

TIE
V24im
2012

Quadcopter Arduino

IMPLEMENTACIÓN DEL CONTROL DE VUELO DE UN PROTOTIPO TIPO
QUADCOPTER USANDO RADIO FRECUENCIA

ELECTRÓNICA - USTA
TUNJA

ADRIAN MAURICIO VELANDIA RODRÍGUEZ

0 9 4 5

UNIVERSIDAD SANTO TOMÁS
FACULTAD DE INGENIERÍA ELECTRÓNICA

TUNJA

2012

IMPLEMENTACIÓN DEL CONTROL DE VUELO DE UN PROTOTIPO TIPO
QUADCOPTER USANDO RADIO FRECUENCIA

ADRIAN MAURICIO VELANDIA RODRÍGUEZ

Trabajo de grado para optar el título de
INGENIERO ELECTRÓNICO

Director

M.Sc FABIÁN ROLANDO JIMÉNEZ LÓPEZ

UNIVERSIDAD SANTO TOMÁS
FACULTAD DE INGENIERÍA ELECTRÓNICA

TUNJA

2012

*Las ideas expresadas en esta tesis
son responsabilidad exclusiva del autor y
no comprometen a la Universidad Santo Tomás*

NOTA DE ACEPTACIÓN

Observaciones

Firma Decano

Firma Primer Jurado

Firma Segundo Jurado

Firma Tutor

Tunja, Octubre 25 de 2012

DEDICATORIA

A DIOS, por permitirme culminar esta gran meta con éxito, a mi mama María Inés Rodríguez Rojas que no desfalleció en ningún momento y siempre espero que yo culminara mi proceso de educación, ahora le ofrezco este trabajo como símbolo de su logro por que sin ella nunca se habría dado este gran paso en mi vida.

AGRADECIMIENTOS

A mi madre María Inés Rodríguez Rojas, que con su esfuerzo, dedicación y amor me apoyó incondicionalmente y me motivo para superar los obstáculos que se presentaron durante el camino y gracias a ella, hoy tuve la posibilidad de crear un proyecto útil para la sociedad, símbolo de esfuerzo y de constancia.

A mi hermana Nelly Carolina Velandia Rodríguez, que estuvo en toda la evolución del proyecto apoyándome y animándome a continuar con la frente en alto, es un gran ejemplo en mi vida.

A mi novia Camila Andrea Monsalve Moreno, compañera de batallas que siempre confió en mí y me motivo para plantear metas cada vez más altas.

A mi tutor, Fabián Jiménez López, por su guía y colaboración durante el proceso académico.

A mis docentes que con tan solo una teoría simple le daban un gran empujón a este proyecto.

A mis compañeros y amigos que con su apoyo incondicional, sus experiencias y aportes me motivaron para desarrollar cada una de las labores en las diferentes áreas del proyecto.

TABLA DE CONTENIDO

LISTA DE ANEXOS12

INTRODUCCIÓN21

I. PLANTEAMIENTO DEL PROBLEMA24

II. OBJETIVOS.....25

 A. OBJETIVO GENERAL25

 B. OBJETIVOS ESPECÍFICOS.....25

III. METODOLOGÍA26

IV. MARCO TEÓRICO27

 A. ESTADO DEL ARTE.....27

 1. DESARROLLOS MUNDIALES27

 a. MULTICOPTER XIFOS27

 b. AEROQUAD28

 c. ARDUCOPTER28

 d. OPENPILOT29

 e. PARROT AR.DRONE30

 2. DESARROLLOS EN EL ÁMBITO NACIONAL.....30

 a. QUADROTOR UIS GRUPO DE INVESTIGACIÓN ERA30

 b. SCANEAGLE.....31

 c. HERMES 900.....32

 B. MARCO EPISTEMOLÓGICO33

 1. SENSORES.....33

 2. SISTEMA DE TELECOMUNICACIONES.....33

3. CONTROL.....	34
4. MICROCONTROLADOR.....	34
V. QUADCOPTER.....	35
A. DEFINICIÓN.....	35
B. CLASIFICACIÓN DE LOS MULTIROTORES.....	37
C. APLICACIONES DE LOS UAVS Y QUADCOPTER.....	39
D. IMPACTO SOCIAL.....	39
E. MODELO DINÁMICO DEL QUAD-ROTOR.....	39
F. CARACTERÍSTICAS DEL HELICÓPTERO QUAD-ROTOR.....	44
G. CONTROL DE QUADROTORES.....	46
H. FACTORES EXTERNOS QUE INFLUYEN EN LOS UAVS.....	50
1. VIENTO.....	50
2. TEMPERATURA.....	50
3. PESO.....	51
VI. SELECCIÓN DE HARDWARE QUADCOPTER.....	52
A. PROCESADOR.....	53
1. ARDUINO.....	54
a. LIBRERÍAS.....	57
B. SENSORES.....	58
1. L3G4200D.....	60
2. LPY550AL.....	60
3. ITG-3200.....	61
4. LSM303DLH.....	62

C. ACTUADORES	64
1. MOTORES	64
a. MOTORES BRUSHLESS.....	64
b. MOTORES TIPO GLOW	67
c. SERVO MOTOR	68
D. ACCESORIOS NECESARIOS	70
1. VARIADORES ESC.....	70
2. HÉLICES.....	75
a. HÉLICES DE PASO FIJO.....	76
b. HÉLICES DE PASO VARIABLES.....	76
3. ESTRUCTURA	77
4. BATERÍAS.....	79
5. TRANSMISOR DE RADIO FRECUENCIA	81
6. OTROS ACCESORIOS	82
a. CONECTORES	83
b. AMORTIGUACIÓN	84
c. CINTA.....	84
d. TIPO DE CABLE.....	84
e. CARGADOR DE BATERÍAS.	84
VII. DESARROLLO DEL PROYECTO	86
A. COMUNICACIÓN CON SENSORES.....	87
1. ACTUALIZACIÓN MATRIZ DE COSENOS DIRECTOS (DCM)	95
2. NORMALIZACIÓN DE LA MATRIZ	98

3. CORRECCIÓN DE LA DERIVA.....	98
4. CALCULO DE ÁNGULOS DE EULER.....	99
B. RECEPCIÓN DE DATOS DE RADIO CONTROL.....	99
C. MANEJO DE ACTUADORES.....	101
D. ALGORITMO DE CONTROL	103
1. LÓGICA DIFUSA.....	104
a. TEORÍA DE CONJUNTOS DIFUSOS.....	105
b. CONJUNTOS CLÁSICOS.....	105
c. CONJUNTOS DIFUSOS.....	105
d. FUNCIONES DE PERTENENCIA	106
2. CONTROL LÓGICO DIFUSO.....	108
3. CONSTRUCCIÓN DEL PROTOTIPO Y PRUEBAS DE FUNCIONAMIENTO.....	120
VIII. COSTOS TOTALES DEL PROYECTO	147
IX. CONCLUSIONES.....	149
X. DESARROLLOS FUTUROS.....	152
XI. RECOMENDACIONES	153
XII. REFERENCIAS.....	154
DATOS DEL AUTOR.....	161

LISTA DE TABLAS

TABLA 1. CLASIFICACIÓN DE MULTIROTORES SEGÚN NÚMERO DE ROTORES.....	38
TABLA 2. ESTRATEGIA DE CONTROL.....	49
TABLA 3. CARACTERÍSTICAS MICROCONTROLADOR ATMEGA2560.	55
TABLA 4. COMPARACIÓN PARÁMETROS SENSOR GIRÓSCOPO.	62
TABLA 5. CARACTERÍSTICAS IMU LSM303DLH.	63
TABLA 6. CARACTERÍSTICAS ELÉCTRICAS Y FÍSICAS MOTOR, TURNIGY D3548/6 790KV.	69
TABLA 7. CARACTERÍSTICAS ESC DE FABRICANTE HOBBY KING DE LA SERIE SS DE 6000 KV.	73
TABLA 8. CONFIGURACIÓN PREDETERMINADA DEL SBEC 55A MARCA TURNIGY.....	75
TABLA 9. PARÁMETROS FÍSICOS Y ELÉCTRICOS DEL SBEC 55A MARCA TURNIGY.....	75
TABLA 10. CARACTERÍSTICAS BATERÍA TURNIGY 4000MAH 3S 20C LIPO PACK.	81
TABLA 11. DATOS OBTENIDOS EN PRUEBA SENSOR GIRÓSCOPO.	93
TABLA 12. COSTOS TOTALES DEL PROYECTO.	148

LISTA DE IMÁGENES

IMAGEN 1. DAEDALUS MULTICOPTER XIFOS	27
IMAGEN 2. AEROQUAD CYCLONE.	28
IMAGEN 3. MULTIROTOR ARDUOPTER.	28
IMAGEN 4. OPEN PILOT COPTER CONTROL 3D.	29
IMAGEN 5. PARROTAR.DRONE.....	30
IMAGEN 6. QUADROTOR UIS GRUPO DE INVESTIGACIÓN ERA.	31
IMAGEN 7. SCANEAGLE COLOMBIA.....	31
IMAGEN 8. HERMES 900.	32
IMAGEN 9. FOUR-ROTOR ROTORCRAFT (MULTI-ROTOR CONFIGURATION).	35
IMAGEN 10. FOUR-ROTOR ROTORCRAFT. CREDITS – DRAGANFLY INNOVATIONS INC.....	36
IMAGEN 11. THE QUAD-ROTOR ROTORCRAFT OF BOTHEZAT.	37
IMAGEN 12. SENTIDO DE GIRO DE LOS ROTORES.	38
IMAGEN 13. QUAD-ROTOR EN UNA ESTRUCTURA INERCIAL.	41
IMAGEN 14. ESQUEMA DE FUNCIONAMIENTO DEL HELICÓPTERO QUADROTOR.....	45
IMAGEN 15. ESQUEMA DE FUERZAS DEL HELICÓPTERO QUADROTOR.....	46
IMAGEN 16. (A) PITCH, (B) ROLL Y (C)YAW.	47
IMAGEN 17. DIAGRAMA FUNCIONAL ESTRATEGIA DE CONTROL.....	48
IMAGEN 18. DISTRIBUCIÓN DE TEMÁTICAS SELECCIÓN DE DISPOSITIVOS.....	52
IMAGEN 19. SELECCIÓN DE PROCESADOR.	53
IMAGEN 20. ARDUINO MEGA2560.....	54
IMAGEN 21. PROCESO DE SELECCIÓN DE SENSORES.	59
IMAGEN 22. SENSOR L3G4200D.....	60
IMAGEN 23. SENSOR LPY550AL.	61
IMAGEN 24. SENSOR ITG-3200.	61
IMAGEN 25. SENSOR LSM303DLH.....	63
IMAGEN 26. MOTOR BRUSHLESS TURNIGY 2211 INDOOR 2300KV.....	65
IMAGEN 27. MOTOR TIPO GLOW INC .46 GLOW ENGINE WITH MUFFLER.....	67
IMAGEN 28. TURNIGY 380MAX MICRO SERVO (METAL GEAR).	68
IMAGEN 29, BRUSHLESS OUTRUNNER MOTOR, TURNIGY D3548/6 790KV.	69

IMAGEN 30. HOBBYKING SS SERIES 40-50A ESC.....	70
IMAGEN 31. TRES ESC Y UN SBEC ADQUIRIDOS PARA EL PROYECTO.....	73
IMAGEN 32. SBEC 55A MARCA TURNIGY.....	74
IMAGEN 33. SBEC 55A MARCA TURNIGY SIN COBERTURA.....	74
IMAGEN 34. HÉLICES DE PASO FIJO ADQUIRIDAS PARA EL PROYECTO MARCA GWS 10 X 6.....	76
IMAGEN 35. POSICIONES POSIBLES HÉLICE PASO VARIABLE.....	77
IMAGEN 36. ESTRUCTURA DISEÑADA Y CONSTRUIDA PARA EL PROTOTIPO.....	78
IMAGEN 37. ESTRUCTURA TURNIGY TALON QUADCOPTER (2.0).....	78
IMAGEN 38. TURNIGY 4000MAH 3S 20C LIPO PACK.....	80
IMAGEN 39. EQUIPO DE RADIO SPEKTRUM DX8.....	82
IMAGEN 40. POLYMAX 3.5MM GOLD CONNECTORS.....	83
IMAGEN 41. ENSAMBLE DE LOS ESC CON CONECTORES POLYMAX 3.5MM GOLD.....	83
IMAGEN 42. CARGADOR DE BATERÍAS IMAX B6 ADQUIRIDO PARA EL PROYECTO.....	85
IMAGEN 43. FLUJO DEL PROYECTO.....	86
IMAGEN 44. ESQUEMA DE COMUNICACIÓN I2C.....	88
IMAGEN 45. CONEXIÓN ARDUINO CON SENSORES L3G4200D Y LSM303.....	89
IMAGEN 46. FLUJO DEL MANEJO DE SENSORES EN EL PROCESADOR ARDUINO.....	90
IMAGEN 47. DERIVA DE LOS DATOS DEL SENSOR GIRÓSCOPO.....	93
IMAGEN 48. FLUJO DEL ALGORITMO PARA LA OBTENCIÓN DE LOS ÁNGULOS DE EULER.....	95
IMAGEN 49. SISTEMA DE REFERENCIA PROTOTIPO.....	96
IMAGEN 50. RADIO CONTROL SPEKTRUM DX8, ADQUIRIDO PARA EL PROYECTO.....	100
IMAGEN 51. CONEXIÓN FÍSICA DEL RECEPTOR AL MICROPROCESADOR ARDUINO.....	101
IMAGEN 52. PRIMERA PRUEBA LIBRERÍA SERVO1.H CON MOTOR.....	103
IMAGEN 53, CONJUNTOS LÓGICA DIFUSA.....	106
IMAGEN 54. FUNCIÓN DE PERTENENCIA TRIANGULAR.....	107
IMAGEN 55. FUNCIÓN DE PERTENENCIA TRAPEZOIDAL.....	107
IMAGEN 56. ESTRATEGIA DE CONTROL.....	108
IMAGEN 57. CONTROL EN TOOLBOX FUZZY MATLAB®.....	109
IMAGEN 58. CONJUNTOS DE ENTRADA Y SALIDA.....	109
IMAGEN 59. FUNCIONES DE PERTENENCIA <i>PosX</i>	110
IMAGEN 60. FUNCIONES DE PERTENENCIA <i>PosY</i>	111

IMAGEN 61. FUNCIONES DE PERTENENCIA <i>POSZ</i>	112
IMAGEN 62. FUNCIONES DE PERTENENCIA <i>ELEV</i>	112
IMAGEN 63. FUNCIONES DE PERTENENCIA <i>GIRO</i>	113
IMAGEN 64. FUNCIONES DE PERTENENCIA <i>CAB</i>	114
IMAGEN 65. FUNCIONES DE PERTENENCIA <i>ALAB</i>	114
IMAGEN 66. FUNCIONES DE PERTENENCIA <i>MOTAD</i>	115
IMAGEN 67. FLUJO DE CONTROL DEL PROTOTIPO.....	116
IMAGEN 68. PERFILES DE ALUMINIO ESTRUCTURA QUADCOPTER.....	120
IMAGEN 69. EMPALME DE PERFILES DE ALUMINIO.....	120
IMAGEN 70. ARMADO DE LOS MOTORES.....	121
IMAGEN 71. LÁMINA DE ACRÍLICO SUJETA A LA ESTRUCTURA.....	121
IMAGEN 72. SUJECIÓN DE LOS MOTORES.....	122
IMAGEN 73, ESTRUCTURA CON MOTORES.....	122
IMAGEN 74. ESTRUCTURA CON DOBLE LÁMINA DE ACRÍLICO.....	123
IMAGEN 75. UBICACIÓN DE DISPOSITIVOS EN EL PROTOTIPO.....	123
IMAGEN 76. CONECTORES DE REGLETA 40A.....	124
IMAGEN 77, ENSAMBLE DE CIRCUITERÍA.....	124
IMAGEN 78. PRIMERA PRUEBA EJE X.....	125
IMAGEN 79. ENSAMBLE DE LAS PLATINAS DE ALUMINIO.....	126
IMAGEN 80. UBICACIÓN SENSORES CON AMORTIGUACIÓN.....	126
IMAGEN 81. ENSAMBLE DE DISPOSITIVOS EN EL PROTOTIPO.....	127
IMAGEN 82. DETALLE DE LA UBICACIÓN DE LOS SENSORES.....	127
IMAGEN 83. LECTURA DE DATOS POSICIÓN 1.....	128
IMAGEN 84. PRUEBA AISLANDO LOS SENSORES.....	129
IMAGEN 85. PCB SENSORES Y POTENCIA.....	130
IMAGEN 86. PCB IMPLEMENTADO.....	130
IMAGEN 87. PCB DE POTENCIA PUESTO EN EL PROTOTIPO.....	131
IMAGEN 88. UBICACIÓN DEL MICROPROCESADOR EN EL PROTOTIPO.....	131
IMAGEN 89. DETALLE DE LA UBICACIÓN DEL PCB EN EL PROTOTIPO.....	132
IMAGEN 90. ENSAMBLE DE CAJA DE BALSO PROTOTIPO.....	132
IMAGEN 91. DETALLE CAJA PARA SENSOR Y PROCESADOR.....	133

IMAGEN 92. DISEÑO Y UBICACIÓN DEL SENSOR CON ESPUMA.	133
IMAGEN 93. PRUEBA CON ESPUMA.	134
IMAGEN 94. PRIMERA PRUEBA ALGODÓN.	134
IMAGEN 95, SEGUNDA PRUEBA ALGODÓN.	135
IMAGEN 96. SENSOR AMORTIGUADO POR GUATA.	135
IMAGEN 97. PRUEBA CON GUATA.....	136
IMAGEN 98. PRUEBA CON GUATA, ELEVACIÓN.	136
IMAGEN 99. DESPIECE DE LA ESTRUCTURA DE FIBRA DE CARBONO.....	137
IMAGEN 100. DETALLE ENSAMBLE DEL MOTOR EN UN EJE DE LA ESTRUCTURA DE FIBRA DE CARBONO.....	137
IMAGEN 101. DETALLE ESTRUCTURA ARMADA CON PCB DE POTENCIA.	138
IMAGEN 102. PROTOTIPO ESTRUCTURA FIBRA DE CARBONO PERFIL SUPERIOR.....	138
IMAGEN 103. COMPARACIÓN ESTRUCTURAS QUADCOPTER.	139
IMAGEN 104. UBICACIÓN DISPOSITIVOS ESTRUCTURA FIBRA DE CARBONO.	139
IMAGEN 105. DETALLE MOTOR ESTRUCTURA FIBRA DE CARBONO.	140
IMAGEN 106. PERFIL SUPERIOR ESTRUCTURA DE FIBRA DE CARBONO LISTA.	140
IMAGEN 107. PERFIL INFERIOR ESTRUCTURA DE FIBRA DE CARBONO LISTA.	141
IMAGEN 108. UBICACIÓN DE LOS SENSORES EN LA ESTRUCTURA DE FIBRA DE CARBONO.	141
IMAGEN 109. PRUEBA ESTRUCTURA DE FIBRA DE CARBONO.	142
IMAGEN 110. PRUEBA CONFIGURANDO SENSIBILIDAD A 250DPS.	142
IMAGEN 111. PRUEBA CONFIGURANDO SENSIBILIDAD A 2000DPS.	143
IMAGEN 112. BALANCEADOR MAGNÉTICO ADQUIRIDO PARA EL PROYECTO.	144
IMAGEN 113. PRUEBA LUEGO DEL BALANCEO Y PROMEDIO CON 10 DATOS.	145
IMAGEN 114. PRUEBA LUEGO DEL BALANCEO Y PROMEDIO CON 5 DATOS EJE X.....	145
IMAGEN 115. PRUEBA LUEGO DEL BALANCEO Y PROMEDIO CON 5 DATOS EJE Z.....	146

LISTA DE ANEXOS

ANEXO1	SENSORES	1
ANEXO2	VARIADORES ESC	4
ANEXO3	RADIO CONTROL.....	4
ANEXO4	FOTOS Y VIDEOS	4

SIP (Session Initiation Protocol): Es un protocolo desarrollado por el grupo de trabajo MMUSIC del IETF con la intención de ser el estándar para la iniciación, modificación y finalización de sesiones interactivas de usuario donde intervienen elementos multimedia como el video, voz, mensajería instantánea, juegos en línea y realidad virtual.

Wake Up: Sistemas de activación automática disponible en algunos dispositivos electrónicos.

UART (Universal Asynchronous Receiver-Transmitter): Controla los puertos y dispositivos serie. Se encuentra integrado en la placa base o en la tarjeta adaptadora del dispositivo.

