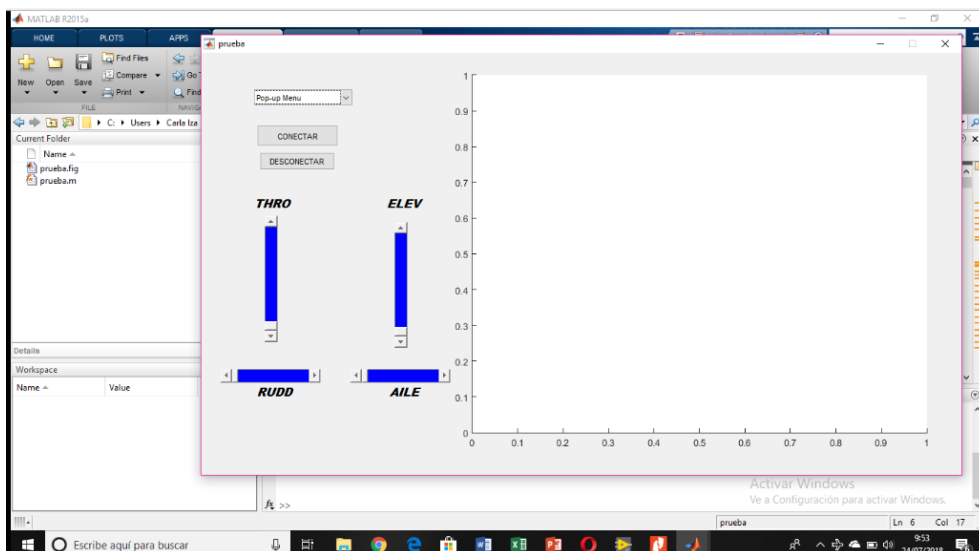


Figura 30 Interfaz Principal del Telecontrol

3.4.2 DESCRIPCION DE LOS ELEMENTOS UTILIZADOS EN LA PROGRAMACION

A. ELEMENTO “AXES”

Este elemento sirve para mostrar gráficos e imágenes en el entorno GUI, es así como la selección de puntos comprendidos en su interior indica cómo se está manipulando el joystick y como el dron sigue sus señales.

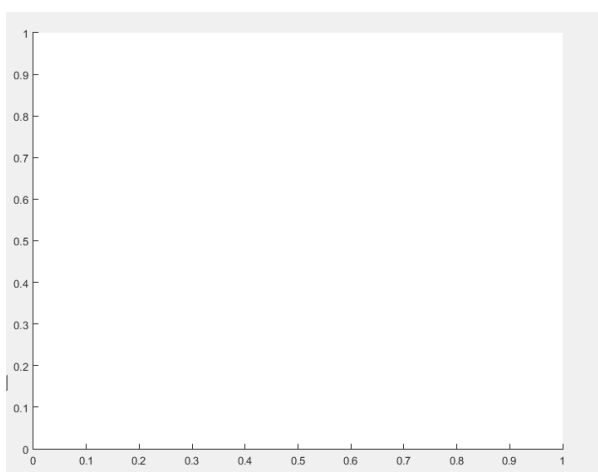


Figura 31 Elementos “AXES” de la interfaz de Navegación

B. ELEMENTO “PUSH BUTTON”

Este elemento tiene su propio callback es decir es la propiedad que nos permite ejecutar una acción al momento de presionar el botón, los botones utilizados en esta programación son:

- **BOTÓN “CONECTAR”**

Genera la conexión de la trama preestablecida en la programación y se empieza a realizar el control del UAV.

- **BOTÓN “DESCONECTAR”**

Corta la comunicación con la antena receptora 3DR, suspendiendo la comunicación serial utilizado durante todo este proceso.

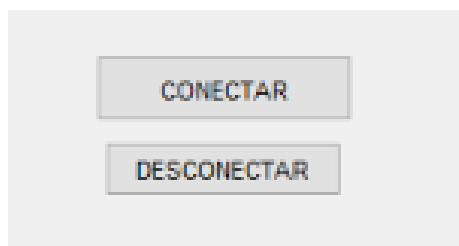


Figura 32 Botones de indicación

C. ELEMENTO “POP-UP MENU”

Este elemento permite elegir el puerto de comunicación serial utilizado durante todo este proceso.



Figura 33 Elemento Pop-Up Menu

D. ELEMENTO “SLIDER”

Este elemento puede representar un valor de rangos mediante el deslizamiento de un indicador. En esta programación se incorporó cuatro elementos como se muestra en la figura 34, cada uno brinda una acción diferente sobre los ejes de rotación.

- **THRO**

Este slider indica la velocidad de giros de los cuatro motores, es decir este elemento controla la altitud de vuelo del UAV.