PWM (Pulse-Width Modulation): De una señal o fuente de energía es una técnica en la que se modifica el ciclo de trabajo de una señal periódica (seno y cuadrada), ya sea para transmitir información a través de un canal de comunicaciones o para controlar la cantidad de energía que se envía a una carga.

USM (Universal Serial Bus): Es un estándar industrial desarrollado en el año 1990, que define los cables, conectores y protocolos usados en un bus para conectar, comunicar y proveer de alimentación eléctrica entre ordenadores, periféricos y dispositivos electrónicos

ICSP (In Circuit Serial Programming): Es el método de programación directa usada generalmente en AVR, Parallax Propeller y microcontroladores PIC.

Shields: Son placas que pueden ser conectadas encima de la placa Arduino extendiendo sus capacidades. Las shields son fáciles de ubicar y baratas de producir.

Android: Es un sistema operativo móvil basado en Linux, que junto con aplicaciones middleware, está enfocado para ser utilizado en dispositivos móviles como teléfonos inteligentes, tabletas, Google TV y otros dispositivos.

Middleware: Es un software que asiste a una aplicación para interactuar o comunicarse con otras

GLOSARIO

UAV: Unmanned Aerial Vehicle (vehículo aéreo no tripulado).

Dinámica: La dinámica es la parte de la física que describe la evolución en el tiempo de un sistema físico en relación a las causas que provocan los cambios de estado físico y/o estado de movimiento. El objetivo de la dinámica es describir los factores capaces de producir alteraciones de un sistema físico, cuantificarlos y plantear ecuaciones de movimiento para dicho sistema.

Pitch (Cabeceo): Inclinación del morro del avión, o rotación respecto al eje ala-ala.

Roll (Alabeo): Rotación intrínseca alrededor del eje longitudinal del avión.

Yaw (Guiñada): movimiento del avión respecto del eje imaginario vertical que pasa por el centro de gravedad de la aeronave. Este eje es perpendicular al eje de cabeceo y al de balanceo, está contenido en un plano que pasa por el morro y la cola del aparato y que normalmente divide a este en dos partes simétricas.

RTD (Resistance Temperature Detector): Los termómetros de resistencia o termómetros a resistencia son transductores de temperatura, los cuales se basan en la dependencia de la resistencia eléctrica de un material con la temperatura, es decir, son capaces de transformar una variación de temperatura en una variación de resistencia eléctrica.

Termopar: Un termopar es un dispositivo para la medición de temperatura, basado en efectos termoeléctricos. Es un circuito formado por dos conductores de metales diferentes o aleaciones de metales diferentes, unidos en sus extremos y entre cuyas uniones existe una diferencia de temperatura, que origina una fuerza electromotriz efecto Seebeck.

MEMS (Microelectromechanical Systems): Se refieren a la tecnología electromecánica, micrométrica y sus productos, y a escalas relativamente más pequeñas (escala nanométrica) se fusionan en sistemas nano-electromecánicos (Nanoelectromechanical Systems, NEMS) y Nanotecnología.

aplicaciones, software, redes, hardware y/o sistemas operativos.

iOS: Es un sistema operativo móvil de Apple. Originalmente desarrollado para el iPhone, siendo después usado en dispositivos como el iPod Touch, iPad y el Apple TV

WI-FI (*Wireless Fidelity*): Es un mecanismo de conexión de dispositivos electrónicos de forma inalámbrica. Los dispositivos habilitados con Wi-Fi, tales como: un ordenador personal, una consola de videojuegos, un smartphone o un reproductor de audio digital, pueden conectarse a Internet a través de un punto de acceso de red inalámbrica.

Drone: Término usado para denominar a los vehículos aéreos no tripulados, generalmente de uso militar.

AVR: Los AVR son una familia de microcontroladores RISC del fabricante estadounidense Atmel.

Hobby: Es una actividad cuyo valor reside en el entretenimiento de aquél que lo ejecuta, que algunas veces no busca una finalidad productiva concreta y se realiza en forma habitual.

Inrunner: El término inrunner se refiere a un tipo de motor sin escobillas usado en los modelos de radio control. Estos motores son construidos con los imanes en la parte interior, que es la que gira y las bobinas en el exterior.

Outrunner: El término outrunner se refiere a un motor brushless del tipo utilizado principalmente en aviones modelo propulsado eléctricamente, radio-controlado. Este tipo de motor gira su carcasa externa alrededor de sus bobinas.

RESUMEN

El presente documento explora los parámetros de implementación de un prototipo de máquina voladora tipo quadcopter, que utiliza un controlador lógico difuso para su estabilización. Un quadcopter es un helicóptero de cuatro rotores, el cual tiene una mayor estabilidad debido a su arquitectura en forma de cruz y los cuatro rotores situados en sus extremos. Para la estabilización del prototipo es usado un controlador lógico difuso el cual actúa de manera efectiva; mediante la selección de una acción de control establecida por los parámetros de entrada definidos por la lectura de sensores, la cual se reproduce en forma de la potencia suministrada a los cuatro motores con el fin de mantener estable el vuelo del prototipo.

El presente documento esta organizado de la siguiente manera: en la sección IV se realiza un marco teórico el cual indica al lector los presentes desarrollos a nivel mundial que motivaron la implementación del proyecto, así como un marco epistemológico básico que forma la base del desarrollo del proyecto. En la sección V, se encuentra el desarrollo teórico del quadcopter enumerando todos los parámetros que influyen en este tipo de dispositivos. Seguidamente en la sección VI se desarrolla el proceso de selección de hardware contemplando las opciones de microprocesador, sensores y actuadores hasta los dispositivos que son indispensables para el prototipo pero no son objeto de estudio de este proyecto. En la sección VII se presenta el desarrollo del proyecto dividido por etapas donde se realiza la captura y procesamiento de la información proveniente de los sensores, se muestra el proceso de recepción de datos del radio control. Posteriormente se explica de manera detallada el manejo de los actuadores junto con el algoritmo de programación desarrollado para el proyecto y finalmente el proceso de construcción del prototipo con pruebas y evidencias del funcionamiento del algoritmo. En la sección IX son plasmadas las conclusiones obtenidas a partir del desarrollo de este proyecto.

INTRODUCCIÓN

Los prototipos de quadrotores son una de las plataformas con más auge de desarrollo en el área de robótica, gracias a su funcionalidad, capacidad de vuelo estacionario, despegue y aterrizaje vertical y según la efectividad del controlador, mediante una interfaz de control relativamente sencilla. Es comúnmente utilizado para tareas de búsqueda y rescate, sistemas de vigilancia, exploración en áreas de difícil acceso, entre otras; la investigación en este campo ha sido enfocada principalmente al desarrollo de un vehículo totalmente autónomo que proporcione maniobrabilidad y una excelente respuesta ante perturbaciones generadas por factores externos al prototipo.

El quadcopter presentado en este trabajo está formado por un sistema realimentado implementado con un sistema de control que verifica las señales de entrada y ejerce el control necesario en los actuadores.

Para la estabilización se utilizaron tres sensores tridimensionales, un giróscopo, un acelerómetro y un magnetómetro, que enlazados permiten tener el pleno conocimiento de sus seis grados de libertad, tres que definen la orientación del prototipo: alabeo, cabeceo y guiñada o giro y tres que definen la posición del prototipo: altura, desplazamiento horizontal y desplazamiento vertical.

La implementación se realizó iniciando desde cero, eligiendo cuidadosamente los dispositivos sensores, actuadores y demás que conforman el prototipo. El controlador implementado usó lógica difusa para tomar decisiones, es un controlador que ha sido aplicado en las áreas de la industria, medicina, manufactura entre otras.

Este prototipo es el primer desarrollo de esta clase en la Universidad Santo Tomás Seccional Tunja y es implementado con el propósito de motivar a la comunidad académica para que se realice una proyección de metas cada vez más altas con desarrollos a la vanguardia de la tecnología actual.

JUSTIFICACIÓN

La manufactura de aparatos mecánicos capaces de elevarse en vuelo es uno de los campos de investigación y desarrollo más atractivos en la ingeniería, ya que integra una gran variedad de disciplinas y conocimientos basados en sistemas de control aplicados en diferentes áreas, diseñados con la capacidad de adaptarse a entornos controlados o cambiantes siempre con la tendencia de reducir costos y aumentar rendimiento.

Los vehículos no tripulados ofrecen gran variedad de aplicaciones las cuales estimulan más la investigación en este campo permitiendo acoplar diversos dispositivos para la optimización de un prototipo, aumentando así el desarrollo intelectual y el nivel investigativo de la comunidad académica vinculando ingeniería, física y aerodinámica, con esto es posible obtener de una manera teórico-práctica un modelo de gran eficiencia como punto de partida para diversas modificaciones e implementaciones dependiendo de la aplicación.

El rendimiento de dispositivos de tipo quadcopter en ambientes controlados es objeto de estudio gracias al acople de diferentes áreas del conocimiento, por eso se ha querido introducir el área aeronáutica en el desarrollo de este proyecto de tesis demostrando de una manera práctica la aplicabilidad e incentivando a estudiantes de la facultad a introducirse en este nuevo campo de desarrollo con gran diversidad de conocimiento.

En los sistemas de monitoreo aéreo se combina instrumentación, control y telecomunicaciones, en donde, para este caso de estudio, se hizo énfasis en el desarrollo de la instrumentación y control, a partir de estas tecnologías se crean dispositivos con diferentes fines pero fundamentados en los procesos de control y automatización incluyendo un alto grado de complejidad de acuerdo a las aplicaciones del mismo. El desarrollo del proyecto se realizó teniendo en cuenta variables medidas para así ejercer una efectiva acción de control partiendo del manejo de la sensórica, su respectivo procesamiento de datos y el control del sistema ejercido sobre los actuadores para garantizar la calidad de los procesos.

Ahora el desarrollo de sistemas tipo quadcopter busca el mejoramiento y la efectividad en diversas tareas de monitoreo gracias a su versatilidad y la cómoda manipulación por radio frecuencia.

El sistema Quadcopter requiere de un operador el cual administra los controles de vuelo

permitiendo ejercer un control directo sobre el dispositivo reduciendo riesgos y costos de desplazamiento en ambientes controlados.

Desde el punto de vista económico las ventajas son apreciables ofreciendo la minimización de recursos invertidos en el personal dedicado a ejercer las tareas de monitoreo y reduciendo riesgos de accidentalidad donde las cuantías por concepto de indemnizaciones y demás gastos de riesgos profesionales son muy elevadas.

I. PLANTEAMIENTO DEL PROBLEMA

- ¿Es posible controlar un prototipo de vehículo volador no tripulado a partir de un sistema electrónico?
- ¿Es viable desarrollar un prototipo de vehículo no tripulado basado en ingeniería de integración?

El desplazamiento es una tarea necesaria para el personal de seguridad de un lugar y es realizado con el fin de monitorear diferentes áreas donde existe la posibilidad de presencia de intrusos, con el riesgo que en este desplazamiento ocurra un accidente o con la posibilidad de un ataque cuerpo a cuerpo en situaciones específicas, este sistema es planteado para minimizar riesgos y aumentar el tiempo crítico de reacción del personal. Se propone diseñar el control electrónico de un prototipo Vehículo Aéreo no Tripulado UAV (Unmanned Aerial Vehicle) a escala propulsado por cuatro motores suministrando la energía suficiente para mantenerse en vuelo, controlados por un sistema embebido con la capacidad de tomar decisiones de acuerdo a las características de estabilidad del prototipo adquiridas por medio de sensores capaces de identificar y transmitir las señales de las variables medidas con el objeto de su procesamiento.

El prototipo se desarrolló para ser controlado remotamente por personal capacitado para su manejo por medio de Radio Frecuencia y capaz de desplazarse por espacios cerrados ofreciendo la posibilidad de transportar un sistema de transmisión de video en tiempo real y capacidad de desplazamiento de manera vertical, para ser implementado en un tiempo no mayor a seis meses en las instalaciones de la Universidad Santo Tomás haciendo uso de las distintas herramientas ofrecidas por la Facultad de Ingeniería Electrónica y contando con la accesoria de los docentes con conocimientos de los temas de desarrollo.

II. OBJETIVOS

A. OBJETIVO GENERAL

Implementar el control de vuelo de un prototipo tipo quadcopter UAV, (Unmanned Aerial Vehicle) a escala adaptable para ser utilizado en monitoreo de ambientes controlados mediante el uso de tecnologías de instrumentación y control.

B. OBJETIVOS ESPECÍFICOS

- Revisar y seleccionar información documental soporte para el desarrollo del proyecto.
- Seleccionar dispositivos de instrumentación para el sistema.
- Definir las variables a monitorear para el sistema de control de vuelo.
- Adquirir e implementar un sistema mecánico y de adquisición de datos de los sensores.
- Proponer un sistema de control que permita el tratamiento de los datos de las variables a manipular.
- Diseñar un código de programación integrando las funciones de control y la adquisición de datos de los sensores.
- Identificar los datos enviados por el mando de radio frecuencia.
- Realizar el control requerido para los motores y realizar un monitoreo de los sensores para así controlar el vuelo del helimodelo.
- Implementar una estructura móvil que se acondicione a los requerimientos de vuelo del prototipo.
- Socializar el proceso y resultados obtenidos a partir del proyecto a la comunidad académica.

III. METODOLOGÍA

Este proyecto se desarrolló por medio de investigación experimental ya que parte de lo general, de la teoría ya conocida sobre las temáticas y por medio de diseño y prácticas se llega al conocimiento específico objeto del proyecto. Basado en los principios básicos de funcionamiento de los diferentes componentes de helimodelo además se hizo una implementación gradual del prototipo con registros del trabajo realizado.

IV. MARCO TEÓRICO

A. ESTADO DEL ARTE.

A continuación se describen los prototipos que han sido objeto de desarrollo a nivel mundial seleccionados por su características de hardware y software similares al prototipo objeto de este proyecto.

1. DESARROLLOS MUNDIALES

a. MULTICOPTER XIFOS



Imagen 1. Daedalus Multicopter XIFOS
Tomado de www.daedalusmulticopter.com

En la Imagen 1. Se muestra la foto del prototipo quadrotor XIFOS diseñado por la compañía Daedalus Multicopter. XIFOS es el avión insignia de este fabricante diseñado específicamente para ser una plataforma de cámara o inspección. Implementado con fibra de carbono grado aeroespacial para innovar en aerodinámica, fuerza y estabilidad¹.

¹ <http://www.daedalusmulticopter.com/>

b. AEROQUAD



Imagen 2. AeroQuad Cyclone.

Tomado de www.aeroquad.com

AeroQuad es un proyecto de hardware y software de código abierto, está dedicado a la construcción de multirrotores controlados remotamente. AeroQuad es normalmente la unión de un control de vuelo AeroQuad o un microcontrolador Arduino (Mega 2560 o Uno) además un shield armado con varios sensores como acelerómetro, giróscopo entre otros².

c. ARDUICOPTER



Imagen 3. Multirrotor ArduCopter.

Tomado de www.arducopter.com

ArduCopter es un UAV multicopter de código abierto. Basado en piloto automático de ArduPilot

² <http://aeroquad.com/content.php>

Mega 2 creado por la comunidad de bricolaje Drones³, soporta cuatro a ocho motores, así como helicópteros tradicionales y permite misiones totalmente autónomas, así como controlados por radio frecuencia, incluyendo puntos de referencia, planificación de misiones y telemetría manteniendo enlace con una estación en tierra de gran alcance⁴.

ArduCopter está a la vanguardia de la robótica aérea y esta destinado a aquellas personas que quieren probar la tecnología avanzada y nuevos estilos de vuelo.

d. OPENPILOT

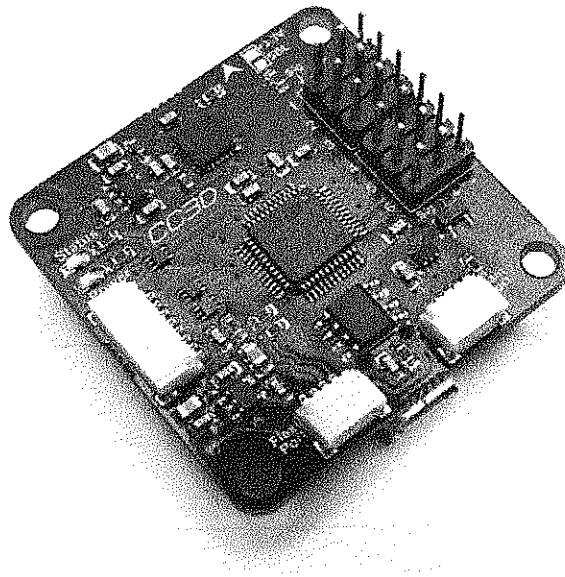


Imagen 4. Open Pilot Copter Control 3D.

Tomado de www.openpilot.org

OpenPilot es un piloto de UAV de código abierto de última generación, es una plataforma capaz de controlar todo tipo de helicópteros multirotores, así como aeronaves con hélices de paso fijo.

³ <http://diydrones.com/profiles/blogs/ardupilot-mega-home-page>

⁴ <http://code.google.com/p/arducopter/wiki/ArduCopter?tm=6>

Ha sido construido por una comunidad de desarrolladores apasionados de todo el mundo, sus principios de diseños se basan en calidad y facilidad de uso⁵.

e. PARROT AR.DRONE

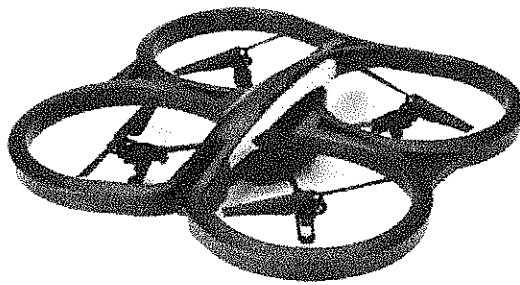


Imagen 5. ParrotAR.Drone.

Tomado de www.ardrone2.org

ParrotAR.Drone es un cuadricóptero controlado remotamente, cuenta con una serie de sensores, incluyendo una cámara frontal de alta definición, cámara vertical, un altímetro de ultrasonido y el modo de control absoluto. Con la AR.Drone 2.0, puede ahora grabar video y tomar fotos mientras en vuelo y compartirlos directamente en Youtube y Picasa. Construido por LORO SA, diseñado para ser controlado con dispositivos Android o iOS. Permite compartir escenarios de vuelo con otros dispositivos mediante conexión WIFI⁶.

2. DESARROLLOS EN EL ÁMBITO NACIONAL

a. QUADROTOR UIS GRUPO DE INVESTIGACIÓN ERA

⁵ <http://www.openpilot.org/>

⁶ <http://ardrone2.parrot.com/gallery/drone/>

Prototipo de Robot quadrotor (robot aéreo de cuatro aspas con control autónomo primera fase) desarrollado por el grupo de investigación ERA (Electrónica y Robótica Aplicada) de la Universidad Industrial de Santander, este robot ganó uno de los dos reconocimientos por robot destacado en Exporobot 2010, Campus Party.

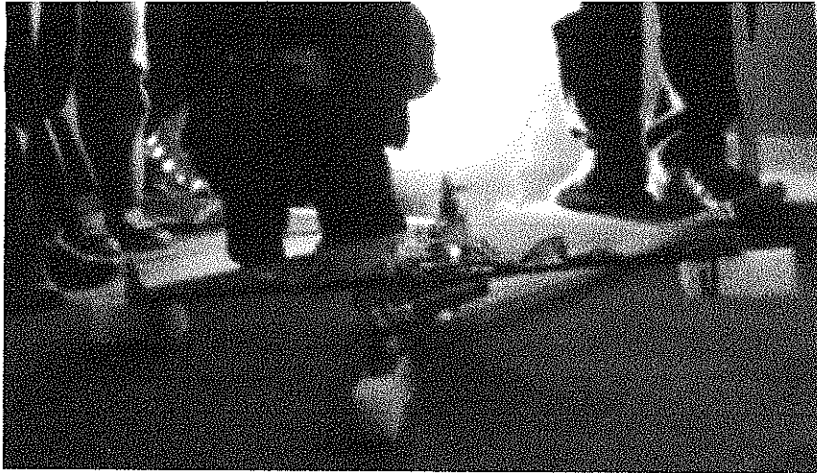


Imagen 6. Quadrotor UIS grupo de investigación ERA.

Tomado de www.dynamoelectronics.com

b. SCANEAGLE



Imagen 7. ScanEagle Colombia.

Tomado de www.webinfomil.com

En Colombia los UAV han tenido una lenta incorporación en las fuerzas armadas, los primeros pasos en esta materia fueron dados por la Fuerza Aérea Colombiana, cuando en el segundo semestre de 2005 adquirió un mini helicóptero no tripulado llamado AutoCopter VTOL UAV, fabricado por la compañía Neutral Robotics INC el cual fue comprado en EEUU. El aparato mide 70 cms. Tiene una autonomía de vuelo de una hora, posee cámaras de video y equipos de cartografía.

En el 2006 la Fuerza Aérea de Estados Unidos dono a la Fuerza Aérea Colombiana un lote de Aeronaves No Tripuladas de tipo táctico ScanEagle fabricadas por las compañías InSitu y Boeing.

El ScanEagle es una aeronave no tripuladas de 40 lbs diseñada para misiones continuas de más de 15 horas, el cual tiene una velocidad de crucero de 50 nudos a una altitud de hasta 5000 pies. El sistema fue diseñado para operaciones de campo autónomas, lleva una carga útil de 6 kilogramos y puede ser lanzado o recuperado en cualquier terreno incluyendo operaciones naval⁷.

c. HERMES 900



Imagen 8. Hermes 900.

Fuente: Tomado de www.webinfomil.com/

⁷ <http://www.webinfomil.com/2011/12/aeronaves-no-tripuladas-en-colombia-el.html>

El Hermes 900 es un desarrollo Israelí incluido dentro de un contrato de adquisición de una flota mixta de vehículos aéreos no tripulados (UAVs) que ganó la compañía Israelí Elbit Systems para suministrar a la Fuerza Aérea Colombiana.

El Hermes 900 es una plataforma moderna con una capacidad de portar variados equipos y sensores electro-ópticos, configurados en una arquitectura modular que permite el rápido cambio de equipos, dependiendo de la misión a realizar. Su capacidad de carga es de unos 400 kilos y entre los equipos se puede incluir, sensores de imagen IR (Infrarrojo), sensores de fijación de objetos por laser, sistemas SAR(Synthetic Aperture Radars) / GMTI (Ground Moving Target Indicator), sistemas Comint DF (Communications Intelligence Direction Finding) y ELINT (ELectronic INTelligence), además cuenta uno de los sistemas más avanzados de comunicación y encriptación de datos⁸.

B. MARCO EPISTEMOLÓGICO

1. SENSORES.

Un sensor es un dispositivo capaz de detectar magnitudes físicas o químicas, llamadas variables y transformarlas en variables eléctricas. Las variables que identifican pueden ser: temperatura, intensidad luminica, distancia, aceleración, inclinación, desplazamiento, presión, fuerza, torsión, humedad, pH, etc. Una magnitud eléctrica puede ser una resistencia eléctrica como en una RTD (Resistance Temperature Detector), una capacitancia eléctrica (como en un sensor de humedad), una Tensión eléctrica (como en un termopar), una corriente eléctrica (como en un fototransistor), etc.

2. SISTEMA DE TELECOMUNICACIONES.

Es una técnica consistente en transmitir un mensaje desde un punto a otro, normalmente con el atributo típico adicional de ser bidireccional. El término telecomunicación cubre todas las formas de comunicación a distancia, incluyendo radio, telegrafía, televisión, telefonía, transmisión de datos e interconexión de computadoras a nivel de enlace. Telecomunicaciones, es toda transmisión, emisión o recepción de signos, señales, datos, imágenes, voz, sonidos o información de cualquier naturaleza que se efectúa a través de cables, medios ópticos, físicos u otros sistemas electromagnéticos. Los protocolos de comunicación utilizados en el proyecto son explicados en la sección Desarrollo del Proyecto.

⁸ <http://www.webinfomil.com/2012/08/colombia-adquiere-aeronaves-no.html>

3. CONTROL.

La estrategia fundamental es utilizar el control de procesos para extraer el máximo desempeño de los equipos que lo gobiernan, basando la supervisión en el monitoreo de variables físicas como temperatura, corrientes eléctricas, voltajes de fase, vibración, posición, velocidades etc. con las cuales se detectan oportunamente anomalías en el funcionamiento de un proceso, máquina o equipo y así tomar decisiones automáticas orientadas a evitar la parálisis de la producción y reducir los riesgos que implica la toma de decisiones afectadas por la subjetividad humana. Los sistemas de control utilizados están detallados en la sección Desarrollo del Proyecto.

4. MICROCONTROLADOR.

Un microcontrolador es un circuito integrado o chip que incluye en su interior las tres unidades funcionales de una computadora: CPU (Central Processing Unit), Memoria y Unidades de E/S, es decir, se trata de un computador completo en un circuito integrado.

V. QUADCOPTER

A. DEFINICIÓN.

Un quadcopter también llamado, cuadrocóptero o cuadricóptero, es un multicopter que es elevado y propulsado por cuatro rotores, el término drone, es auto descriptivo para usos civiles y militares de las últimas generaciones de aviones sin piloto.

El helicóptero es una de las máquinas más complejas. Esta complejidad se debe a su versatilidad y capacidad de maniobra para realizar muchos tipos de tareas. El helicóptero clásico es comúnmente equipado con un rotor principal y un rotor de cola. Entre otros tipos de helicópteros se encuentran los de rotor gemelo o helicóptero tándem y el helicóptero de rotor coaxial.

El énfasis de este proyecto fue desarrollar un mini-helicóptero que tiene cuatro rotores más conocido como helicóptero quadrotor o quadcopter.

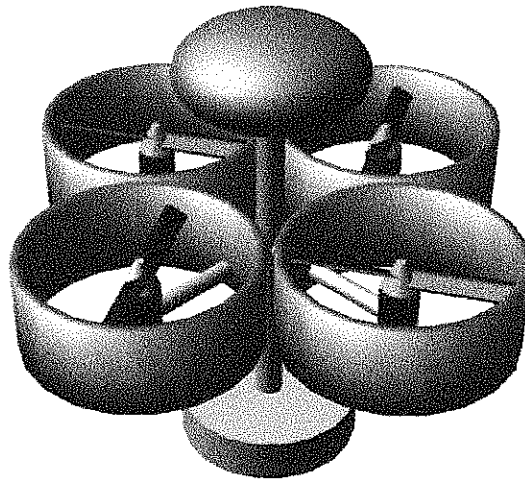


Imagen 9. Four-rotor rotorcraft (multi-rotor configuration).

Fuente: *Modelling and Control of Mini-Flying Machines*, Pedro Castillo, Rogelio Lozano and Alejandro E. Dzul. France 2004.

Hay varias ventajas de los quadcopter sobre los helicópteros a escala comparables. Es, en primer lugar, los quadcopter no requieren acoplamientos mecánicos para variar la inclinación del modelo. Esto simplifica el diseño y mantenimiento del vehículo. En segundo lugar, el uso de

cuatro rotores permite a cada rotor individual tener un diámetro más pequeño que el rotor del helicóptero equivalente, lo que les permite requerir menos energía cinética durante el vuelo, esto reduce la potencia consumida. Los UAVs en pequeña escala, son vehículos más seguros para una interacción estrecha. Algunos pequeños quadcopter tienen marcos que encierran los rotores, permitiendo vuelos a través de los entornos exigentes, con menor riesgo de dañar el vehículo o sus alrededores.

Debido a su facilidad de construcción y control, los quadcopter son aeronaves utilizadas frecuentemente como proyectos de aviones de modelo amateur y proyectos más complejos de investigación como el mostrado en este documento.



Imagen 10. Four-rotor rotorcraft. Credits – Draganfly Innovations Inc.

Fuente: Modelling and Control of Mini-Flying Machines, Pedro Castillo, Rogelio Lozano and Alejandro E. Dzul. France 2004.

Para ejercer control sobre un quadcopter se realiza alterando la velocidad de rotación de los motores modificando así su par y empuje característicos.

Los quadcopter son una evolución simplificada de los helicópteros a escala eliminando el problema siempre presente en el vuelo vertical de la eficiencia del rotor de cola el cual limitaba la utilidad de un ascenso vertical. El estudio realizado dio como resultado la eliminación de estas fuerzas implementando la contra rotación de las hélices para así contrarrestar con la misma fuerza de empuje el movimiento inercial de cada rotor.

En la década de 1920 aparecieron los vehículos VTOL (*Vertical Take-Off and Landing*), despegue y aterrizaje verticales) siendo los primeros tuvieron un gran éxito sin embargo estos primeros prototipos tenían un desempeño pobre y un control de vuelo limitado.

En enero de 1921, la US Army Air Corps otorgó un contrato a Dr. George de Bothezat e Ivan Jerome para desarrollar una máquina de vuelo vertical. La estructura en forma de X de 1678 kg apoyado un rotor de seis palas de diámetro de 8,1 m en cada extremo de los brazos de 9 m (ver Imagen 11). En los extremos de los brazos laterales, dos pequeñas hélices con paso variable fueron usados para control de empuje y guiñada. Un rotor de elevación también fue montado sobre el motor radial Le Rhone de 180 HP en el cruce de los marcos, pero más tarde fue retirado como innecesario⁹.

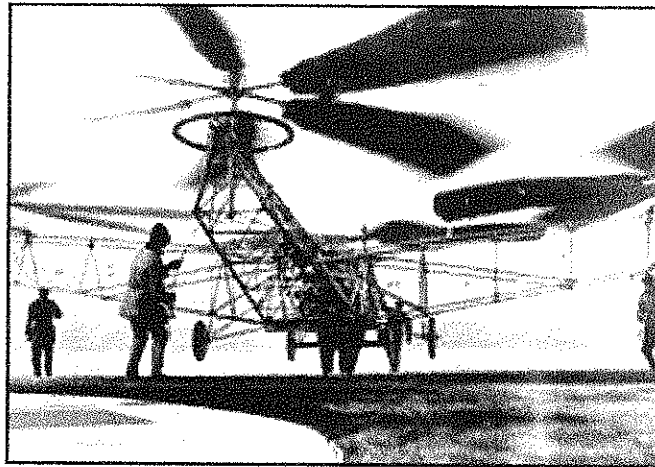


Imagen 11. The quad-rotor rotorcraft of Bothezat.

Fuente: Modelling and Control of Mini-Flying Machines, Pedro Castillo, Rogelio Lozano and Alejandro E. Dzul. France 2004.

Posteriormente se polarizaron los vehículos (UAV), estos vehículos utilizan un sistema de control electrónico y sensores electrónicos para estabilizar la aeronave, poseen un tamaño pequeño y ágil maniobrabilidad ofreciendo la posibilidad de volar en interiores y exteriores.

B. CLASIFICACIÓN DE LOS MULTIROTORES.

Los quadcopter son clasificados como helicópteros a diferencia de los aviones por que permiten tener una elevación vertical, para mantenerse en vuelo generalmente tienen palas (hélices) de paso fijo y son simétricamente agudas comportándose como un grupo calificando como “cíclica”, existen varios tipos de multirotores clasificándose de acuerdo al número de rotores que usan para

⁹ The Quad-rotor Rotorcraft, Modelling and Control of Mini-Flying Machines, Pedro Castillo, Rogelio Lozano and Alejandro E. Dzul. France 2004.

mantenerse en vuelo como se muestra en Tabla 1.

NOMBRE	No DE ROTORES
Tricopter	3
Quadcopter	4
Pentacopter	5
Hexacopter	6
Octocopter	8
Decacopter	10
Dodecacopter	12

Tabla 1. Clasificación de Multirotores Según Número de Rotores.

Fuente: Autor.

Generalmente el quadcopter es construido realizando una intercepción de perfiles de una longitud similar, ensamblados en un punto medio común para así dar la simetría característica de estas maquinas voladoras.

Cada motor ejerce una fuerza de empuje sobre su eje de rotación y según la disposición de las hélices ejerce una fuerza de arrastre opuesta a la dirección del vehículo, los rotores giran en sentido opuesto como se muestra en la Imagen 12.

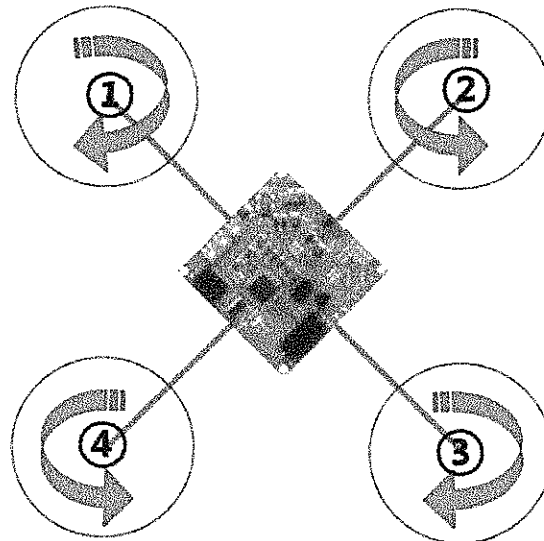


Imagen 12. Sentido de giro de los rotores.

Fuente: Tomado de www.aeromodelismovirtual.com

Si todos los rotores giran de acuerdo a la Imagen 12. Rotor 1 y rotor 3 en un sentido, rotor 2 y rotor 3, en el sentido contrario y con la misma potencia las fuerzas de giro de los motores se contrarrestan ejerciendo así un empuje en el eje Z.

C. APLICACIONES DE LOS UAVS Y QUADCOPTER.

Se utilizan para realizar inteligencia, vigilancia y misiones de reconocimiento. La promesa tecnológica de los UAVs es servir a través de toda la gama de misiones. Los UAV tienen varias ventajas básicas sobre sistemas tripulados incluyendo mayor maniobrabilidad, coste reducido, radar reducido, mayor resistencia y menos riesgo para las tripulaciones.

D. IMPACTO SOCIAL.

El diseño y la implementación de este tipo de proyectos contribuyen directamente en el mejoramiento y la innovación de las técnicas y herramientas que son necesarias en situaciones de riesgo.

Permite suplir la necesidad de desplazamiento de personal en lugares de difícil acceso tales como edificios en ruinas, espacios con un alto grado de contaminación, y lugares donde el ser humano no le es posible acceder por sus dimensiones, previene todo tipo de riesgos al personal en situaciones de vigilancia y monitoreo evitando el contacto cuerpo a cuerpo.

En el área académica, estimula la investigación en temáticas relacionadas, vincula a la comunidad con la innovación e influye positivamente en la proyección de metas académicas.

E. MODELO DINÁMICO DEL QUAD-ROTOR.

En esta sección se describen el modelo dinámico del Quad-Rotor. Este modelo se obtiene básicamente porque representa el Quad-Rotor como un cuerpo sólido en 3D sujeto a una fuerza y tres momentos. La dinámica de los cuatro motores eléctricos es relativamente rápida, por lo tanto no es tenida en cuenta, igualmente se desprecia la flexibilidad de las aspas de las hélices.

Las coordenadas generalizadas para el helicóptero son:

$$q = (x, y, z, \psi, \theta, \varphi) \in R^6 \quad (1)$$

Donde (x, y, z) indican la posición del centro de masa de los cuatro rotores del helicóptero y (ψ, θ, φ) son los tres ángulos de Euler que representan la orientación del helicóptero.

Por lo tanto, el modelo divide naturalmente las coordenadas de traslación y rotación:

$$\xi = (x, y, z) \in R^3, \quad \eta = (\psi, \theta, \varphi) \in S^3 \quad (2)$$

La energía cinética de traslación del helicóptero es:

$$T_{trans} \triangleq \frac{m}{2} \dot{\xi}^T \dot{\xi} \quad (3)$$

donde m es la masa del helicóptero.

La energía cinética rotacional es:

$$T_{rot} \triangleq \frac{1}{2} \dot{\eta}^T J \dot{\eta} \quad (4)$$

La matriz J actúa como la matriz de inercia para la energía cinética rotacional completa del helicóptero, expresada directamente en términos de las coordenadas generalizadas η .

La única energía potencial que debe ser considerada es el potencial gravitacional estándar dado por:

$$U = mgz \quad (5)$$

El Lagrangiano es

$$L(q, \dot{q}) = T_{trans} + T_{rot} - U \quad (6)$$

$$L(q, \dot{q}) = \frac{m}{2} \dot{\xi}^T \dot{\xi} + \frac{1}{2} \dot{\eta}^T J \dot{\eta} - mgz \quad (7)$$

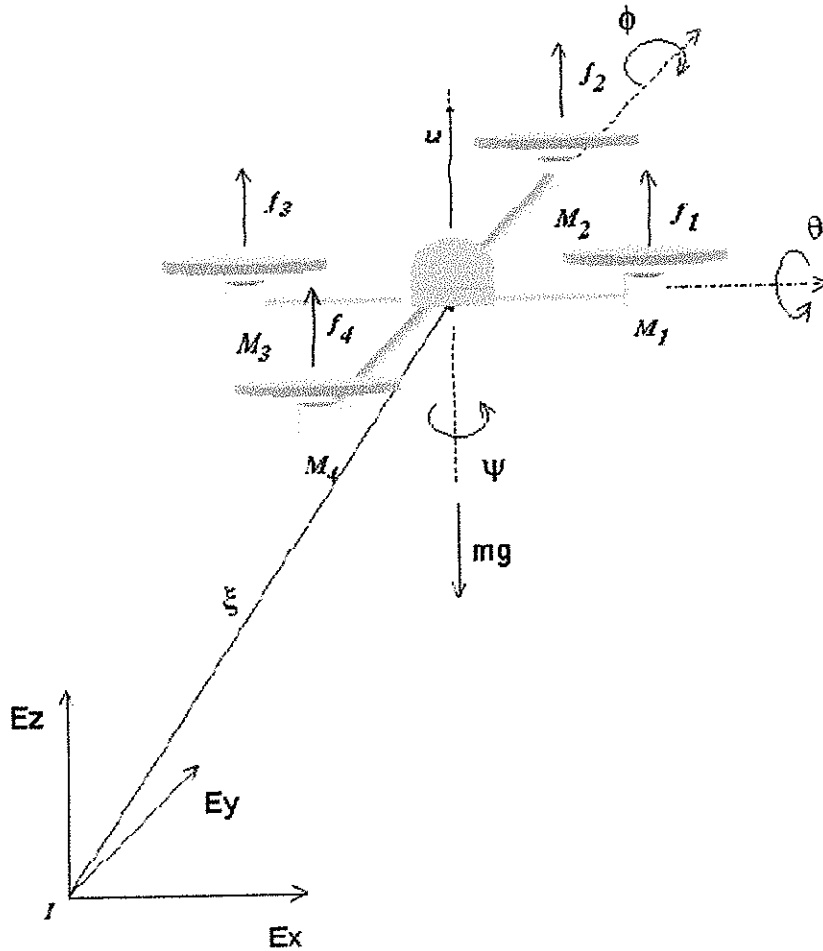


Imagen 13. Quad-Rotor en una estructura inercial.

Fuente: Modelling and Control of Mini-Flying Machines, Pedro Castillo, Rogelio Lozano and Alejandro E. Dzul. France 2004.

El modelo para la dinámica del helicóptero completo es obtenido de las ecuaciones de LaGrange Euler con fuerza generalizada externa:

$$\frac{d}{dt} \frac{\partial \mathcal{L}}{\partial \dot{q}} - \frac{\partial \mathcal{L}}{\partial q} = F \quad (8)$$

Donde $F = (F\xi, \tau)$. τ son los momentos generalizados y $F\xi$ es la fuerza de traslación aplicada al helicóptero debido a las entradas de control. Se descartan las fuerzas de cuerpo pequeño porque generalmente son de una magnitud mucho menor que la u de entradas de control principal y τ .

Luego se escribe:

$$\hat{F} = \begin{pmatrix} 0 \\ 0 \\ u \end{pmatrix} \quad (9)$$

Donde:

$$u = f1 + f2 + f3 + f4 \quad (10)$$

y

$$Fi = kiwi^2, \quad i = 1, \dots, 4 \quad (11)$$

Donde $ki > 0$ es una constante y w_i es la velocidad angular del motor "i" ($M_i, i = 1, \dots, 4$)

A continuación,

$$F\xi = \hat{F} \quad (12)$$

donde R es la matriz de transformación que representa la orientación del helicóptero.

Se utiliza $c\theta$ para $\cos\theta$ y $s\theta$ para $\sin\theta$.

$$R = \begin{pmatrix} c\theta c\psi & s\psi s\theta & -s\theta \\ c\psi s\theta s\varphi - s\psi c\varphi & s\psi s\theta s\varphi + c\psi c\varphi & c\theta s\varphi \\ c\psi s\theta c\varphi + s\psi s\varphi & s\psi s\theta c\varphi - c\psi s\varphi & c\theta c\varphi \end{pmatrix} \quad (13)$$

Son los momentos generalizados sobre las variables η

$$\tau = \begin{pmatrix} \tau\psi \\ \tau\theta \\ \tau\varphi \end{pmatrix} \quad (14)$$

Donde;

$$\tau\psi = \sum_{i=1}^4 \tau M_i \quad (15)$$

$$\tau\theta = (f2 - f4) \ell \quad (16)$$

$$\tau\varphi = (f3 - f1) \ell \quad (17)$$

Donde ℓ es la distancia desde los motores hasta el centro de gravedad y τM_i es el par

producido por el motor de M_i .