- **RUDD**

Este elemento ejerce un cambio en la velocidad de los motores que giran en oposición, disminuyendo la velocidad de un par de motores mientras que eleva la del otro par provocando que el UAV gire horizontalmente sobre su eje.

- **ELEV**

En este slider la velocidad de giro de los motores laterales es constante, mientras que uno de los motores longitudinales aumenta la velocidad y el otro motor disminuye su velocidad provocando un desplazamiento longitudinal.

- **AILE**

Este elemento a diferencia del slider ELEV mantiene constante la velocidad de giro de los motores longitudinales, disminuyendo la velocidad de uno de los motores laterales y aumenta la velocidad del otro, consiguiendo un desplazamiento lateral.

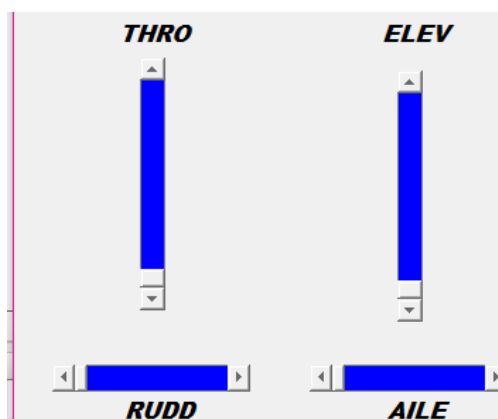
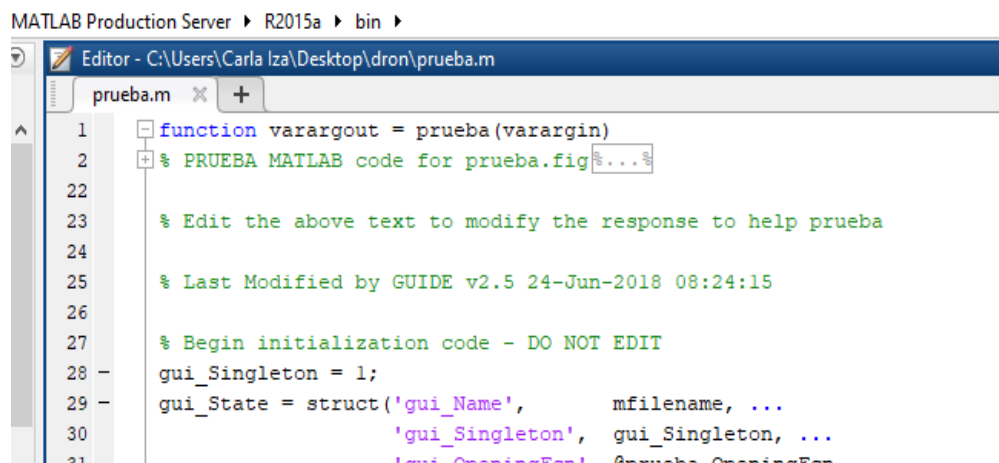


Figura 34 Slider de control

Después de haber realizado la interfaz en GUI se crea automáticamente un código que la interfaz necesita para su funcionamiento.

- Al inicio de la programación se crea la función `varargout` que quiere decir que va a declararse el último argumento de una función, recopilando todas las entradas o salidas desde ese punto en adelante.



```

MATLAB Production Server > R2015a > bin >
Editor - C:\Users\Carla Iza\Desktop\dron\prueba.m
prueba.m x +
1 function varargout = prueba(varargin)
2 % PRUEBA MATLAB code for prueba.fig
22
23 % Edit the above text to modify the response to help prueba
24
25 % Last Modified by GUIDE v2.5 24-Jun-2018 08:24:15
26
27 % Begin initialization code - DO NOT EDIT
28 - gui_Singleton = 1;
29 - gui_State = struct('gui_Name',      mfilename, ...
30                    'gui_Singleton',  gui_Singleton, ...
31                    'gui_OpeningFcn', @prueba_OpeningFcn

```

Figura 35 Inicio de la programación

- Al mismo tiempo que se creó la interfaz en GUI con cada elemento se programó automáticamente partes de su programación.

```

93 % --- Executes during object creation, after setting all properties.
94 function listaCOM_CreateFcn(hObject, eventdata, handles)
95 % hObject    handle to listaCOM (see GCBO)
96 % eventdata  reserved - to be defined in a future version of MATLAB
97 % handles    empty - handles not created until after all CreateFcns called
98
99 % Hint: popupmenu controls usually have a white background on Windows.
100 %     See ISPC and COMPUTER.
101 - if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
102     set(hObject,'BackgroundColor','white');
103 - end
104
105 % --- Executes on button press in btnCon.