Puesto que el Lagrangiano no contiene ningún término de producto cruz en la combinación de energía cinética $\dot{\xi}$ y $\dot{\eta}$, la ecuación de Euler-LaGrange puede ser dividida en la dinámica de las coordenadas ξ y la dinámica η .

$$m\ddot{\xi} + \begin{pmatrix} 0 \\ 0 \\ mg \end{pmatrix} = F_{\xi} \quad (18)$$

$$J\ddot{\eta} + \dot{J}\dot{\eta} - \frac{1}{2} \frac{\partial}{\partial \eta} (\dot{\eta}^T J \dot{\eta}) = \tau \quad (19)$$

Definiendo el vector Coriolis/centrípeto

$$\bar{V}(\eta, \dot{\eta}) = \dot{J}\dot{\eta} - \frac{1}{2} \frac{\partial}{\partial \eta} (\dot{\eta}^T J \dot{\eta}) \quad (20)$$

Se puede escribir:

$$J\ddot{\eta} + \bar{V}(\eta, \dot{\eta}) = \tau \quad (21)$$

Pero se puede rescribir $\bar{V}(\eta, \dot{\eta})$ como

$$\bar{V}(\eta, \dot{\eta}) = (J\dot{\eta} - \frac{1}{2} \frac{\partial}{\partial \eta} (\dot{\eta}^T J \dot{\eta})) \dot{\eta} \quad (22)$$

$$\bar{V}(\eta, \dot{\eta}) = C(\eta, \dot{\eta})\dot{\eta} \quad (23)$$

Donde $C(\eta, \dot{\eta})$ se conoce como los términos de Coriolis y contiene los términos giroscópicos y centrífugos asociados con η dependiendo de J .

Finalmente se obtiene:

$$m\ddot{\xi} = u \begin{pmatrix} -\sin \theta \\ \cos \theta \sin \varphi \\ \cos \theta \cos \varphi \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ -mg \end{pmatrix} \quad (24)$$

Para simplificar la ecuación se propuso un cambio de las variables de entrada:

$$\tau = C(\eta, \dot{\eta})\dot{\eta} + J\ddot{\tau} \quad (25)$$

donde,

$$\tilde{\tau} = \begin{pmatrix} \tilde{\tau}\psi \\ \tilde{\tau}\theta \\ \tilde{\tau}\varphi \end{pmatrix} \quad (26)$$

son las nuevas entradas, entonces:

$$\ddot{\eta} = \tilde{\tau} \quad (27)$$

Rescribiendo las ecuaciones (18) y (19):

$$m\ddot{x} = -u \sin\theta \quad (28)$$

$$m\ddot{y} = u \cos\theta \sin\varphi \quad (29)$$

$$m\ddot{z} = \cos\theta \cos\varphi - mg \quad (30)$$

$$\ddot{\psi} = \tilde{\tau}\psi \quad (31)$$

$$\ddot{\theta} = \tilde{\tau}\theta \quad (32)$$

$$\ddot{\varphi} = \tilde{\tau}\varphi \quad (33)$$

donde x y y son las coordenadas en el plano horizontal, y z es la posición vertical (ver Imagen 13) ψ es el ángulo de guiñada alrededor del eje z , θ es el ángulo alrededor del eje y también llamado cabeceo, φ es el ángulo alrededor del eje x también llamado alabeo.

Las entradas de control u , $\tilde{\tau}\psi$, $\tilde{\tau}\theta$ y $\tilde{\tau}\varphi$ son el total de empuje o entrada colectiva (dirigida hacia afuera desde la parte inferior de la aeronave) y los nuevos momentos angulares (desviación de momento, momento de cabeceo y el balanceo de momento).

F. CARACTERÍSTICAS DEL HELICÓPTERO QUAD-ROTOR.

El quad-rotor mostrado en la Imagen 14, muestra algunas de las ventajas sobre los helicópteros convencionales. Se aprecia los sentidos de giro de los motores actuando de tal manera que cancelan entre si el vuelo descontrolado.

G. CONTROL DE QUADROTORES.

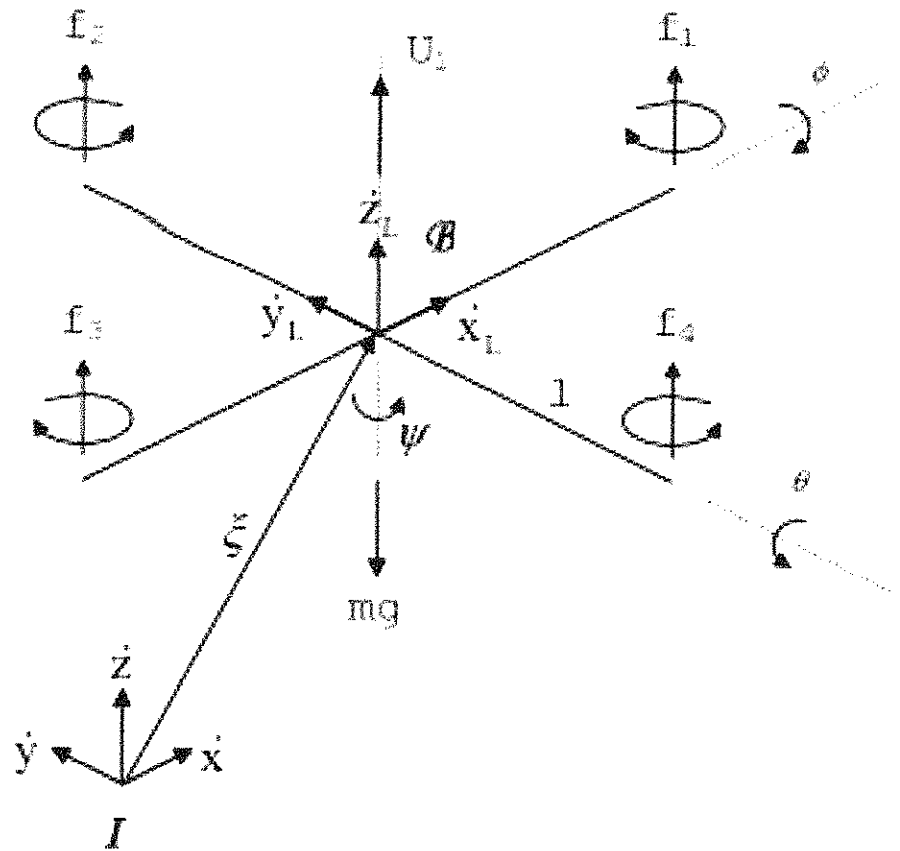


Imagen 15. Esquema de fuerzas del helicóptero quadrotor.

Fuente: MODELADO Y CONTROL DE UN HELICÓPTERO QUADROTOR, GUILHERME VIANNA RAFFO, Sevilla, Diciembre de 2007.

Se supone el vehículo como un cuerpo rígido en el espacio, sujeto a una fuerza principal (empuje) y tres momentos (pares). En la Imagen 15, se muestra las fuerzas que ejercen las distintas hélices para generar el movimiento del vehículo.

El quad-rotor no tiene un solo rotor cíclico demostrando así que no es necesario un control de pitch. La salida de potencia y movimiento es el resultado de la suma de las potencias suministradas a los rotores.

Los movimientos fundamentales de un quad-rotor son:

Pitch (ángulo θ) = cabeceo = movimiento que tiene como centro el eje Y.

Roll (ángulo φ) = alabeo = movimiento que tiene como centro el eje X.

Yaw (ángulo ψ) = guiñado = movimiento que tiene como centro el eje Z.

La dinámica de los movimientos es la siguiente: El par para generar un movimiento de balanceo o de roll (ángulo ϕ) se realiza mediante un desequilibrio entre las fuerzas f_2 y f_4 (ver Imagen 16). Para el movimiento de cabeceo o de pitch (ángulo θ), el desequilibrio se realizará entre las fuerzas f_1 y f_3 . El movimiento en el ángulo de guiñada o de yaw (ángulo ψ) se realizará por el desequilibrio entre los conjuntos de fuerzas (f_1, f_3) y (f_2, f_4). Este movimiento será posible ya que los rotores 1 y 3 giran en sentido contrario a los rotores 2 y 4. Finalmente, el empuje total, que hará que el helicóptero se desplace perpendicularmente al plano de los rotores, se obtendrá como suma de las cuatro fuerzas que ejercen los rotores.

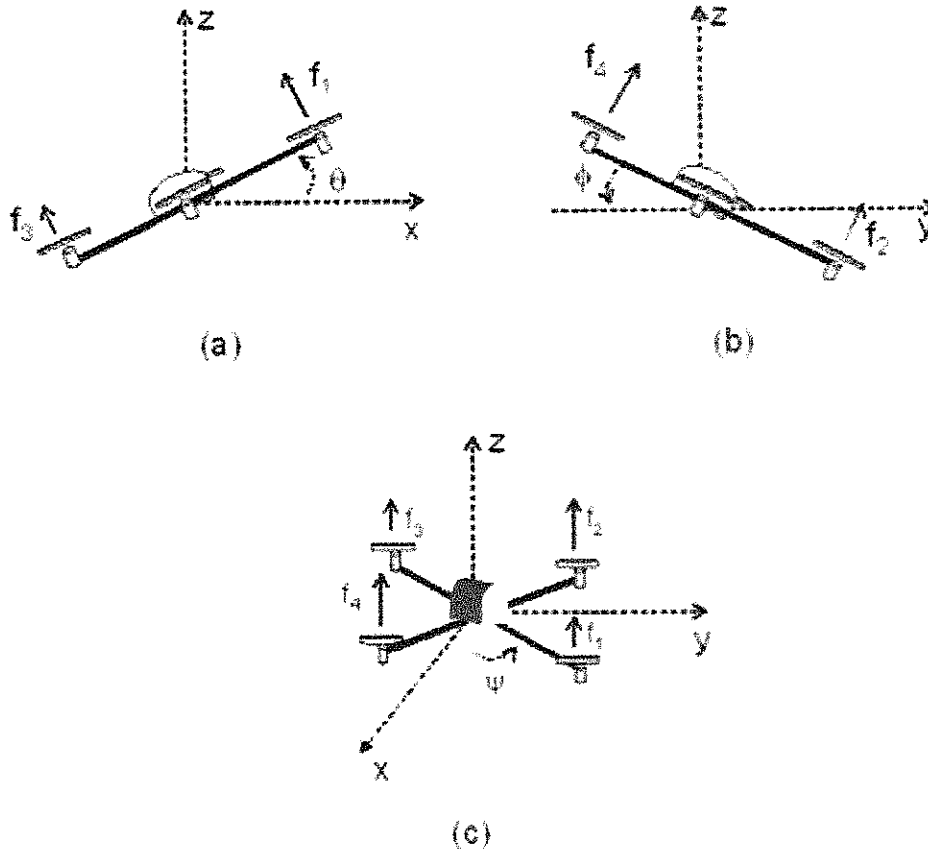


Imagen 16. (a) Pitch, (b) Roll y (c) Yaw.

Fuente: Modelling and Control of Mini-Flying Machines, Pedro Castillo, Rogelio Lozano and Alejandro E. Dzul. France 2004.

- Movimientos en el eje Y (pitch +, ángulo θ):
Incrementar la fuerza $f1$ y reducir la fuerza $f3$, mientras $f2$ y $f4$ permanecen estable.
- Movimientos en el eje Y (pitch -, ángulo θ):
Incrementar la fuerza $f3$ y reducir la fuerza $f1$, mientras $f2$ y $f4$ permanecen estable.
- Movimientos en el eje X (roll +, ángulo ϕ):
Incrementar la fuerza $f4$ y reducir la fuerza $f2$, mientras $f1$ y $f3$ permanecen estable.
- Movimientos en el eje X (roll - ángulo ϕ):
Incrementar la fuerza $f2$ y reducir la fuerza $f4$, mientras $f1$ y $f3$ permanecen estable.

Continuando con la estrategia de control para la estabilización de un helicóptero quad-rotor. El procedimiento de síntesis del controlador regula cada una de las variables de estado mediante las siguientes reglas de prioridad:

ESTRATEGIA DE CONTROL

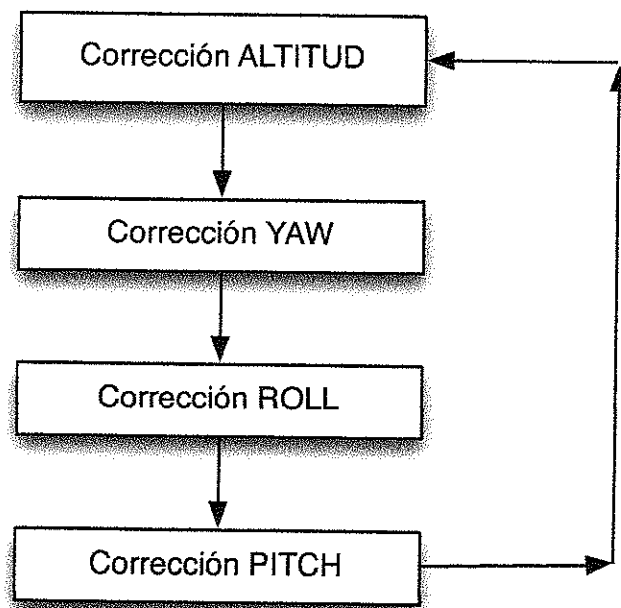


Imagen 17. Diagrama Funcional Estrategia de Control.

Fuente: Autor.

Primero es necesario el diseño de un controlador para estabilizar el desplazamiento angular Yaw, luego se realiza el control del movimiento angular Roll y finalmente se controla el ángulo de inclinación pitch.

La estrategia de control propuesta es relativamente simple, pero muy efectiva cuando se realizan

aplicaciones en tiempo real. Es importante tener en cuenta que la entradas de control pueden actuar independientemente dependiendo el modo de vuelo siendo manual o automático, por razones de seguridad de vuelo esta característica es particularmente importante cuando se aplica la estrategia de control. Como conclusión cada una de las entradas de control puede utilizarse para controlar uno o dos grados de libertad como se muestra en la Tabla 2.

FASE	NOMBRE	DESCRIPCIÓN
1	Control de Altitud	τu Es usa para hacer que alcance la altitud deseada.
2	control de Yaw	$\tau \psi$ Es usa para hacer el desplazamiento yaw a cero.
3	control de Roll	$\tau \phi$ Es usado para controlar roll ϕ y el movimiento horizontal del eje Y.
4	control de Pitch	$\tau \theta$ Es usado para controlar pitch θ y el movimiento horizontal en el eje X.

Tabla 2. Estrategia de Control.

Fuente: Modelling and Control of Mini-Flying Machines, Pedro Castillo, Rogelio Lozano and Alejandro E. Dzul. France 2004.

Los quadrotores siendo plataformas cada vez más populares de vehículos aéreos no tripulados (UAV), han permitido el desarrollo de todo tipo de métodos de control iniciando con el método clásico más usado como lo es el control PID (proporcional, integral y derivativo) de ganancia fija ofreciendo un control estable en sistemas dinámicos ya que entregan tiempos de respuesta rápidos, pero entre mayor es la precisión requerida en el sistema resulta más difícil el ajuste de este tipo de controlador, debido a su gran sensibilidad ante señales de ruido y en ocasiones la presencia de oscilaciones cuando se existen retardos en el sistema. En situaciones en las cuales la dinámica de los sistemas o procesos a controlar es no lineal, es necesario que el controlador tenga la capacidad de compensar esta no-linealidad, el controlador PID asume relaciones lineales pero

no tiene la capacidad de responder en estos tipo de sistemas, en ocasiones se suele compensar las falencias del controlador PID implementándolos en un microcontrolador de lógica difusa, ofreciendo la capacidad de ejercer acciones de control mediante definición de parámetros cualitativos de control y la capacidad de implementar un comportamiento de control altamente flexible.

El control difuso parece ser la herramienta que puede suplir las falencias donde el controlador PID puede ser efectivo ofreciendo un cubrimiento en todas las situaciones posibles en las cuales el prototipo se puede ver envuelto.

Basados en el software de desarrollo Matlab® se planteo el control puntual del modelo quadrotor con todas las condiciones de vuelo y manejo de conjuntos de entrada y salida, para los cuales un control difuso ofrece una cantidad de combinaciones de estados, lo cual conlleva al controlador a navegar dentro de un numero equivalente de condiciones de trabajo para así obtener una respuesta del control para cada punto de operación posible del prototipo.

H. FACTORES EXTERNOS QUE INFLUYEN EN LOS UAVS.

1. VIENTO.

Todas las maquinas voladoras se ven afectadas por condiciones ambientales, como las corrientes de aire presentes en el espacio de vuelo, ya que estas afectan la estabilidad del prototipo actuando como perturbación ante los métodos de control de cada aeronave. De manera teórico-práctica el quadrotor realiza su vuelo creando una capa de aire suficientemente densa para soportar su peso incrementando esta densidad a medida que la potencia de sus rotores aumente y reduce su densidad si se disminuye la potencia de los rotores.

2. TEMPERATURA.

Los dispositivos incluidos dentro del prototipo como sensores actuadores y elementos de ensamble poseen un rango de temperatura de -25° a 125° , dentro del cual se considera confiable su operación, fuera de estos límites los dispositivos actúan de manera errónea afectando sus resultados y exponiéndose a salir de operación definitivamente.

3. PESO.

Se considera ideal un dispositivo que sea capaz de transformar toda su energía en fuerza, basados en este concepto y caracterizando el prototipo diseñado se tiene que la influencia del peso del prototipo afecta el rendimiento del mismo. Cuando el peso es menor; la energía aplicada a los motores es mas efectiva realizando menos esfuerzo y ahorrando energía de las baterías que luego se transformará en más tiempo de vuelo. Cuando el peso es mayor; la energía aplicada a los motores tiene que ser mayor para tener el mismo resultado que en el ejemplo anterior, si se extrae mayor energía de las baterías se reduce el tiempo de vuelo significativamente.

VI. SELECCIÓN DE HARDWARE QUADCOPTER

El capítulo VI explica de una manera detallada todos los aspectos de diseño del hardware tenidos en cuenta para el desarrollo del proyecto, en primera instancia es realizada una distribución temática iniciando con la selección de dispositivos, abordando temas puntuales como sensores, procesador, actuadores y demás accesorios necesarios para el desarrollo del prototipo.

Dentro del proceso de selección de los dispositivos se tuvo en cuenta varios parámetros de acuerdo a los requerimientos básicos del prototipo como lo son la capacidad de mantenerse en vuelo y el correcto funcionamiento en conjunto de todos los dispositivos para garantizar la estabilidad del mismo.

Los parámetros de selección se determinaron a través de la investigación previa al inicio del proyecto los cuales se detallan posteriormente a medida que se aborden los temas específicos de selección de dispositivos.

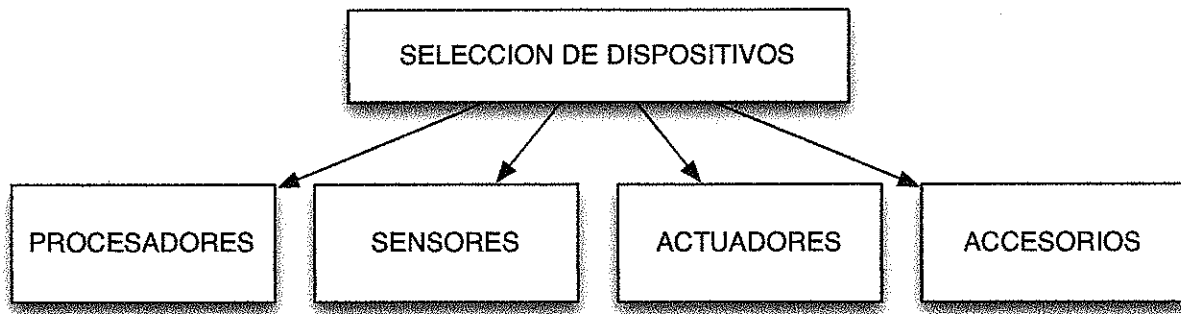


Imagen 18. Distribución de Temáticas Selección de Dispositivos

Fuente: Autor.