```

Figura 36 Programación de la interfaz GUI

- En nuestro caso toda la programación se lleva a cabo en la función del botón “ CONECTAR” ya que este botón al ser seleccionado inicia toda la programación, que la describire a continuación:
 - Se programa global run que indica la ejecución del programa/ lista de menú que da función a un la conexión de un puerto USB.
 - Luego se programa las características del puerto en las que se configura la velocidad a 57600 baudios/ bit de parada / el dato de bit que va entre 5 y 8 / la paridad.
 - Al terminar la configuración se indica el carácter de finalizar el envío siendo esto el número de bytes que se va a enviar o recibir durante un tiempo establecido.

```

Editor - C:\Users\Carla Iza\Desktop\dron\prueba.m
prueba.m x +
105 % --- Executes on button press in btnCon.
106
107 function btnCon_Callback(hObject, eventdata, handles)
108 % hObject handle to btnCon (see GCBO)
109 % eventdata reserved - to be defined in a future version of MATLAB
110 % handles structure with handles and user data (see GUIDATA)
111
112 global run;
113 run=1;
114 contenido = cellstr( get( handles.listaCOM, 'String' ) );
115 puerto = contenido{ get( handles.listaCOM, 'Value' ) };
116 S = serial( puerto );
117 set( S, 'Baudrate', 57600 ); % se configura la velocidad a 57600 Baudios
118 set( S, 'StopBits', 1 ); % se configura bit de parada a uno
119 set( S, 'DataBits', 8 ); % se configura que el dato es de 8 bits, debe estar entre 5 y 8
120 set( S, 'Parity', 'none' ); % se configura sin paridad
121 set( S, 'Terminator', 'CR/LF' ); % caracter con que finaliza el envío
122 set( S, 'OutputBufferSize', 50 ); % es el numero de bytes a enviar
123 set( S, 'InputBufferSize', 50 ); % es el numero de bytes a recibir
124 set( S, 'Timeout', 5 ); % 5 segundos de tiempo de espera
125 fopen( S );
  
```

Figura 37 Programación de configuración del puerto USB

- Al finalizar la configuración del puerto USB se realiza la programación para el control del joystick en el que se calibro el punto exacto del GPS, la desactivación de los motores y cada uno de los controles que va a realizar en este caso THRO, AILE, ELEV, RUDD.


```

Editor - C:\Users\Carla Iza\Desktop\dron\prueba.m
prueba.m x +
126 - k=vrjoystick(1);
127 - cont=0;
128 - run=1;
129 - gps=859;
130 - cali=gps;
131
132 - axes(handles.axes1)
133 - axis;
134
135 - while(run==1)
136
137 - [ejes,botones] = read(k);
138 - v_elev=round((-ejes(2)+1)*500);
139 - v_aile=round((ejes(1)+1)*500);
140 - v_rudd=round((ejes(3)+1)*500);
141 - v_thro=round((-ejes(4)+1)*500);
142
143 - set(handles.elev,'value',v_elev);
144 - set(handles.aile,'value',v_aile);
145 - set(handles.rudd,'value',v_rudd);
146 - set(handles.thro,'value',v_thro);
147
148 - if (abs(v_elev-500)<=10) v_elev=500; end
149 - if (abs(v_rudd-500)<=10) v_rudd=500; end
150 - if (abs(v_aile-500)<=10) v_aile=500; end
151
152 - if(botones(1)==1);

```

Figura 38 Programación de control de los slider

3.5 COMUNICACIÓN INALÁMBRICA

Para realizar la comunicación inalámbrica entre Matlab y el UAV se contempló la necesidad de utilizar una antena telemétrica 3DR de 915 Mhz.

La verificación de la comunicación entre la antena transmisora y la receptora se utilizó el software SikRadio Config 1.4 como se muestra en la figura.



Figura 39 Identificación de cada elemento de la interfaz

3.5.1 CONFIGURACION DE LA ANTENA 3DR

Para configurar la antena telemétrica 3DR tanto transmisor como receptor se alimenta a RX con 5 V debido a que es tecnología TTL como se muestra en la figura y el transmisor se conecta a un puerto USB de la computadora como se muestra en la figura.

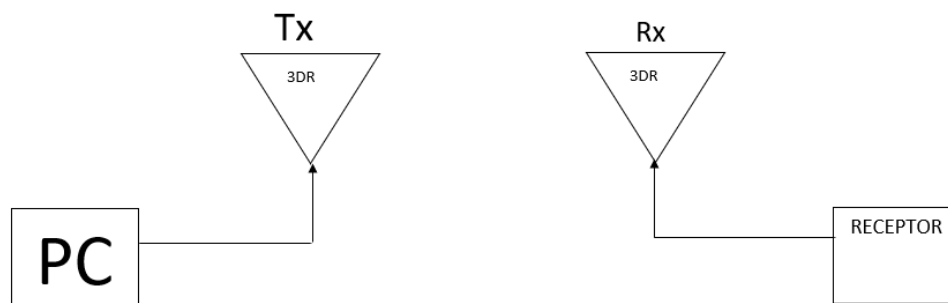


Figura 40 Alimentación de receptor y el transmisor

Una vez alimentadas las antenas en el software se elige el puerto en la que la antena transmisora se encuentra conectada, al momento de configurar la antena transmisora se puede constatar que la velocidad de transmisión de la antena 3DR es 57600 bps (baudios por segundo) este valor viene dado directamente del fabricante.

Seguidamente se da clic en Load Settings obteniendo diferentes parámetros tanto del transmisor como del receptor, es importante mencionar que para tener una conexión fiable los valores mostrados en el programa deben ser iguales, como se muestra en la figura 41, ya que si uno es diferente no existe la transmisión de datos.



Figura 41 Parámetro de Tx y Rx

Para verificar que exista comunicación entre el transmisor (Tx) y el receptor (Rx): clic RSSI / revisión del puerto / revisión de la velocidad de transmisión / connect.

Para comprobar una buena comunicación la línea roja y verde de la siguiente figura 42 deben estar relacionadas entre sí.

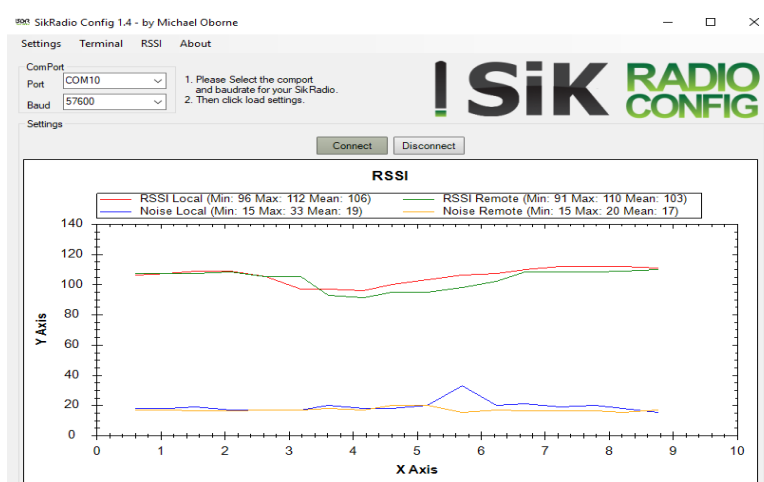


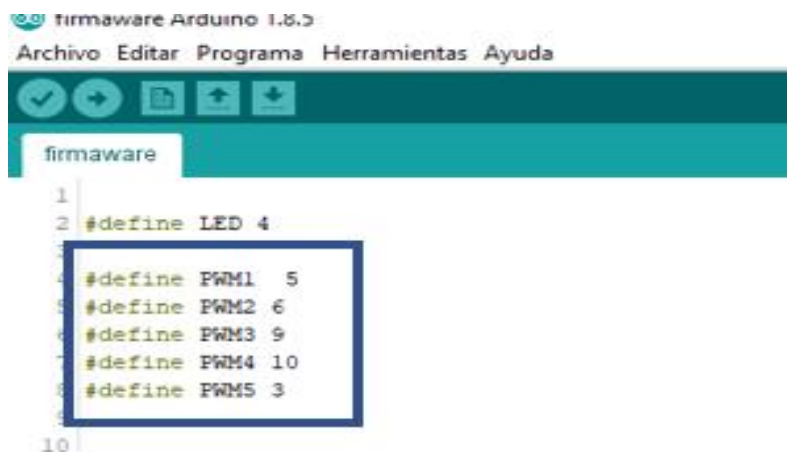
Figura 42 Verificación de Comunicación entre Tx y Rx

3.6 PROGRAMACION EN ARDUINO

3.6.1. RECEPCION DE DATOS

Una vez realizada interfaz de programación en Matlab se procede a realizar la programación en Arduino Uno como se muestra a continuación:

- Primero se realizó la definición de los pines que tiene Arduino que generan PWM.



```

firmware
1
2 #define LED 4
3
4 #define PWM1 5
5 #define PWM2 6
6 #define PWM3 9
7 #define PWM4 10
8 #define PWM5 3
9
10

```

Figura 43 Pines que generan PWM en Arduino Uno

- El enunciado de “void setup” se incluyó el inicio de la comunicación serial para la conexión con Matlab a una velocidad de 97600 baudios

```

void setup() {
  Serial.begin(57600);
  // put your setup code here, to run once:
  pinMode(LED, OUTPUT);
  pinMode(PWM1, OUTPUT);
  pinMode(PWM2, OUTPUT);
  pinMode(PWM3, OUTPUT);
  pinMode(PWM4, OUTPUT);
  pinMode(PWM5, OUTPUT);
}

```

Figura 44 Establece la velocidad de datos en bits por segundo (baudios) para la transmisión de datos en serie

- Se realiza la programación para establecer el movimiento que van a generar el RUDD, THRO, ELEV, AILE.