En la Imagen 18. Se muestra la distribución de temáticas que se abordaron en esta sección iniciando por los parámetros de selección del procesador Arduino donde se identifican sus principales características y se realiza una comparación directa con los microcontroladores PIC.

Posteriormente es realizado el análisis de los dispositivos sensores y sus parámetros de selección para el presente proyecto, el proceso de identificación y selección de actuadores (motores) y finalmente se establece la selección completa de accesorios indispensables para el proyecto como la fuente de alimentación y hélices entre otros.

A. PROCESADOR

La selección del microprocesador a utilizar en este proyecto se realizó en base a comparaciones entre los microprocesadores ARDUINO y PIC ya que son dispositivos del mismo nivel y con los cuales el autor ha tenido experiencia de trabajo.

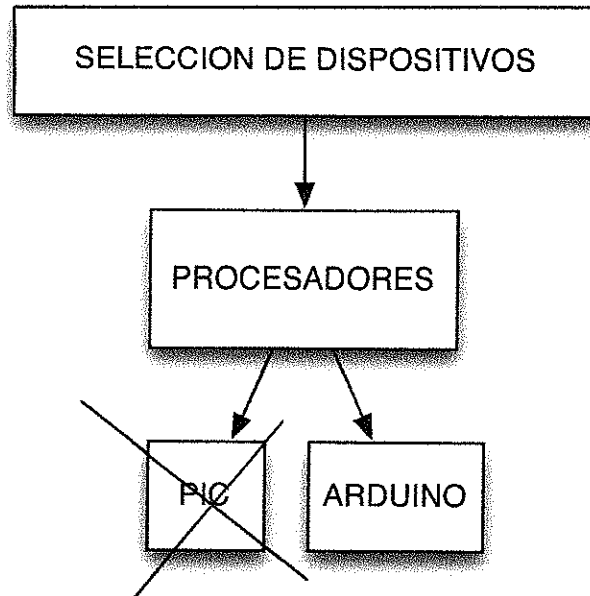


Imagen 19. Selección de Procesador.

Fuente: Autor.

Lo que se buscó fue seleccionar un procesador capaz de leer los datos de varios sensores procesarlos, leer las señales provenientes del radio control seleccionado para el proyecto, comodidad en su manejo, además interpretar y ejercer el respectivo control sobre los actuadores. El procesador seleccionado para el proyecto fue el Arduino Mega2560 (Imagen 20) ya que satisface los parámetros de selección planteados para el desarrollo del proyecto.

1. ARDUTNO

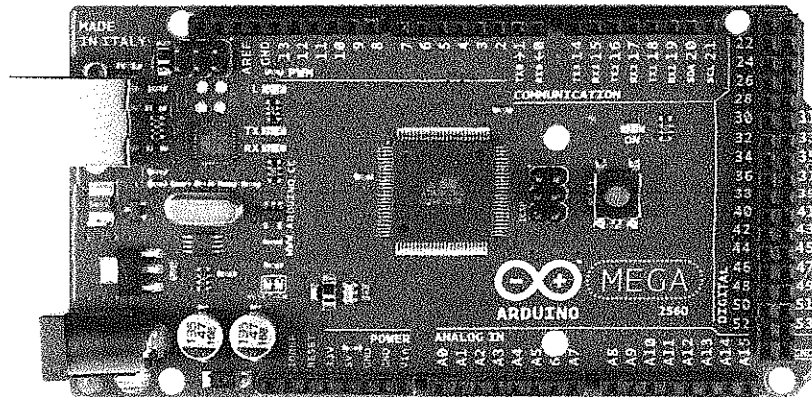


Imagen 20. Arduino Mega2560.

Tomado de: www.arduino.cc/es

Arduino mega 2560 es una board basada en el microcontrolador ATmega2560¹⁰ que cuenta con 54 pines de entradas/salidas digitales de las cuales 14 pueden ser usadas como PWM (Pulse-Width Modulation), además posee 16 entradas analógicas, 4 módulos UART (Universal Asynchronous Receiver-Transmitter), un cristal oscilador de 16 MHz. Adicionalmente cuenta con una conexión USB (Universal Serial Bus), un conector de poder regulado, conector para ICSP (In-Circuit Serial Programming) y un botón de reset, la board contiene todo lo necesario para apoyar el microcontrolador con tan solo conectarlo a un computador con un cable USB o alimentación con una batería o adaptador AC-DC para empezar, esta board es compatible con la mayoría de Shields diseñados para productos Arduino Duemilanove¹¹ y Diecimila¹².

La tarjeta Arduino Mega 2560 cuenta con los pines SDA (Serial Data) y SCL (Seria Clock) para establecer la comunicación basada en el protocolo I²C (Inter-Integrated Circuit), estos pines funcionan de igual manera para la comunicación SPI(Serial Peripheral Interface).

Es posible trabajar este microprocesador con interrupciones con funciones específicas dentro de la interfaz de Arduino IDE (Interfaz de Arduino). El software de Arduino se encuentra disponible para los sistemas operativos Windows, Mac OS X y Linux.

¹⁰ <http://www.atmel.com/Images/doc2549.pdf>

¹¹ <http://www.arduino.cc/en/Main/ArduinoBoardDuemilanove>

¹² <http://www.arduino.cc/en/Main/arduinoBoardDiecimila>

La plataforma Arduino es compatible y se encuentra enlazada directamente con software de nivel profesional como Matlab® y Labview™ permitiendo expandir mediante una conexión USB los alcances de estos Software actuando como una tarjeta económica y efectiva de adquisición de datos. El sistema de desarrollo de Arduino puede configurarse para una gran variedad de Microcontroladores AVR.

Visual Studio es un software, usado para la programación alternativa de Arduino que incluye: tablero de selección, complementación de código, buscar y reemplazar archivos, bibliotecas, núcleo y usuario, compilación (con GoTo error), carga, programador, múltiples ventanas de herramientas, acoplable con puerto serie y más. Es 100% compatible con Arduino y puede actualizarse para incluir un depurador.

Microcontrolador	ATmega2560
Voltaje de Operación	5V
Voltaje de Entrada (recomendado)	7 – 12V
Voltaje de Entrada (limites)	6 – 20V
I/O Digitales	54 (de los cuales 15 salidas PWM)
Entradas Análogas	16
Corriente por pin I/O	40 mA
Corriente por pin 3.3V I/O	50 mA
Memoria Flash	256 KB de los cuales 8 KB usados para Bootloader
SRAM	8 KB
EEPROM	4 KB
Velocidad de Reloj	16 MHz

Tabla 3. Características microcontrolador ATmega2560.

Tomado de: www.arduino.cc/es

Los microcontroladores tenidos en cuenta en esta comparación fueron los microcontroladores de gama media PIC de la familia Microchip y los microcontroladores ATmega de la familia Atmel. En el momento de seleccionar el microcontrolador idóneo es necesario analizar parámetros

regulares como entradas, salidas, velocidad, memoria y efectividad.

La familia Atmel al igual que la familia Microchip son empresas de nivel mundial con una gran trascendencia en el desarrollo de dispositivos electrónicos se especializan en diversos frentes de trabajo enfocados siempre al desarrollo y la innovación tecnológica.

En cuanto a criterios personales, luego de haber experimentado los dos microprocesadores se seleccionó trabajar con Arduino porque es cómodo en su programación ya que el lenguaje Arduino está basado en C/C++ y soporta todas las construcciones de C estándar y algunas funcionalidades de C++, además vincula la librería AVR Libc y permite el uso de todas sus funciones.

A diferencia de los micro controladores PIC, Arduino viene completamente listo para trabajar ofreciendo la comodidad de su conexión USB sin necesidad de utilizar accesorios para su programación, debido a la experiencia personal, Arduino simplifica las tareas teniendo librerías de uso general, permitiendo la creación y modificación de las mismas dependiendo de los requerimientos del proyecto, todas sus librerías oficiales son totalmente confiables reduciendo así el tiempo de programación y creación de funciones que en lenguajes como Assembler se convierte en una multiplicación de tiempo debido a la interacción entre bancos de configuración ofrecidos por los microcontroladores PIC.

El tiempo de compilación y programación se reduce en un 70 % con un Arduino teniendo la posibilidad de tener asesoría de un sin fin de sitios de internet especializados en este hardware ya que al ser trabajado en una plataforma de software libre permite realizar modificación y vinculo directamente con las raíces del microcontrolador omitiendo las configuraciones preestablecidas por el fabricante.

Arduino tiene un enlace directo con la plataforma Android permitiendo el desarrollo directo de aplicaciones en el mundo siempre creciente Android.

Otra razón para manejar Arduino es la facilidad de que su tarjeta de desarrollo ya está fabricada y no se consideró necesario volver a diseñar y realizar PCB (Printed Circuit Board) para la conclusión de un proyecto, es claro que la información disponible para los microcontroladores PIC es mucho más extensa que para los microcontroladores Atmel pero también es claro que la bibliografía e infografía disponible para Atmel es muy completa no en cantidad sino en calidad y facilidad de interpretación.

Existen soluciones ATMEL con USB nativo, por lo que no se puede hablar del PIC 18F2550 que

es un PIC de grandes prestaciones, sin hablar del AT90USB32U4 que es similar conceptualmente, Existen compiladores gratuitos para ATMEL así que con estos parámetros no se puede definir cuál es superior. Reiterando, la decisión está en el desarrollador, que se siente más cómodo con tal microcontrolador y no tanto con el otro.

En cuanto a parámetros comerciales y costos el ATmega es cerca de tres dólares mas económico que el microcontrolador PIC de su mismo nivel.

Actualmente los desarrollos de robótica fundamentados de Atmel se están direccionando a la plataforma Arduino siendo una de las mejores soluciones cuando la finalidad es acortar tiempos haciendo uso de placas y soportes conocidos, luego de investigar en la red y bajo conceptos de docentes se tomó como conclusión que los proyectos desarrollados con un microcontrolador son igualmente desarrollados con el otro.

Arduino es una plataforma de desarrollo electrónico abierto con la finalidad de facilitar la creación de prototipos basado en software y hardware flexibles y fáciles de usar, permite tomar información de su entorno y fue creado para desarrolladores, diseñadores, artistas, aficionados y cualquier persona que se encuentre interesada en crear entornos u objetos interactivos.

Se considera un empate técnico entre los dos microcontroladores base de esta comparación pero de una forma conveniente se escogió Arduino.

En todo caso el programador decide que dispositivos utiliza de acuerdo a sus herramientas disponibles basado en experiencia y criterios de ingeniería elementales en la elección de dispositivos.

a. LIBRERÍAS

Debido al manejo de diferentes dispositivos y protocolos, fue necesario el uso de librerías independientes las cuales se enumeran enseguida.

- Librería *servo.h*

Esta librería permite a una placa Arduino controlar servo motores RC (hobby). Los servos tienen engranajes integrados y un eje que se puede controlar con precisión. Los servos estándar permiten

que el eje se sitúe en varios ángulos, por lo general entre 0 y 180 grados. Los servos de rotación continua permiten la rotación del eje a distintas velocidades¹³.

- Librería *wire.h*

Esta librería permite comunicarse bajo el protocolo I²C. En la placa Arduino las líneas de comunicación son, SDA (línea de datos) está en el pin 20 y SCL (línea de reloj) está en el pin 21. Esta biblioteca actúa similar que otras bibliotecas de lectura y escritura. Debido a esto, se manejan funciones similares como read() y write()¹⁴.

- Librería *LSM303.h*

Esta es una librería para Arduino que interactúa con brújulas LSM303DLH, LSM303DLM, LSM303DLHC y los acelerómetros en boards de Pololu¹⁵. La lectura de los datos del magnetómetro y acelerómetro es sencilla, la biblioteca también incluye una función para calcular el encabezamiento compensado de inclinación para ser usado con el LSM303DLH como una brújula de inclinación compensada¹⁶.

- Librería *L3G4200D.h*

Esta biblioteca estructurada para Arduino interactúa con el giróscopo L3G4200D realiza la lectura de los datos crudos de los ejes X, Y y Z, permite la configuración de los registros indicados en la hoja de datos del mismo sensor¹⁷.

B. SENSORES

Debido a los requerimientos del proyecto y de acuerdo a la investigación realizada se identificó la posición exacta del prototipo ya que para realizar un proceso de control es necesario tener la información de su posición.

En la selección de sensores se tuvo en cuenta parámetros esenciales como resolución, confiabilidad, velocidad de respuesta, protocolo de comunicación, rango y durabilidad.

Inicialmente se planteó realizar el proyecto con un solo sensor MEMS (Micro Electro Mechanical Systems) giróscopo de tres ejes; el giroscopio fue escogido por que entrega la información de velocidad angular necesaria para identificar la posición del modelo.

¹³ <http://www.arduino.cc/es/Reference/Servo>

¹⁴ <http://www.arduino.cc/en/Reference/Wire>

¹⁵ <http://www.pololu.com/>

¹⁶ <https://github.com/pololu/LSM303>

¹⁷ <https://github.com/pololu/L3G4200D>

Un giróscopo digital de este tipo entrega información de su entorno gracias a la fuerza que ejerce la gravedad sobre su hardware tomando como referencia su eje simétrico central de los tres ejes *X*, *Y* y *Z*.

Para realizar la selección del giróscopo adecuado se partió de la comparación de los giróscopos presentes en el mercado que se ajustaron a las características arriba anunciadas.

Dentro de los sensores que se tuvieron en cuenta se encuentra el giroscopio L3G4200D¹⁸ suministrado por la empresa STMicroelectronics disponible en el mercado y base de comparación con otros dispositivos sensores.

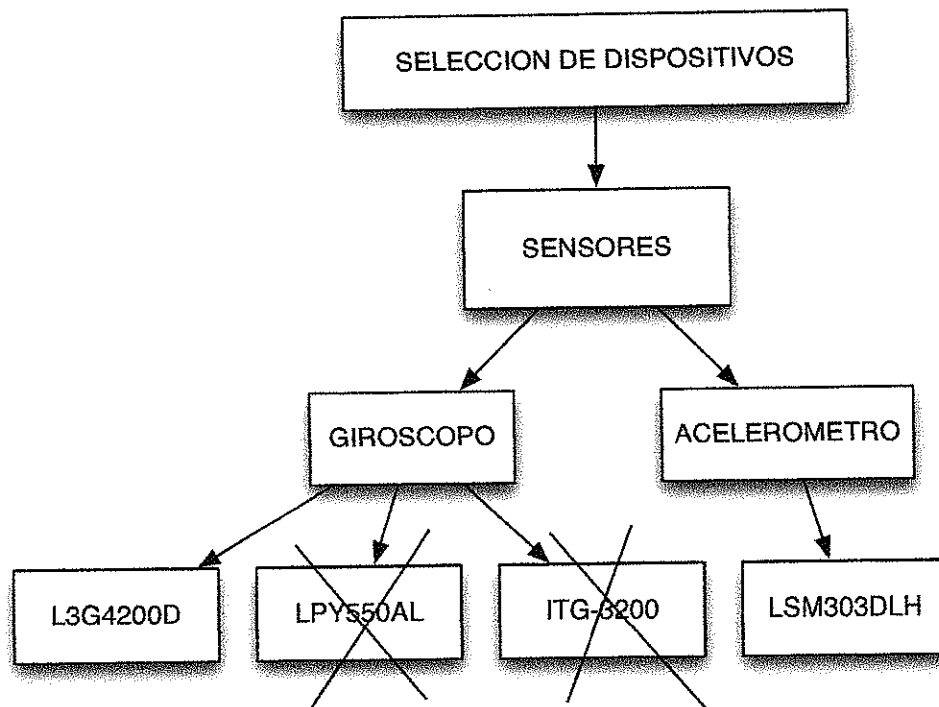


Imagen 21. Proceso de Selección de Sensores.

Fuente: Autor.

En la imagen 21. Se explica el proceso de selección de sensores mediante un diagrama de secuencias.

¹⁸ <http://www.st.com/jp/analog/product/250373.jsp>

1. L3G4200D

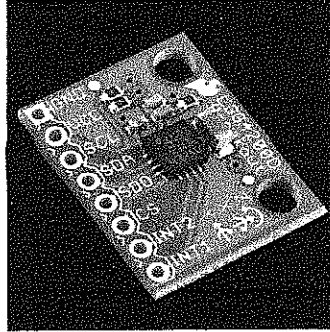


Imagen 22. Sensor L3G4200D.

Tomado de: www.sparkfun.com

El módulo Giroscopio L3G4200D¹⁹ es un sensor de rango angular de 3 ejes con datos de temperatura. El giróscopo muestra el cambio de rango en rotación en sus ejes *X*, *Y* y *Z*. Los datos de salida de temperatura y el rango angular medido. Los datos de salida de temperatura y la medición angular se pueden acceder de la interface seleccionable I²C o SPI. El módulo tiene acceso a una interface SIP (Single Input Package) con un orificio de montaje para una rápida conexión a los proyectos. El módulo está diseñado para trabajar con una amplia gama de micro controladores y diferentes requerimientos de voltaje.

2. LPY550AL

Según la oferta de sensores en el mercado se tuvo en cuenta el sensor giróscopo dos ejes LPY550AL ofrecido por la compañía POLOLU ROBOTICS AND ELECTRONICS.

¹⁹ <http://dlnmh9ip6v2uc.cloudfront.net/datasheets/Sensors/Gyros/3-Axis/CD00265057.pdf>

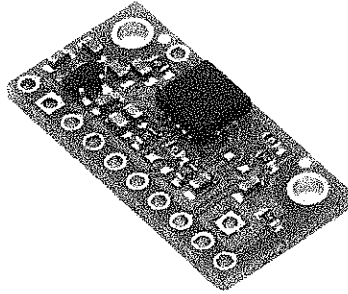


Imagen 23. Sensor LPY550AL.

Tomado de: www.pololu.com

Este gir6scopo montado en board, se basa en el sensor de giro LPY550 de dos ejes (Pitch and Yaw), el cual mide a velocidad angular rotaci6n en el eje X y Z , las se1ales tomadas son de tipo an6logo y viene cada eje en un canal separado, esta board integra un regulador de $3.3V$ para hacer f6cil la integraci6n en sistemas de $5V$, adem6s incluye un filtro pasa bajos para la reducci6n de ruido, las aplicaciones para las que se usan los gir6scopo van desde control de posici6n en rob6tica m6vil hasta veh6culos a6reos aut6nomos.

3. ITG-3200

Finalmente se compar6 con el Giroscopio ITG-3200 con Tres Ejes y Salida Digital.

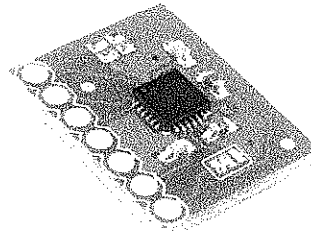


Imagen 24. Sensor ITG-3200.

Tomado de: www.sparkfun.com

El ITG-3200 es un gir6scopo de tres ejes digital, incluye tres conversores an6logo-digital de 16 bits (ADCs) para la digitalizaci6n de las salidas del gir6scopo, tiene un filtro pasa bajos configurable por el usuario, Fast-mode e interfaz I²C (400 kHz).

PAR6METRO	L3G4200D	ITG-3200	LPY550AL
RANGO	3 ejes	3 ejes	2 ejes
COMUNICACI6N	I ² C y SPI	I ² C	Anal6gico.
ESCALA	250/500/2000 grados/seg.	2000 grados/seg.	500/2000 grados/seg.
RANGO SALIDA DIGITAL DE DATOS	16 BITS	16 BITS	N/A
VOLTAJE DE OPERACI6N	2.7 a 6.5 V	2,1 a 3,6 V	3V
FILTRO DIGITAL	PASA BAJO/ALTOS	PASA BAJO	N/A
DIMENSIONES	21.6 X 20.3 mm	17,78 x 13,97 mm	12.7 x 25.4 mm

Tabla 4. Comparaci6n Par6metros Sensor Gir6scopo.

Fuente: Autor.

Como resultado de los tres sensores se escogi6 en L3G4200D debido a que es un sensor digital, cuenta con el protocolo I²C y adem6s posee con escalas de sensibilidad seleccionables.

Luego de realizar el proceso de interpretaci6n de los datos se tuvo como resultado que este sensor ten6a una gran deriva (drift) la cual fue necesario compensarla con otro sensor el cual tuviera como referencia los ejes terrestres. Y se opt6 por utilizar la IMU LSM303DLH la cual compens6 la deriva del gir6scopo y agreg6 precisi6n a los datos le6dos de los tres sensores.

4. LSM303DLH

El LSM303DLH²⁰ combina un aceler6metro digital de 3 ejes y magnet6metro de tres ejes en un solo integrado, lo que lo hace ideal para aplicaciones de br6julas con inclinaci6n compensada entre otras.

²⁰ http://www.pololu.com/file/download/LSM303DLHC.pdf?file_id=0J564

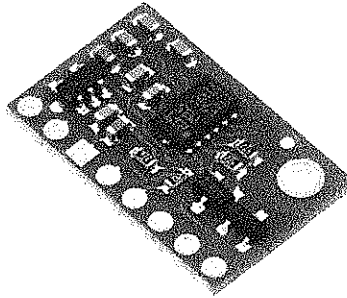


Imagen 25. Sensor LSM303DLH

Tomado de: www.pololu.com

Este sensor acopla una brújula magnética de tres ejes y un acelerómetro de tres ejes con lecturas independientes, la sensibilidad se puede ajustar para el acelerómetro en los rangos de ± 2 a ± 8 g y para el magnetómetro $\pm 1,3$ a $\pm 8,1$ gauss, estas señales se adquieren a través de una interface I²C. Esta board incluye regulación de voltaje para que sea más fácil acoplarla con otros sistemas²¹.

El LSM303DLH tiene muchas opciones configurables, incluyendo la sensibilidad que se puede seleccionar de forma dinámica. El magnetómetro y el acelerómetro pueden encenderse o apagarse de manera individual si se quiere aplicar algún control de ahorro de energía, incluyendo funciones de wake up (Activación Automática) con uso de interrupciones.

CARACTERÍSTICAS:

NÚMERO DE EJES	ACELERÓMETRO	3 EJES
	MAGNETÓMETRO	3 EJES
SENSIBILIDAD	ACELERÓMETRO	12 - BITS
	MAGNETÓMETRO	12 - BITS
RANGO	ACELERÓMETRO	$\pm 2, \pm 4, \text{ o } \pm 8$ g.
	MAGNETÓMETRO	$\pm 1.3, \pm 1.9, \pm 2.5, \pm 4.0, \pm 4.7,$ $\pm 5.6, \text{ o } \pm 8.1$ gauss.
VOLTAJE	2.6 - 5.6V	

Tabla 5. Características IMU LSM303DLH.

Fuente: Autor.

²¹ <http://www.pololu.com/catalog/product/2124>

Debido a que el sensor LSM303DLH complementaba era el único disponible en el mercado que incorporaron dos sensores, acelerómetro y magnetómetro se tomo la decisión de utilizarlo en el proyecto.

C. ACTUADORES

Los actuadores de este sistema son los motores Brushless seleccionado basados en parámetros que se nombrarán enseguida teniendo en cuenta que esos motores para su correcto funcionamiento deben trabajar con una serie de dispositivos que también van a ser objeto de estudio en el presente documento.

1. MOTORES

Un motor es una maquina o dispositivo, que produce energía mecánica a partir de energía eléctrica, química u otra, los motores más utilizados en la implementación de maquinas voladoras son los motores Brushless, motores tipo Glow y servo motores.

a. MOTORES BRUSHLESS.

La palabra Brushless se puede traducir como "sin escobillas", las escobillas son los elementos que hacen contacto en el colector de un motor común. En los motores DC más pequeños, son de una aleación de cobre y en motores más grandes son de un compuesto a base de carbón. Estos motores carecen de colector y escobillas o carbones. En vez de funcionar en DC funcionan en AC, la mayoría se alimentan con una señal trifásica, esta señal idealmente debería ser sinusoidal, pero en la práctica son pulsos, haciendo que la señal sea un continua pulsante o bien una continua con mucho componente de AC sin embargo se los clasifica como de DC porque al igual que los motores comunes tienen imanes permanentes.²²

²² http://www.e-radiocontrol.com.ar/?Motores_Brushless

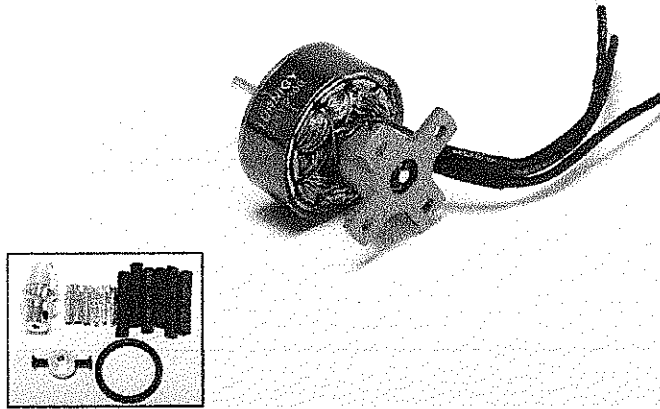


Imagen 26. Motor Brushless Turnigy 2211 Indoor 2300kv.

Tomado de: www.hobbyking.com

Básicamente, hay dos tipos de motores brushless, los inrunner y los outrunner. Los primeros son de más velocidad, su torque máximo lo tienen a muy altas revoluciones, por lo que se usan con cajas reductoras. Los outrunner tienen su torque máximo a baja velocidad, por lo que no necesitan reductoras, van directamente a la hélice.

Los motores brushless en general se encuentran en el mercado de diferentes marcas, precios y potencias.

Aún los más económicos funcionan perfectamente, no tienen desgaste ya que solo constan de un bobinado de varios polos y varios imanes permanentes de alta potencia, normalmente de Neodimio, y dos rodamientos, no tienen fricción de ningún tipo al no tener escobillas (carbones), lo que los hacen casi eternos, hasta se pueden cambiar fácilmente los rulemanes, que podrían llegar a desgastarse o romperse ante un golpe muy fuerte.

Lo que gira en los motores outrunner es la parte exterior, donde están pegados los imanes, quedando en su interior los bobinados fijos, que suelen ser de alambre de cobre grueso, pocas vueltas por bobina, todo muy sólido.

Estos motores trabajan gracias a los variadores de velocidad (speed control), que transforman la corriente continua de las baterías en una tensión alterna trifásica.

Se puede invertir el sentido de rotación fácilmente con solo invertir entre sí dos de los tres cables

que unen al speed con el motor²³.

K_v , de un motor Brushless²⁴:

$K = RPM$, y $v = per\ volt$. Esta constante K_v (Es casi una constante) significa simplemente la cantidad de vueltas RPM (Revoluciones Por Minuto) que da el motor por cada voltio de continua aplicado al ESC (A máxima potencia). Es decir que si a un motor de $1100 K_v$ se le aplica $11,1v$ funcionará a $12210 RPM$ como máximo (Con el ESC se puede disminuir). Esta es su velocidad nominal y nunca subirá más velocidad a no ser que se aumente la diferencia de potencial (Voltaje).

Empuje:

El empuje de un motor Brushless es la fuerza de arrastre que tiene un motor en conjunto con la hélice esta dado por la potencia y K_v del motor. Cuando un motor tiene más potencia, la constante K_v es menor; si el motor es más fuerte (más potente) la velocidad de giro que necesita para dar su máximo empuje es menor, este parámetro está directamente relacionada con las hélices dispuestas en el motor.

Ventajas:

- Mayor eficiencia (menos perdida por calor).
- Mayor rendimiento (mayor duración de las baterías para la misma potencia).
- Menor peso para la misma potencia.
- Requieren menos mantenimiento al no tener escobillas.
- Relación velocidad/par motor es casi una constante.
- Mayor potencia para el mismo tamaño.
- Mejor disipación de calor.
- Rango de velocidad elevado al no tener limitación mecánica.
- Menor ruido electrónico (menos interferencias en otros circuitos).

Desventajas:

²³ <http://www.brushlessmotor.com.ar/#>

²⁴ <http://www.quadruino.com/guia-2/materiales-necesarios-1/motores-brushless>

- Mayor costo de construcción.
- El control es mediante un circuito caro y complejo.
- Siempre hace falta un control electrónico para que funcione (ESC), que a veces duplica el costo.

b. MOTORES TIPO GLOW

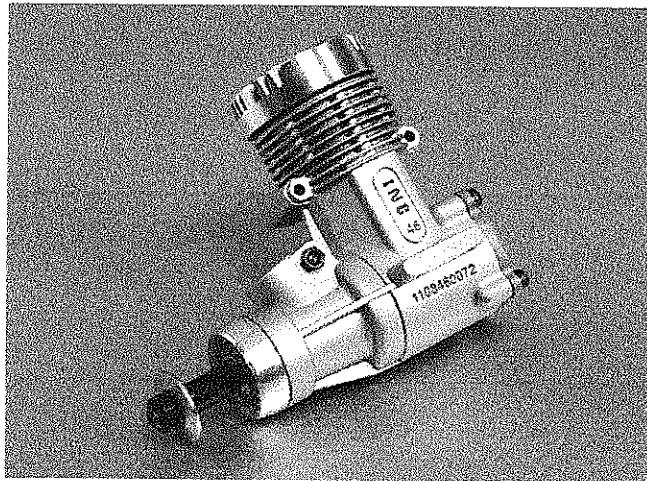


Imagen 27. Motor Tipo Glow INC .46 Glow Engine with Muffler.

Tomado de: www.hobbyking.com

Se denominan Glow a los motores de explosión interna que funcionan con una bujía de incandescencia, en lugar de la conocida bujía de chispa de los motores convencionales de automoción y otros, estos motores se suelen usar habitualmente en Modelismo (Aeromodelismo Automodelismo, etc.) y otros usos similares, incluso para algunos usos militares (aviones espía y otros aparatos similares), su combustible habitual es el Alcohol Metílico (metanol) que se suele mezclar con aceites especiales, para lubricar todas sus partes, y a veces también con algunos aditivos para mejorar sus características²⁵.

²⁵ <http://ellaboratorioloco.wordpress.com/>

c. SERVO MOTOR

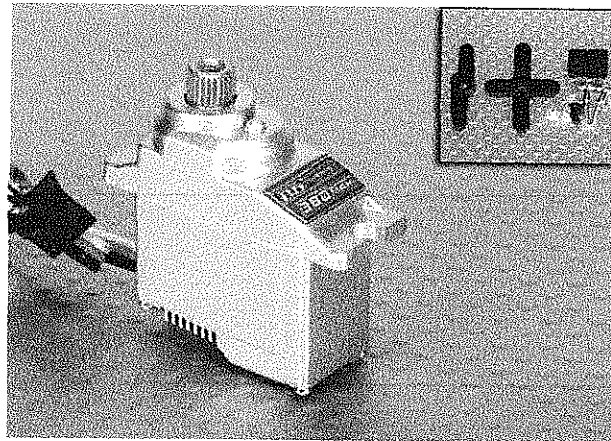


Imagen 28. Turnigy 380MAX Micro Servo (Metal Gear).

Tomado de: www.hobbyking.com

Un Servo es un dispositivo pequeño que tiene un eje de rendimiento controlado. Este puede ser llevado a posiciones angulares específicas al enviar una señal codificada. Con tal de que una señal codificada exista en la línea de entrada, el servo mantendrá la posición angular del engranaje. Cuando la señal codificada cambia, la posición angular de los piñones cambia. En la práctica, se usan servos para posicionar superficies de control como el movimiento de palancas, pequeños ascensores y timones. Ellos también se usan en radio control, títeres, y por supuesto, en robots²⁶.

Los motores seleccionados para el proyecto fueron los motores tipo Brushless debido a sus características de funcionamiento como empuje, velocidad, facilidad de control y rendimiento, el tipo de motor seleccionado es el motor Turnigy D3548/6 790KV Brushless Outrunner Motor (ver Imagen 29).

²⁶ <http://www.todorobot.com.ar/documentos/servomotor.pdf>

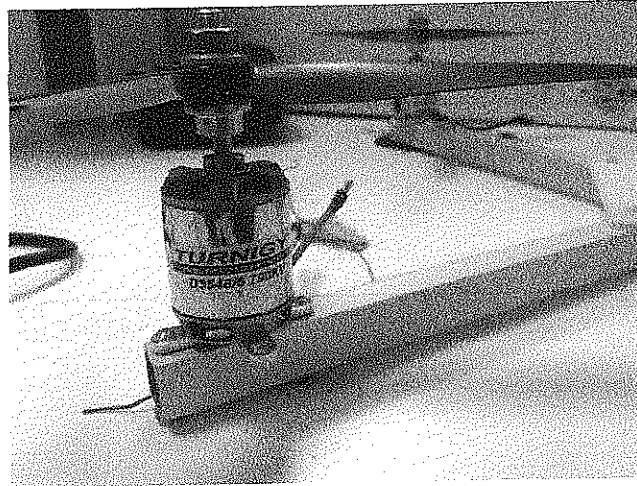


Imagen 29, Brushless Outrunner Motor, Turnigy D3548/6 790KV.

Fuente: Autor.

El motor seleccionado cuenta con las siguientes especificaciones eléctricas y físicas:

Batería soportada	3-5 celdas / 11.1-18.5V
RPM	790 Kv
Máxima Corriente	40A
Corriente Sin Carga	1.8A
Máxima Potencia	715W
Resistencia Interna	0.040 ohm
Peso	159g (incluyendo conectores)
Diámetro del Eje	5mm
Dimensiones	35 x 48mm
Tamaño de Hélice Recomendada	11.1V/12 x 12 18.5/11 x 7
Empuje Máximo	1650g

Tabla 6. Características Eléctricas y Físicas Motor, Turnigy D3548/6 790KV.

Fuente: Autor.

Como se puede apreciar en la tabla 5. El motor seleccionado ofrece un empuje de 1650g lo cual indica que el prototipo de manera teórica teniendo cuatro motores de especificaciones similares ofrece un empuje total de 6600g con la hélice recomendada para la batería de 5 celdas.

D. ACCESORIOS NECESARIOS

Son los dispositivos que son indispensables para la implementación del prototipo como los accesorios para el control de los motores, accesorios para el ensamble de la estructura, hélices y demás accesorios que no son objeto de estudio del proyecto por lo tanto solo se hace una descripción general de cada uno.

1. VARIADORES ESC

Los variadores (ESC) son dispositivos que están encargados de manejar los motores eléctricos, son llamados así por las siglas ESC (Electronic Speed Control), los ESC son utilizados comúnmente en dispositivos de vuelo en el área de aeromodelismo en aviones y helicópteros.

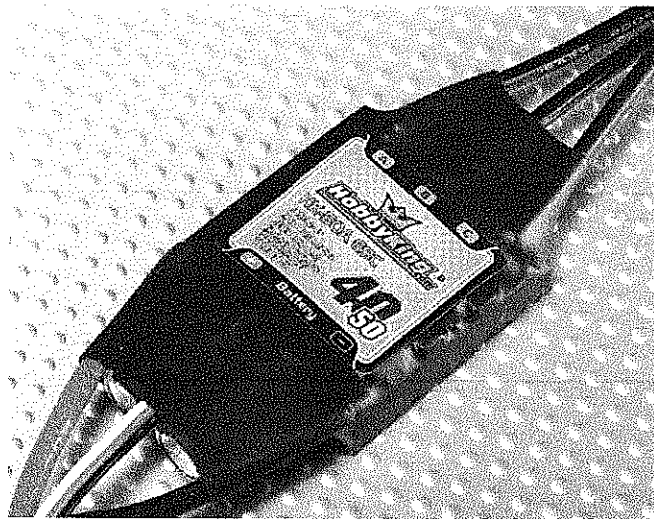


Imagen 30. Hobbyking SS Series 40-50A ESC

Tomado de: www.hobbyking.com

En la Imagen 30, se observa uno de los ESC adquiridos para el proyecto de la marca hobbyKing. Cuando los motores son pequeños de escobillas (Brushed) los ESC actúan como interruptores simples accionados por un canal del receptor de modo ON/OFF. Los ESC más avanzados realizan un manejo PWM, se controlan proporcionalmente enviando un impulso de anchura controlada de esta manera regulan la velocidad del motor.

Los sistemas nombrados anteriormente son poco eficientes dado que los motores Brushed necesitan mantenimiento periódico gracias a sus sistemas mecánico ejercen rozamiento y desgaste de las piezas en la actualidad son más utilizados los motores Brushless, los cuales

cuentan con un bobinado especial similar al de los motores industriales trifásicos, dando la ventaja que ofrecen potencias mucho más elevadas y trabajando a mayor frecuencia. Como particularidad de estos motores resulta que no se pueden conectar directamente a la batería sino que necesitan un circuito electrónico que los maneje como lo es el variador ESC²⁷.

El ESC para motores Brushless tiene una pareja de cables que son la entrada de alimentación proveniente de la batería, tres cables que se dirigen al motor y un cable para conexión al motor o al elemento de control, está compuesto por tres líneas; un cable de señal el cual proviene del receptor contiene la información (ancho de pulso PWM), le indica al ESC la cantidad de potencia que se desea aplicar al motor, otros dos líneas de alimentación una positiva y la otra negativa los cuales generalmente son la alimentación del receptor, servos y otros elementos del conjunto eléctrico del modelo.

Dentro de los variadores se encuentran varios tipos como lo son ESC, ESC+BEC y ESC OPTO.

Los variadores ESC son el tipo nombrados anteriormente, son los más comunes pero no los más útiles, el tipo de variadores ESC + BEC indica que el ESC contiene un BEC interno, un BEC se refiere a Battery Eliminator Circuit y es debido a la necesidad de utilizar una batería independiente para los motores y otra para los servos y el receptor del radio control. Con el ESC + BEC se unifica para que con una sola batería se pueda alimentar los dos servicios. La razón para que sean tan necesarios es que utilizar dos baterías en un prototipo implica más esfuerzo de los motores debido al peso extra por consecuente también el consumo aumenta, resulta que los receptores y los servos trabajan a 5V lo cual no es recomendable tener conectada una batería de 3 celdas las cuales proveen una tensión cercana a los 12V por esto se trabajan estos elementos en conjunto haciendo uso de un ESC + BEC²⁸.

Los variadores ESC – OPTO hacen referencia al termino “optoacoplados”, estos variadores son comúnmente utilizados para realizar el manejo de potencias medias y altas por separado, es decir ofrece la posibilidad de separar el circuito receptor y servos con baterías independientes. Para que el variador pueda acceder a la señal del motor en el receptor incorpora a la entrada del cable de servo un acoplador que funciona por luz (actuando como un mando a distancia en un chip)

²⁷ <http://www.quadruino.com/guia-2/materiales-necesarios-1/esc>

²⁸ <http://www.nitroylitio.com/web/articulos/46-aeromodelismo/53-ique-son-los-esc-y-bec>

permitiendo que los circuitos estén eléctricamente aislados.

Los ESC extraen lo mejor de un motor pero ocurre que algunos variadores ESC necesitan ser programados antes de ser usados. Para ello suelen ofrecer varios métodos, siendo la más simple a través de una secuencia de movimientos de la palanca de gas en la emisora al conectar, pasando por accesorios en forma de tarjeta con botones para las diferentes opciones, hasta los más sofisticados con conexión al ordenador.

Entre los parámetros más habituales que se programaron estuvieron:

Brake: Freno para hélices plegables, sino está activo la hélice sigue girando y no se recoge (por efecto del aire).

Battery Type: Algunos tienen detección automática del tipo de batería (Lipo2S/3S/etc.) y otros hay que indicárselo explícitamente.

Cutoff Mode: Cuando detecta que la batería está baja, indica cómo debe comportarse, reduciendo potencia de motor (preferible) o bien apagándolo completamente.

Cutoff Threshold: Se le indica al ESC el nivel que se considera mínimo de batería, si baja agota la batería. Este parámetro viene predeterminado, es posible subirlo un punto de manera que agote menos la batería (que es bueno para su duración) y en el momento que empiece a bajar el rendimiento del motor, todavía tenga potencia para aterrizar controladamente.

Startup Mode: El modo de arranque, en motores con caja reductora o hélices plegables puede convenir "soft" (suave) para no forzar los piñones y/o darle tiempo a la hélice a que se posicione.

Timing: este es más delicado. Tiene que ver con la frecuencia con que controla el motor. Suelen venir en modo "auto" y no es fácil saber a priori cual es el valor óptimo pues los fabricantes no suelen indicar el valor y la forma de onda que aplica el propio variador también influye, por lo que la mayoría suelen optar por dejarlo en auto si va bien y en modelos de competición o con requerimientos avanzados suelen determinarlo mediante pruebas en banco midiendo los consumos, revoluciones y el empuje estático obtenido²⁹.

El criterio fundamental para la selección de los siguientes dispositivos fue la funcionalidad con

²⁹ <http://tallerdedalo.es/web/ESC>

respecto a los actuadores del proyecto ya que la corriente máxima de los motores es de 40A y los ESC y BEC que seleccione soportan esta corriente, sin dejar a un lado el costo reducido.