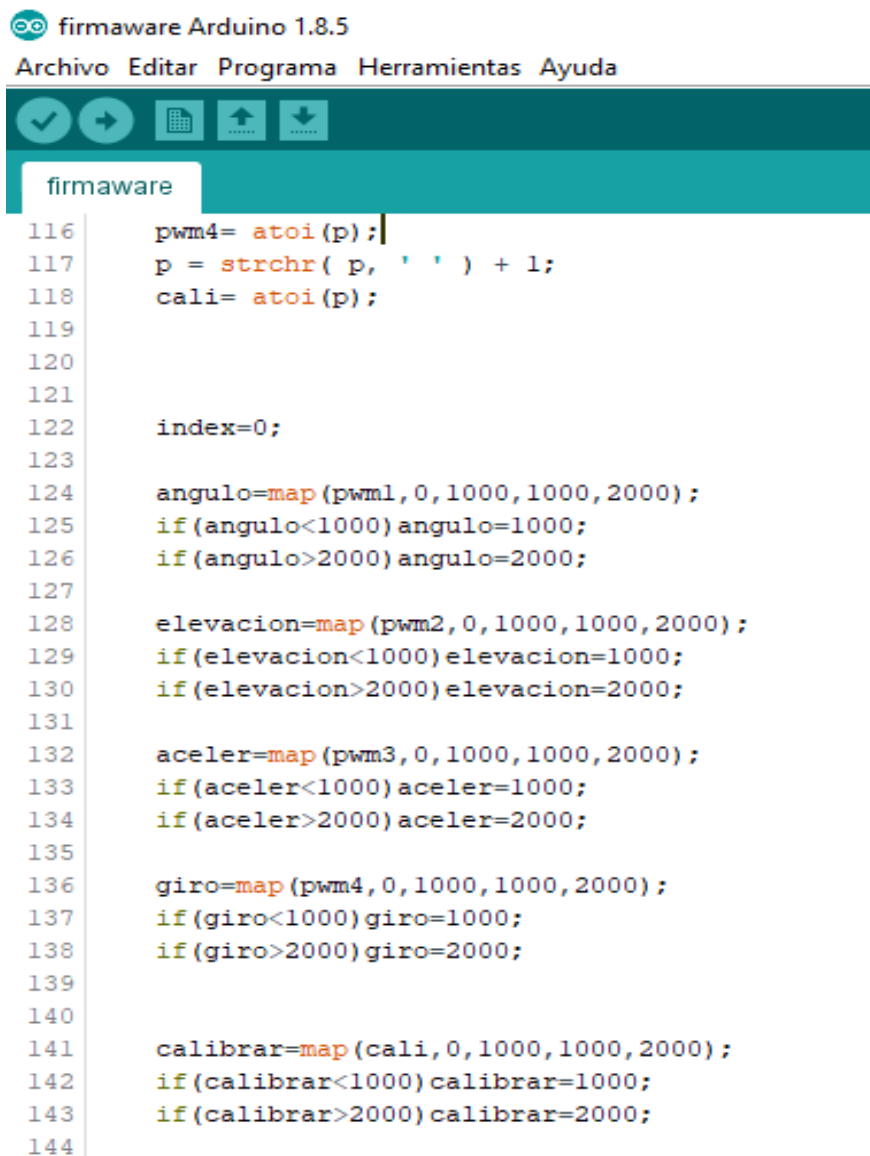
```

44
45 void loop() {
46     serialEvent();
47     if(micros()-t1>tiempo) {
48         t1=micros();
49         digitalWrite(PWM1,HIGH);
50         delayMicroseconds(angulo);
51         digitalWrite(PWM1,LOW);
52     }
53
54     if(micros()-t2>tiempo) {
55         t2=micros();
56         digitalWrite(PWM2,HIGH);
57         delayMicroseconds(elevacion);
58         digitalWrite(PWM2,LOW);
59     }
60 }
61
62     if(micros()-t3>tiempo) {
63         t3=micros();
64         digitalWrite(PWM3,HIGH);
65         delayMicroseconds(aceler);
66         digitalWrite(PWM3,LOW);
67     }
68 }
69     if(micros()-t4>tiempo) {
70         t4=micros();

```

Figura 45 “serialEvent” permite recibir datos seriales de una forma parecida a una interrupción o de forma asíncrona