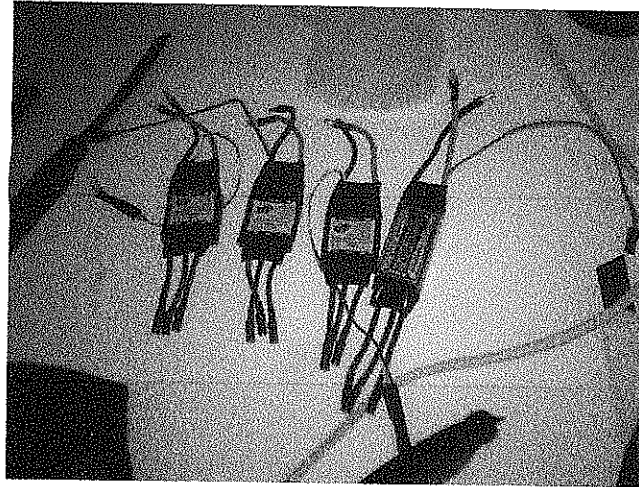


Imagen 31. Tres ESC y un SBEC adquiridos para el proyecto.

Fuente: Autor.

En la Imagen 31, se muestra el ESC de fabricante Hobby King de la serie SS de 6000 kv, tiene incluidas todas las funciones de programación y está diseñado para ser plug and play (listo para su uso), es económico y es fácil de usar³⁰.

Celdas	Soporta de 2 a 4 celdas.
Corriente Máxima	Corriente máxima de 40 A
BEC	NO Incluido.
Timing	1/7/15/30 programable.
PWM	Dese 8 hasta 16K.
Peso	39g
Unidades	4

Tabla 7. Características ESC de fabricante Hobby King de la serie SS de 6000 kv.

Fuente: Autor.

³⁰ http://www.hobbyking.com/hobbyking/store/uh_viewItem.asp?idProduct=6459

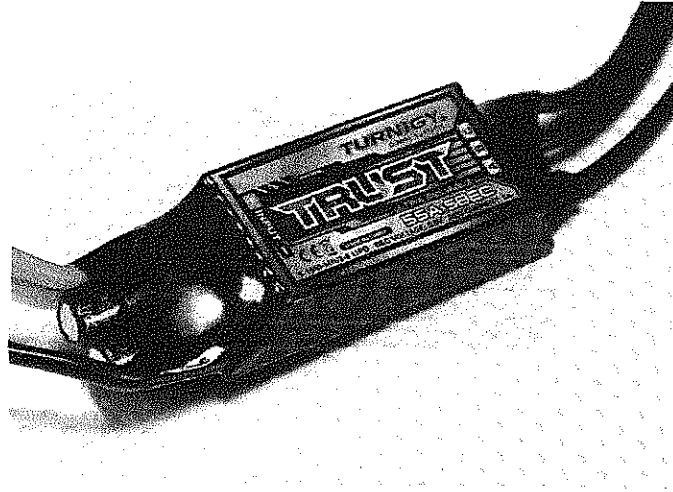


Imagen 32. SBEC 55A marca Turnigy.

Tomado de: www.hobbyking.com

El Turnigy TRUST SBEC ESC tiene un hardware ampliamente capaz y muy flexible, capaz de ejecutar todo tipo de motores sin escobillas (Brushless).

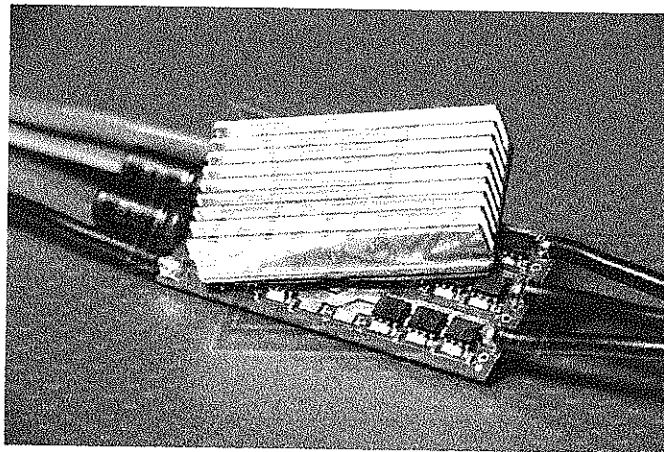


Imagen 33. SBEC 55A marca Turnigy sin cobertura.

Tomado de: www.hobbyking.com

El disipador de calor es de tamaño grande comparado con el circuito de control, es de diseño simple y garantiza una disipación constante, permitiendo al ESC trabajar en su límite sin fallos³¹. Las configuraciones predeterminadas de fábrica se muestran en la Tabla 8 y las características

³¹ http://www.hobbyking.com/hobbyking/store/uh_viewItem.asp?idProduct=12190

eléctricas y físicas se muestran en la Tabla 9.

Freno	Apagado.
Batería	auto detectable.
Baja tensión	3.2V

Tabla 8. Configuración Predeterminada del SBEC 55A marca Turnigy

Fuente: Autor.

Máxima corriente de motor	55A
Máxima corriente de BEC	3A
Voltaje del BEC	5.5v
Celdas de batería	Soporta batería de Lipo de 2 a 6 celdas.
Peso	53g.

Tabla 9. Parámetros Físicos y Eléctricos del SBEC 55A marca Turnigy.

Fuente: Autor.

En la selección de los ESC se decidió agregar a la lista un SBEC ya que los ESC no tienen incorporado BEC por lo tanto no era posible tener una tensión baja de 5V para alimentar el receptor y el microprocesador.

2. HÉLICES

En la actualidad es posible encontrar diferentes tipos de sistemas de propulsión así como aplicaciones. Luego de muchas investigaciones cuyo objeto es la identificación del sistema óptimo en el cual se logre una eficiencia alta, se logre bajar los costos de operación (consumo de energía) sin reducir el desempeño o la vida útil de los sistemas, no se ha podido tener un resultado puntual debido a la gran variedad de especificaciones o características de los motores, condiciones ambientales y demás factores influyentes en el tema de la aerodinámica de los sistemas.

En el área del modelismo, siendo tema intermedio dentro del proyecto se encuentra una serie de recomendaciones y teorías eficientes sobre las cuales se baso la selección de estas hélices

teniendo en cuenta parámetros teóricos y prácticos los cuales luego de ser estudiados dieron como resultado la selección de los mismos.

Dentro del sistema de propulsión más comunes se encuentran las hélices de paso fijo y de paso variable:

a. HÉLICES DE PASO FIJO

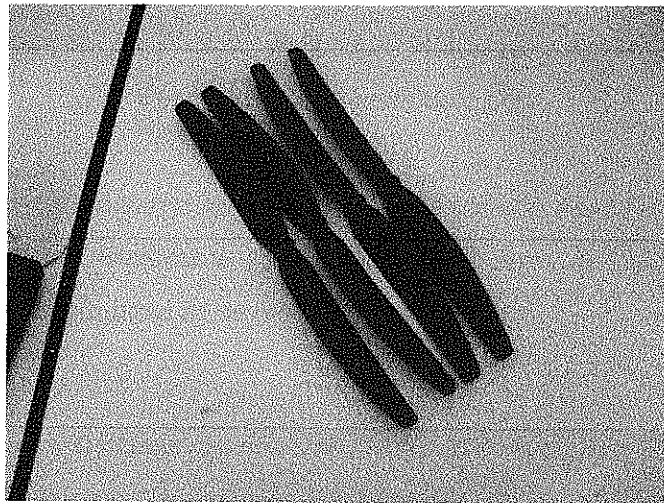


Imagen 34. Hélices de Paso Fijo adquiridas para el proyecto marca GWS 10 X 6.

Fuente: Autor.

La más común debido a su relativo “bajo costo”, es la hélice de paso fijo, estas hélices también se conocen como de “paso constante”, quiere decir que el paso en toda la superficie del aspa (excepto los ángulos de aspa) no cambia y son usadas en la mayoría de los casos donde no se requiera de un control de paso³².

b. HÉLICES DE PASO VARIABLES

Como se mencionó, la mayoría de las hélices tienen un paso constante, pero hay algunas aplicaciones especiales (grandes barcos o embarcaciones de velocidad) donde la necesidad

³² <http://www.ricepropulsion.com/cartas/TNL04/TNL04ESP.htm>

componentes han sido pesados y medidos para lograr una estructura completamente rígida, fue diseñada para quadcopter y permite montajes de motores Brushless de 28 y 35mm, es una estructura probada con 5Kg de peso situados en la mitad de sus ejes y demostrando la rigidez ofreció una mínima flexibilidad, perfecta para este caso.

4. BATERÍAS³⁴

En Aeromodelismo las baterías mas usadas son las baterías LiPo (Lithium Polymer), son una línea evolutiva de las baterías Li-Ion (Lithium Ion) en las que se sustituye el electrolito orgánico por un compuesto solido, son fabricadas de una forma rectangular, en vez de cilíndrico, por lo que al construir un pack de baterías se ahorra el espacio que se desperdicia con las baterías de forma cilíndrica, tienen un peso considerablemente menor que el de las baterías basadas en Níquel, lo que significa un mayor rendimiento en aplicaciones de maquinas voladoras.

Componentes.

Las baterías LiPo suelen llevar un identificador alfanumérico que indica el número de celdas que la componen. Se suele utilizar un número y la letra "S" para identificar el número de celdas conectadas en serie, y un número y la letra "P" para identificar las celdas conectadas en paralelo. Por ejemplo, una batería con el identificador 3S2P indica que la batería está compuesta de seis elementos LiPo, dos grupos conectados en paralelo, formados cada uno de ellos por tres elementos conectados en serie. No obstante, lo más habitual es encontrar packs sin elementos en paralelo, pues se consiguen fabricar tamaños de elementos LiPo cada vez mayores, y no suele ser necesario unir varios en paralelo para conseguir más capacidad.

Capacidad.

Una batería LiPo por lo general tiene una capacidad de almacenamiento de energía directamente relacionada con las dimensiones de la misma (también depende de los materiales utilizados en la fabricación). La capacidad se mide en miliamperios/hora, por lo que es posible encontrar en la pegatina de las baterías este dato con un formato similar a "2200mAh", o "2.2A". En ambos casos se refiere a que el pack tiene una capacidad de 2200 mA/h, esto significa que sería capaz de suministrar 2200 mA (o 2200 mA) durante una hora completa.

³⁴ <http://www.nitroylitio.com/web/articulos/42-electronica/81-baterias-lipo>

Intensidad de Descarga.

Un dato que aparece en todos los packs de baterías LiPo es la intensidad de descarga que admiten ("C's"). Siguiendo la explicación anterior, se dice que una batería que es capaz de suministrar 2200 mA durante una hora, lo hace a $1C$ ($1C =$ "una vez" su capacidad). Una batería marcada como $2C$ permitiría descargarla a dos veces su capacidad, esto es, a 4400mAh (o 4.4Ah). Evidentemente se agotaría al pasar media hora, pues se ha descargado al doble de intensidad. Este es el motivo de que las baterías LiPo tengan una autonomía reducida en aplicaciones de aeromodelismo pues, por ejemplo, una batería de 2200mAh que entrega 40 A/h (18 veces su capacidad, o $18C$) de forma constante al motor de un aeromodelo tan sólo durará unos escasos 3 minutos .

Las baterías seleccionadas para el proyecto son las Turnigy 4000mAh 3S 20C LiPo Pack.



Imagen 38. Turnigy 4000mAh 3S 20C LiPo Pack.

Fuente: www.hobbyking.com

La Imagen 38. muestra la batería usada en el proyecto, debido al tiempo de carga y la necesidad de realizar pruebas o vuelos por un tiempo prolongado se adquirieron dos baterías de este tipo ofreciendo la posibilidad de ser usadas al tiempo en el prototipo o simplemente permanecer como repuesto.

de obtener la máxima eficiencia posible es imperativa.

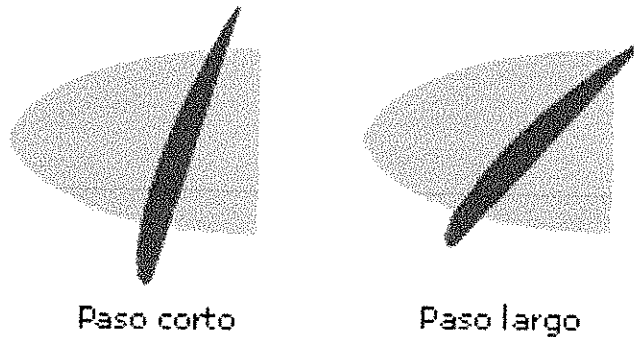


Imagen 35. Posiciones Posibles Hélice paso Variable.

Fuente: www.manualvuelo.com

En estas hélices, el paso puede variar en cada radio (dependiendo del diseño), pero es más común encontrarse aquellas donde usualmente se reduce el paso cerca de las puntas para reducir la presión de las aspas y la posibilidad de cavitación³³.

3. ESTRUCTURA

Estructura hace referencia al armazón donde se ensambla el prototipo, inicialmente dentro del proyecto se propuso la construcción del armazón de perfiles de aluminio situados en forma de X con una longitud de *50cm* cada uno, unidos por unas plataformas de acrílico, pero luego de pruebas de funcionamiento esta estructura presentaba limitaciones debido a su peso, ya que aumentaba en *150 gramos* el peso total del prototipo y esto afectó directamente el empuje de los motores, la velocidad con la que el prototipo se eleva y los niveles de control, dentro de todos los modelos de máquinas voladoras se intenta reducir el peso en todos sus aspectos para así aprovechar de una manera completa los dispuestos en ellos.

³³ <http://www.ricepropulsion.com/cartas/TNL04/TNL04ESP.htm>



Imagen 36. Estructura diseñada y construida para el prototipo.

Fuente: Autor.

La Imagen 36, muestra la estructura inicial diseñada y ensamblada para el prototipo, buscando reducir peso y aumentar la estabilidad se opto por cambiar la estructura por la de la Imagen 37.

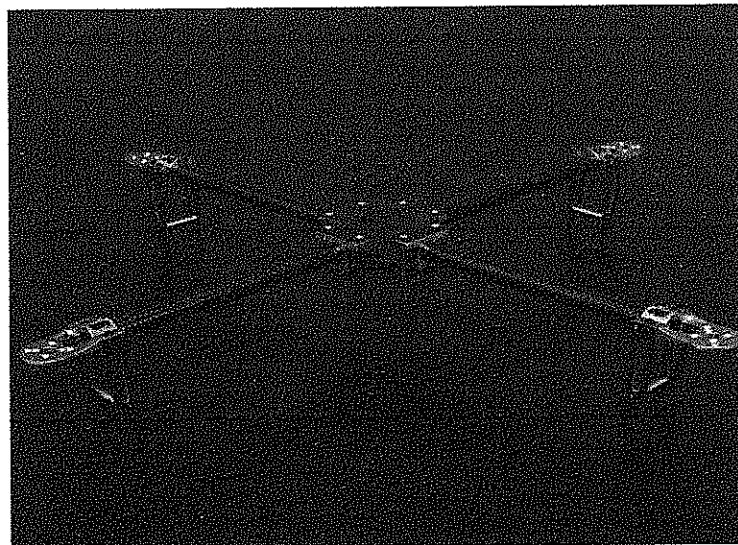


Imagen 37. Estructura Turnigy Talon Quadcopter (2.0).

Tomado de: www.hobbyking.com

En la Imagen 37. Se muestra la estructura definitiva adquirida para el prototipo, esta compuesta por un marco hecho a mano de fibra de carbono genuina de alta calidad que ofrece gran apariencia y rendimiento. La estructura está completamente calibrada es decir todos los

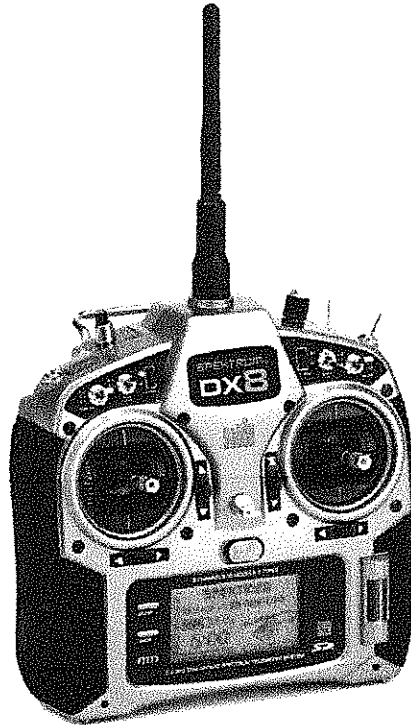


Imagen 39. Equipo de Radio Spektrum DX8.

Fuente: www.horizonhobby.co.uk

En la imagen 39, se muestra el Radio Control adquirido para el proyecto ofreciendo numerosos servicios que no son estudio de este proyecto por lo cual solo se nombran las más relevantes para el desarrollo del proyecto.

- Alta velocidad 11 ms Frame Rate con resolución de 2048.
- 30 modelos en memoria interna.
- Telemetría incluida con alertas.
- 8 canales de transmisión.
- 4 modos seleccionables.
- Tres posiciones de bandera para controles tres estados.

6. OTROS ACCESORIOS

En esta sección se nombran los elementos que hacen parte de la implementación del prototipo y fueron indispensables para su construcción.

Capacidad Mínima	4000mAh
Configuración	3S1P / 11.1V / 3 Cell
Descargas constantes	20C
Descarga Pico	30C
Peso	337g
Dimensiones	151 x 50 x 21 mm
Plug de Carga	JST-XH

Tabla 10. Características Batería Turnigy 4000mAh 3S 20C LiPo Pack.

Fuente: Autor.

La Tabla 9, muestra las características de la batería seleccionada. Fue seleccionada por que los motores que se adquirieron funcionan a $11.1V$ y tienen una corriente pico de $40A$ por lo tanto esta batería suministra la tensión necesaria para que los motores funcionen con su máximo rendimiento.

5. TRANSMISOR DE RADIO FRECUENCIA

Un transmisor de radio frecuencia es un dispositivo capaz de comunicarse con otro dispositivo usando radio frecuencia, el control desde tierra de las máquinas voladoras es indispensable para su manejo y direccionamiento (omitiendo máquinas voladoras autónomas), por esta razón es necesario establecer cómo se va a ejercer el control desde tierra.

Por esta razón se estudiaron diferentes opciones como la comunicación con protocolo ZigBee, módulos de comunicación RF (Radio Frecuencia) con modulación ASK (Amplitude-shift keying), entre otros.

Debido a que establecer la comunicación inalámbrica entre el prototipo y la estación de control en tierra no es objeto de este proyecto se opto por seleccionar un radio transmisor fabricado para este fin sin necesidad de implementar el sistema completo de comunicación.

Primero fue necesario identificar los parámetros de selección de un radio control, se obtuvo que es fundamental el número de canales que maneja, la manera que evita las interferencias y su efectividad; finalmente se tuvo en cuenta los parámetros de configuración ya dispuestos en el radio control como la posibilidad de realizar mezclas, curvas exponenciales y obtener información del modelo a manera de telemetría.

El radio seleccionado fue el serie DX8 de la marca SPEKTRUM, es un equipo de radio de 8 canales con Telemetría integrada.

a. CONECTORES

Cuando se construye un prototipo es necesario que sus partes eléctricas y físicas sean de fácil mantenimiento y por ende que sea cómodo remplazar algún elemento por esta razón se utilizaron conectores PolyMax 3.5mm Gold Connectors, para realizar su ensamble y que resulte cómodo conectar y desconectar los elementos electrónicos en caso de necesitarse.

3.5mm Gold Connector

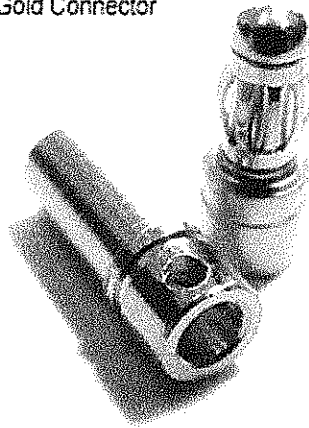


Imagen 40. PolyMax 3.5mm Gold Connectors.

Fuente: www.hobbyking.com.

La Imagen 40 muestra los conectores utilizados para realizar el ensamble de los diferentes dispositivos electrónicos como los ESC, baterías y circuitos de potencia.