- Se define la posición del RUDD, THRO, ELEV, AILE para desbloquear motores y así poder controlar al UAV.



```
firmaware
116     pwm4= atoi(p);
117     p = strchr( p, ' ' ) + 1;
118     cali= atoi(p);
119
120
121
122     index=0;
123
124     angulo=map(pwml, 0, 1000, 1000, 2000);
125     if(angulo<1000) angulo=1000;
126     if(angulo>2000) angulo=2000;
127
128     elevacion=map(pwm2, 0, 1000, 1000, 2000);
129     if(elevacion<1000) elevacion=1000;
130     if(elevacion>2000) elevacion=2000;
131
132     aceler=map(pwm3, 0, 1000, 1000, 2000);
133     if(aceler<1000) aceler=1000;
134     if(aceler>2000) aceler=2000;
135
136     giro=map(pwm4, 0, 1000, 1000, 2000);
137     if(giro<1000) giro=1000;
138     if(giro>2000) giro=2000;
139
140
141     calibrar=map(cali, 0, 1000, 1000, 2000);
142     if(calibrar<1000) calibrar=1000;
143     if(calibrar>2000) calibrar=2000;
144
```

Figura 46 Desbloqueo de motores

- Una vez concluida la programación se procede a realizar la compilación y verificación de algún error en el sketch.
- Comprobada la compilación se procede a subir el programa.



```
firmaware Arduino 1.8.5
Archivo Editar Programa Herramientas Ayuda

firmaware
116   pwm4= atoi(p);
117   p = strchr( p, ' ' ) + 1;
118   cali= atoi(p);
119
120
121
122   index=0;
123
124   angulo=map(pwm1,0,1000,1000,2000);
125   if(angulo<1000)angulo=1000;
126   if(angulo>2000)angulo=2000;
127
128   elevacion=map(pwm2,0,1000,1000,2000);
129   if(elevacion<1000)elevacion=1000;
130   if(elevacion>2000)elevacion=2000;
131
132   aceler=map(pwm3,0,1000,1000,2000);
133   if(aceler<1000)aceler=1000;
134   if(aceler>2000)aceler=2000;
135
136   giro=map(pwm4,0,1000,1000,2000);
137   if(giro<1000)giro=1000;
138   if(giro>2000)giro=2000;
139
140
141   calibrar=map(cali,0,1000,1000,2000);
142   if(calibrar<1000)calibrar=1000;
143   if(calibrar>2000)calibrar=2000;
144

Compilado
El Sketch usa 3734 bytes (11%) del espacio de almacenamiento de programa. El
Las variables Globales usan 495 bytes (24%) de la memoria dinámica, dejando
```

Figura 47 Compilación y verificación de error en el sketch

3.7 ANALISIS DE RESULTADO

Una vez terminado el drone se procedió a realizar las pruebas para comprobar su comportamiento y estabilidad, en lo que se concluye que el control multi-rotor es viable para realizar la ejecución de vuelo gracias a la interfaz desarrollada en el software Matlab GUI controlada por un Joystick (EXTREME 3D PRO).



Figura 48 Drone ensamblado

En la siguiente figura se puede constatar la eficiencia de la navegación, cabe recalcar que la estabilidad de vuelo tiene una variación de posición que es generada por las perturbaciones del viento.

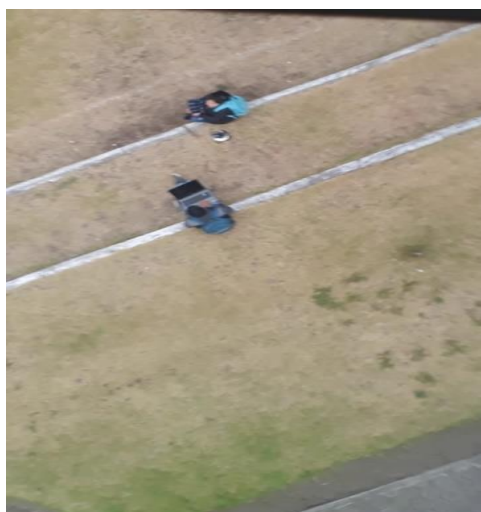


Figura 49 Drone en vuelo

Finalmente, en la siguiente figura se muestran los elementos y herramientas necesarias para realizar el control de la plataforma multirotor mediante la interfaz Matlab.



Figura 50 Elementos para realizar el control

CAPITULO IV

4.1 Conclusiones

- Se logro demostrar que la implementación de una interfaz telecontrol de una plataforma aérea multi-rotor mediante Matlab fue eficiente.
- El algoritmo de programación de Matlab GUI cuenta con diferentes elementos que permite realizar modificaciones personalizando la interfaz del telecontrol para la plataforma aérea.
- Al realizar la implementación para establecer la comunicación entre la plataforma aérea y la interfaz de Matlab su respuesta fue inmediata y se pudo maniobrar en tiempo real.
- Mediante las pruebas realizadas se demuestra el control fiable de la plataforma aérea multi-rotor comandando en tiempo real a través de un joystick EXTEME 3D PRO.

4.2 Recomendaciones

- Conocer y averiguar las características de cada elemento de Matlab GUI para realizar una interfaz amigable con el usuario.
- Verificar la comunicación entre las antenas transmisor (Tx) y receptor (Rx) para tener una buena comunicación.
- Conocer la velocidad de transmisión de la antena 3DR enviar los datos de manera repetitiva.
- Medir el ancho de pulso que genera el radio control para generarla en el Arduino.
- El vehículo aéreo no tripulado no alcanza su altura máxima en lugares donde la norma internacional de GPS no lo permita, en lugares aledaños a los aeropuertos su vuelo será restringido.

GLOSARIO DE TERMINOS

- **Matlab: Matrix Laboratory** (Laboratorio de matrices)
- **UAV:** Vehículo aéreo no tripulado
- **PWMs:** Pulse-Width Modulation (Modulación por ancho de pulsos)
- **Multi-rotor:** Vehículos aéreos de 3 o más hélices
- **CW:** Clockwise Use Pusher Propeller (Hélice de empuje de uso en el sentido de las agujas del reloj)
- **CCW:** Counter Clockwise Use Pusher Propeller. (Hélice de empuje en sentido antihorario)

REFERENCIAS BIBLIOGRÁFICAS

- Aeromodelismo*. (28 de Junio de 2016). Obtenido de <http://www.aeromodelismofacil.com/Motores%20brushless.htm>
- AMAZON. (Julio de 2018). Obtenido de AMAZON: <https://www.amazon.com/FLOUREON-5000mAh-Battery-Helicopter-Airplane/dp/B075FPZL2F>
- Aprenfiendo Arduino*. (2017). Obtenido de <https://aprendiendoarduino.wordpress.com/2017/06/19/que-es-arduino-y-hardware-libre-2/>
- Arriagada, I., & Godoy, L. (s.f.). *Seguridad ciudadna y violencia*. Chile.
- Barrientos , A., del Cerro, J., Gutierrez, P., San Martin , R., Martinez, A., & Rossi, C. (2007). *Vehiculos aereos no tripulados para uso civil. Tecnologia y aplicaciones*. Obtenido de <http://kimerius.com/app/download/5783722030/Veh%C3%ADculos+a%C3%A9reos+no+tripulados+par>
- DJI NAZA M LITE. (21 de Abril de 2014). *Naza-M Lite*. Obtenido de http://dl.djicdn.com/downloads/naza-m%20lite/en/NAZA-M%20LITE_User_Manual_v2.00_en.pdf
- Dronecenter*. (Septiembre de 2016). Obtenido de <http://dronecenter.blogspot.com/p/construye-tu-drone.html>
- Fernanda, V. C. (2017). *La seguridad ciudadana y la influencia de la participación ciudadana en las estrategias de prevención del delito*. Quito: Flacso Ecuador.
- FERNANDO, D. (25 de Enero de 2017). *Xataka*. Obtenido de <https://www.xataka.com/especiales/guia-del-arduinomaniaco-todo-lo-que-necesitas-saber-sobre-arduino>
- Fluoreon. (2017). *Amazon*. Obtenido de Amazon: <https://www.amazon.com/FLOUREON-5000mAh-Battery-Helicopter-Airplane/dp/B075FPZL2F>
- Gizmojo*. (2014-2015). Obtenido de <https://www.gizmojo.com.ar/products/dji-naza-m-lite-flight-controller-with-gps-compass>
- Guiadrone*. (2017). Obtenido de <https://guiadrone.com/drones-de-carreras/helices/>
- International, A. f. (04 de Mayo de 2016). *Hemaw*. Obtenido de <https://hemav.com/las-5-principales-aplicaciones-de-los-drones-para-uso-civil/>
- Masterairscrew. (2018). Obtenido de <https://www.masterairscrew.com/products/3dr-solo-propellers-mr-sl-10x4-5-prop-set-x4-black-3dr-solo>
- Matlab. (2018). *Mathworks*. Obtenido de <https://la.mathworks.com/products/matlab.html>
- Modeltronic*. (2015). Obtenido de <http://www.modeltronic.es/chasis-f450-drone-quadcopter-kit-version-pcb-p-9920.html>
- Myrcmart*. (2008-2017). Obtenido de <http://www.myrcmart.com/rcx-2212-920kv-brushless-motor-for-dji-style-prop-cw-p-8029.html>
- Roman Nuñez , E. M., & Ruiz Veintimilla, B. S. (2012). *PLAN DE SEGURIDAD CIUDADANA*. Obtenido de <http://www.dspace.uce.edu.ec/handle/25000/422>
- Taloselectronics. (2018). Obtenido de <https://www.taloselectronics.com/products/controlador-de-velocidad-electronico-esc-30a-multi-axis-para-brushless>
- Turbo Hobby . (26 de Mayo de 2013). *Turbo Hobby Blog*. Obtenido de <http://blogturbohobby.blogspot.com/2013/05/bateriaslipo.html>
- Turnigy*. (s.f.). Obtenido de Turnigy: <http://www.dronestore.mx/Turnigy-Accucel-Balancedor-Cargador-Lipo>
- Veintimilla, E. M. (2012). *PLAN DE SEGURIDAD CIUDADANA* . Quito.

HOJA DE VIDA

DATOS PERSONALES

NOMBRE	Iza Barrionuevo Carla Belén	
DOCUMENTO DE IDENTIDAD	C.C 1724128572	
FECHA DE NACIMIENTO	17 Julio de 1996	
LUGAR DE NACIMIENTO	Quito - Ecuador	
ESTADO CIVIL	Soltera	
DIRECCIÓN	Barrio El Rosal Calle Luis Cordero y Pedro Capiro	
TELÉFONO	0984152065 (móvil)	
E-MAIL	carlitabelen96@gmail.com	

FORMACIÓN ACADÉMICA

Estudios Primarios:	Lucia Franco de Castro
Estudios Secundarios:	Unidad Educativa 24 de Mayo
Estudios Superiores	Universidad de las Fuerzas Armadas ESPE-UGT Carrera: Electrónica Mención Instrumentación y Aviónica. Egresado

Seminarios y otros

Primeras Jornadas Tecnológicas Internacionales en Electromecánica dictado por varios expositores Internacionales; duración 24 horas; Campus ESPE Latacunga; 12 de diciembre del 2016

EXPERIENCIA LABORAL**PASANTIAS COMUNITARIAS IGLESIA LA MERCED**

FUNCIONES: Rehabilitación del sistema eléctrico

JEFE INMEDIATO: Tcnlga.Maritza de Lourdes Nauñay

TELEFONO: 0961061820

Fecha de inicio 29 de agosto del 2016

Fecha de finalizacion 07 de octubre del 2016

PRACTICAS PROFESIONALES GAE #44 Pastasa

JEFE INMEDIATO: SubS.Miller Manuel Chuquimarca

TELEFONO: 0990538402

Fecha de inicio 06 de marzo del 2017

Fecha de finalizacion 31 de marzo del 2017

**PASANTIAS PROFESIONALES Escuela Superior Militar de Aviacion
"Cosme Rennella B"**

JEFE INMEDIATO: Mayor EMT Avc Jose Fred Espin Vaca

TELEFONO: 099274783

Fecha de inicio 28 de agosto del 2017

Fecha de finalizacion 22 de septiembre del 2017

**SERVICIO A LA COMUNIDAD Centro de desarrollo infantil Nueva
Esperanza "El Boliche"**

JEFE INMEDIATO: Lic Jorge Pastor

TELEFONO:099516823

Fecha de inicio 28 de febrero del 2018

Fecha de finalizacion 05 de abril del 2017

ACEPTACIÓN DEL USUARIO

Latacunga, 01 de agosto del 2018

Yo, ING PABLO PILATÁSIG en calidad de encargado del Laboratorio de Instrumentación Virtual de la Unidad de Gestión de Tecnologías, me permito informar lo siguiente:

El proyecto técnico elaborado por el Srta. **IZA BARRIONUEVO CARLA BELEN**, con el tema: **“IMPLEMENTACIÓN DE UNA INTERFAZ TELECONTROL PARA UNA PLATAFORMA AEREA MULTI-ROTOR MEDIANTE MATLAB”**, ha sido efectuado de forma satisfactoria en las dependencias de mi cargo y que la misma cuenta con todas las garantías de funcionamiento, por lo cual extiendo este aval que respalda el trabajo realizado por el mencionado estudiante.

Por tanto, me hago cargo de todas las instalaciones realizadas por el Sr. estudiante.

Atentamente,

ING. PABLO PILATÁSIG
ENCARGADO DEL LABORATORIO DE INSTRUMENTACIÓN VIRTUAL

Latacunga, 01 de agosto del 2018

HOJA DE LEGALIZACIÓN DE FIRMAS

**DEL CONTENIDO DE LA PRESENTE INVESTIGACIÓN SE
RESPONSABILIZA EL AUTOR**

**IZA BARRIONUEVO CARLA BELEN
ID L00033882**

**DIRECTOR DE LA CARRERA DE ELECTRÓNICA MENCIÓN
INSTRUMENTACIÓN & AVIÓNICA**

**Ing. Pablo Xavier Pilatasig Panchi
Director Carrera de Electrónica Mención Instrumentación & Aviónica**