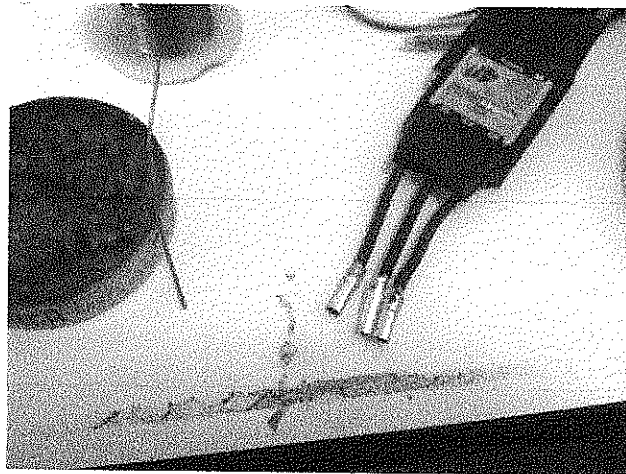


Imagen 41. Ensamble de los ESC con conectores PolyMax 3.5mm Gold.

Fuente: Autor.

b. AMORTIGUACIÓN

Las máquinas voladoras están sujetas a diferentes factores externos como se nombró anteriormente, debido a estos factores y además factores internos surgió la necesidad de aislar elementos para que el movimiento de algunas piezas no afecte el funcionamiento de las demás; por esta razón se usaron varios tipos de materiales para aislar la vibración generada, se utilizaron elementos como espuma, guata, mini-amortiguadores comúnmente usados para asilar la vibraciones de los radios de los carros, configuraciones simétricas de resortes y caucho. Todo esto enfocado a asilar la vibración producida por el prototipo y los factores externos.

c. CINTA

La cinta doble faz 3M hizo parte del proyecto en todo su desarrollo ya que permitió fijar elementos de prueba temporalmente con alto grado de seguridad.

d. TIPO DE CABLE.

Debido a la gran carga de corriente que se estaba manejando fue necesario establecer un circuito de potencia el cual debía responder satisfactoriamente a los momentos en que el prototipo estuviera en su carga máxima, por esta razón las conexiones se realizaron con cable diseñado para manejar una carga superior a los $40A$.

e. CARGADOR DE BATERÍAS.

El cargador de baterías fue un elemento indispensable para el desarrollo del proyecto, debido a que las baterías que se manejaron fue tipo LiPo , era necesario tener un instrumento capaz de cargar y balancear estas baterías cuando su rendimiento empezara a disminuir debido a las altas descargas de tensión que generaban.



Imagen 42. Cargador de Baterías iMax B6 adquirido para el proyecto.

Fuente: Autor.

Se adquirió un cargador de baterías iMax B6, el cual carga y balancea con precisión baterías de polímero de Lito, de NiCd y de NiMH y ofrece la posibilidad de cargar hasta 6s (LiPo/LiFe) muestreando el voltaje de cada celda en tiempo real, además posee conector de carga JST-XH compatible con baterías Zippy, HTX, iMax Entre otras.

Características:

- Voltaje de entrada: 11- 18V.
- Celdas 1 – 15 de Ni-MH/NiCd.
- Celdas 1 -6 de Li-ion/Poly.
- Amplia gama de corrientes de carga.
- Función almacenamiento, permite almacenamiento seguro.
- Función tiempo límite.
- Monitoreo de entrada de voltaje.
- Almacenamiento de datos (almacena hasta 5 paquetes en memoria).
- Carga máxima 50W / descarga máxima: 5W.
- Corriente de carga: 1 – 6.0A.
- Dimensiones 133mm x 87 mm x 33mm.
- Peso: 277g.
- Monitoreo inteligente.

VII. DESARROLLO DEL PROYECTO

El capítulo VII muestra la descripción del desarrollo por temas siguiendo un orden cronológico en el cual se realizó el mismo.

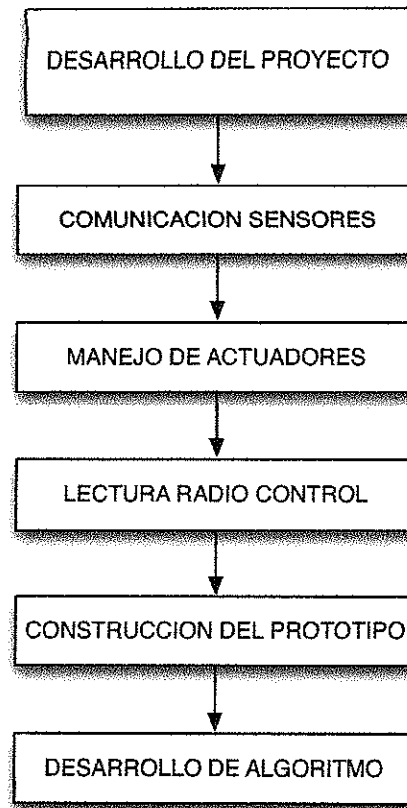


Imagen 43. Flujo del Proyecto.

Fuente: Autor.

El proyecto se desarrolló por áreas de aplicación generales como comunicación con los sensores, recepción de datos del radio control, manejo de actuadores y ensamble del prototipo. Cabe mencionar que estas etapas del proyecto se desarrollaron por separado en el siguiente orden: primero se estableció la comunicación con los sensores, luego se desarrolló la manera de controlar los motores por medio de los variadores (ESC), enseguida se realizó la lectura de datos del receptor de radio control, posteriormente se diseñó, construyó y ensambló la estructura del prototipo y como etapa final se integraron todos estos desarrollos y se vincularon a un algoritmo el cual realizó todas las tareas con el fin de asignarle el control deseado al prototipo.

A. COMUNICACIÓN CON SENSORES.

La comunicación con los sensores es necesaria en el desarrollo del proyecto dado que los sensores son los instrumentos con los que el prototipo va a identificar las variables externas para así ejercer un determinado control en sus actuadores y sus sistemas anexos. Los sensores seleccionados anteriormente, basados en que tuvieran el modelo de comunicación I²C, el cual permite establecer la comunicación bidireccional de los dispositivos con la ventaja de que solo necesita dos líneas de transmisión de datos para su comunicación.

El bus I²C³⁵, es un protocolo de comunicación serie. Su estructura de conexión es muy simple ya que con tan solo dos cables es posible conectar multitud de sensores, controladores o actuadores (en función de la red que se quiera construir) hasta llegar al límite de 400pF.

El bus tiene las siguientes características:

- Requiere dos líneas de comunicación, por un lado SDA (Signal Data) y por otro lado SCL (Signal Clock). Las dos líneas son bidireccionales.
- Ambas líneas de comunicación deben tener conectadas una resistencia pull-up de 10KΩ. Éstas se conectan para provocar que en el momento que no exista comunicación alguna entre los dispositivos que conformen la red, las líneas se encuentren a nivel alto (5V).
- Ambas líneas de comunicación son las que transportan la información a los dispositivos que conforman la red, estos datos están formados por palabras de 8 bits.
- El protocolo de comunicación es Master-Slave.
- Está permitida la conexión de varios masters.
- Todos los dispositivos conectados pueden ser Master o Slave.
- Cada uno de los dispositivos conectados disponen de un código propio para ser reconocido por el resto de dispositivos, por lo que en el momento de iniciar la comunicación, el dispositivo que quiera comunicarse con otro, deberá hacerlo mediante la dirección del dispositivo destinatario.

³⁵ Cap 6. Comunicación I²C, creado por David Álvarez para tallerarduino.wordpress.com

La transmisión bidireccional serie se puede realizar a dos velocidades, el modo más común de funcionamiento es el estándar de 100Kbit/s , esta velocidad también puede llegar a 400Kbit/s en modo rápido o 3.4Mbit/s para aplicaciones industriales.

Una vez montada la red tiene una apariencia como la mostrada en la Imagen 44. En esta imagen es posible observar ambas resistencias de pull-up en las dos líneas de comunicación y la conexión de los diversos dispositivos.

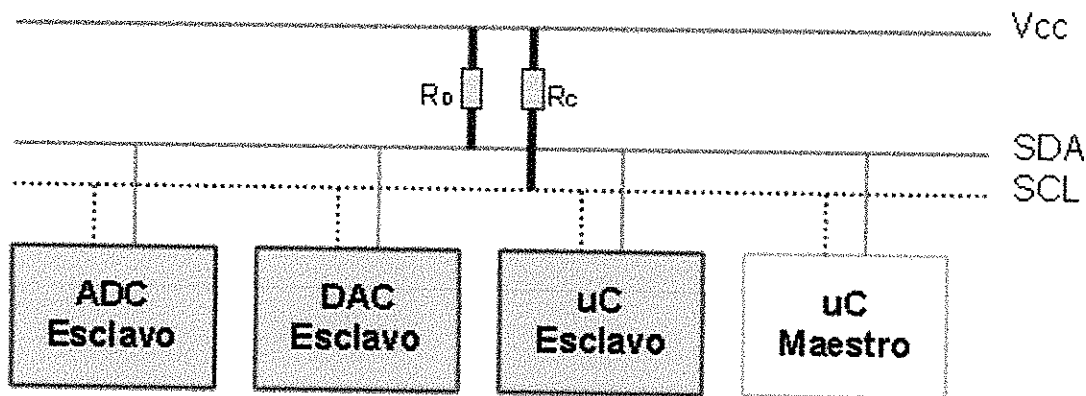


Imagen 44. Esquema de Comunicación I²C

Fuente: Cap 6. Comunicación I²C, creado por David Álvarez para tallerarduino.wordpress.com

Según la Imagen 44, se muestra un momento donde el microcontrolador de la derecha está funcionando como Master, es decir, estaría enviando un mensaje a uno de los otros dispositivos. El Master, tal y como se ha comentado anteriormente, es el dispositivo emisor, es decir, es el que decide a “quien” y “que” información o datos se envía.

Como primera medida se estableció el tipo de dato que entregaban los sensores era digital ya que se iban a trabajar por medio del protocolo I²C, luego de tener esta información se hizo la implementación de los códigos en el entorno de programación del microcontrolador Arduino.

La conexión física que realizada se muestra en la imagen 45.

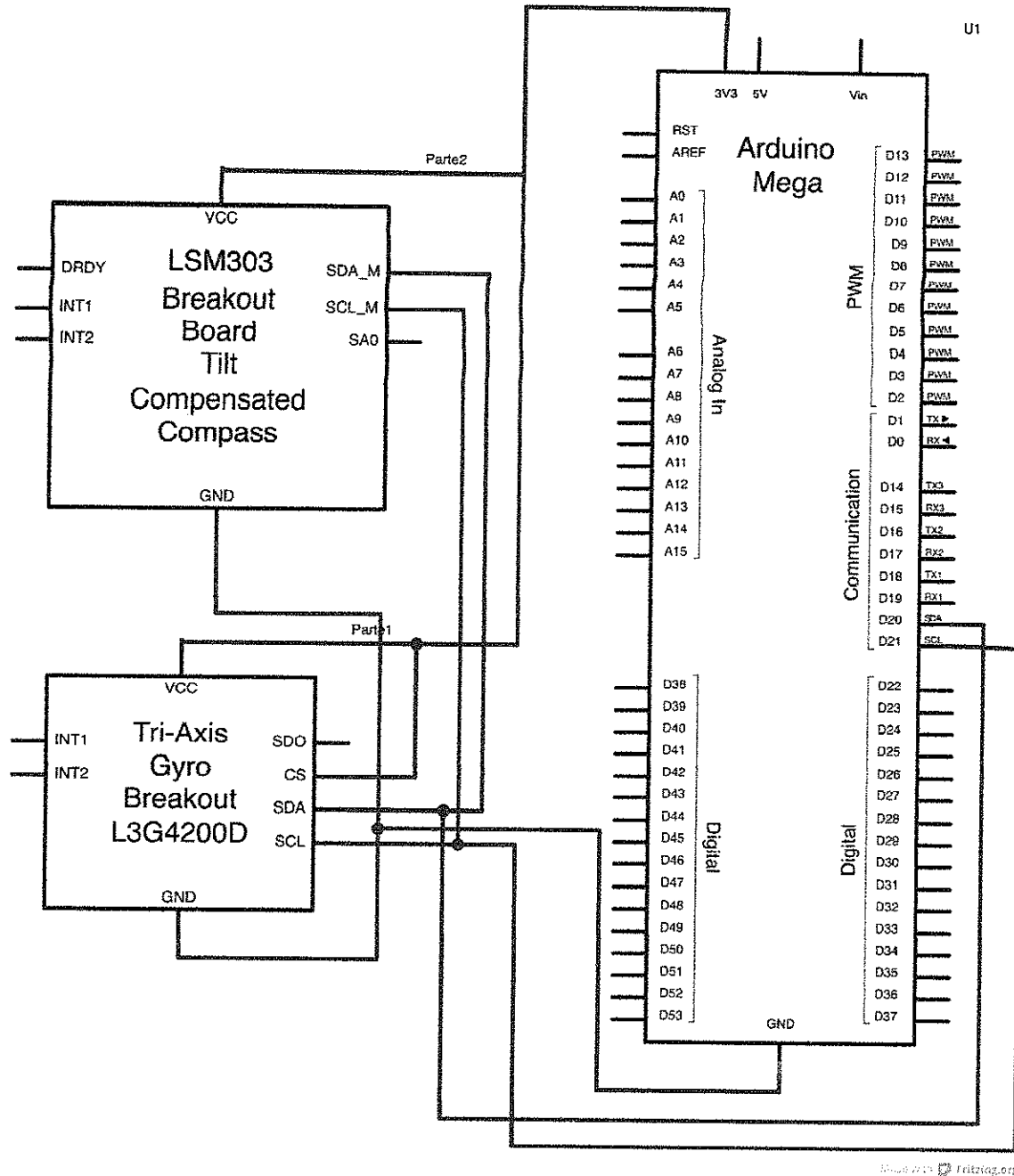


Imagen 45. Conexión Arduino con Sensores L3G4200D y LSM303.

Fuente: Autor.

Como se puede observar en la Imagen 45, los sensores se conectaron en paralelo compartiendo las líneas SDA y SCL, también se puede observar que el pin correspondiente a CS del giróscopo se encuentra conectado a VCC esta conexión se realiza debido a que este es pin habilitador de la comunicación I²C.

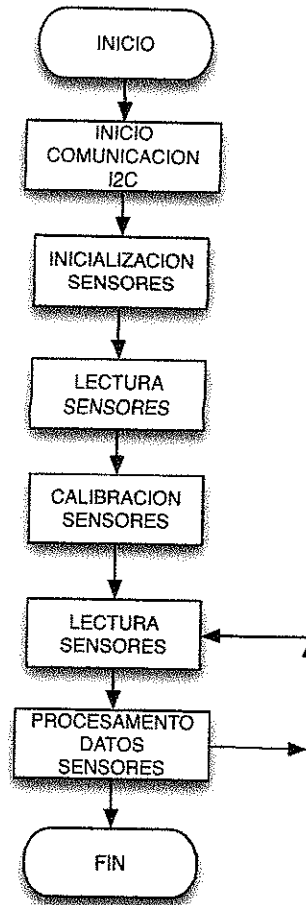


Imagen 46. Flujo del manejo de sensores en el procesador Arduino.

Fuente: Autor.

En la imagen 46, se indica la secuencia por etapas que realiza el procesador para el manejo de los sensores.

Como primera etapa fue realizada la configuración de la comunicación I²C de cada sensor, comenzando por la inicialización de la comunicación mediante la librería WIRE con las dirección de cada sensor, para el sensor L3G4200D la dirección es 110100xb, el LSM303 contiene dos direcciones debido a que son dos sensores dispuestos en la misma board, para el acelerómetro la dirección usada es 001100xb y para el magnetómetro es 0011110xb.

Enseguida es necesario que los sensores se configuren de acuerdo a los requerimientos del prototipo, según la selección de parámetros que se realizó y el estudio profundo de las hojas de datos de cada sensor fue necesario configurar los registros iniciales de cada sensor dentro de la función que inicializa cada sensor envía un parámetro que modifica el registro predefinido de cada sensor, estas funciones de acuerdo con el manejo de las librerías son llamados Gyro_Init()

para la inicialización del sensor giróscopo dentro de los cuales se modifican los parámetros L3G4200D_CTRL_REG1 y L3G4200D_CTRL_REG4 que de acuerdo con el Datasheet se encuentran predefinidos en (20h) y (23h) lo que indica según la distribución de bits de ese registro que se encuentran desactivando la lectura de los tres ejes, lo cual no es ideal para el proyecto ya que no se va a obtener ninguna información de velocidad angular del sensor, además configura de manera predeterminada la escala del sensor asignando 250 dps .

En consecuencia a esto, se realiza la configuración inicial enviando un valor de (0F) y (20) respectivamente configurando el sensor en modo normal, todos los ejes conectados una frecuencia de trabajo de 100 Hz y activando la Full escala dada en 2000 dps .

Posteriormente se realiza la lectura de los datos del sensor giróscopo y son guardados en vectores los cuales almacenan estos datos para su posterior procesamiento.

En el caso de el acelerómetro y el magnetómetro se realizó un proceso similar al del giróscopo, en donde se inicializaron los dos sensores de tal modo que mediante el protocolo I²C se configuren inicialmente modificando los parámetros por defecto y asignando los valores correspondientes para que el sensor acelerómetro trabaje realizando la lectura de los tres ejes y asignándole la escala en la cual va a operar en este caso 8 g llamado también Full escala, luego de la configuración son guardados estos datos en variables las cuales van a ser objeto del procesamiento habitual de las señales de este tipo.

En el sensor magnetómetro fue realizada la configuración de modo que trabaje en modo continuo y al igual que los otros dos sensores guarda estos datos en variables de tipo global para su uso en todo el resto de programa.

Luego de tener lista la configuración de cada sensor, se realizó la lectura de los datos.

Examinando las hojas de datos de los sensores y luego de analizar algoritmos dispuestos en la red de forma gratuita, se logró establecer la comunicación y recepción de datos; los datos de los sensores eran guardados directamente en variables configuradas dentro de las librerías.

Como en el proyecto se manejaron tres sensores de tres ejes cada uno, por lo tanto se tiene 9 variables con datos.

El giróscopo realiza la lectura de velocidad angular de forma análoga, realiza una conversión análogo/digital y entrega los valores medidos en las variables *gyro.g.x*, *gyro.g.y* y *gyro.g.z*.

El acelerómetro y el magnetómetro funcionan de la misma manera que el giróscopo con la diferencia que el acelerómetro realiza la lectura de aceleración lineal y el magnetómetro realiza la

lectura del campo magnético presente en los tres ejes, las variables de almacenamiento son *compass.a.x*, *compass.a.y* y *compass.a.z* para el acelerómetro y *compass.a.x*, *compass.a.y* y *compass.m.x* para el magnetómetro.

Luego de tener estas variables con los datos leídos por los sensores son operadas para alterar su signo con el fin de asignar que sea positivo cuando la rotación fuese de acuerdo a las manecillas del reloj y negativo cuando la rotación fuese en el sentido contrario a las manecillas del reloj haciendo referencia a la siguiente configuración.

- Eje X apuntando hacia delante,
- Eje Y apuntando hacia la derecha.
- Eje Z apuntando hacia abajo.
- Tono (PITCH): nariz hasta.
- Positivo ALABEO (roll): ala derecha hacia abajo.
- Positivo de guiñada (YAW): hacia la izquierda.

Los datos recibidos del giróscopo son de velocidad angular, esto quiere decir que dan lectura de la velocidad instantánea con la cual el sensor gira en cada uno de los ejes teniendo como centro el punto ortogonal en medio de ellos.

La velocidad angular es una variable importante pero no define posición del prototipo, por lo cual fue necesario realizar un procedimiento matemático para que a partir de la velocidad angular se obtuviera la posición angular que es la variable final a controlar.

Al integrar la velocidad da como resultado la posición, por esta razón se realizó la integral en el microcontrolador con la finalidad de obtener la posición (inclinación)

Usando el método de trapezoide para la integración numérica:

$$angulo = \frac{(cambio anterior + cambio actual) \times tiempo de muestreo}{2 \times 1000} \quad (34)$$

de esta manera se realizó el cambio de velocidad angular a posición angular.

Luego de obtener la posición del giróscopo se estudió la posibilidad de implementar el prototipo sin utilizar más sensores, para esto se realizaron pruebas de giro del sensor y la confiabilidad de

los datos, estas medidas fueron realizadas aplicando movimientos controlados al sensor y registrando los datos obtenidos. Se ubicó el sensor en una posición horizontal, se giraba 90° grados y luego se volvía a la posición inicial, indicados por un graduador de utilería unido al sensor. Este procedimiento se realizó varias veces obteniendo los resultados de la Tabla 11.

Numero de movimientos	Posición Final
1	0,3
2	0,2
3	0,5
4	-0,7
5	-0,9
6	-1,6
7	-1,8
8	-3,9
9	5,8
10	0
11	5,2
12	7,9
13	16,5
14	4,7
15	23

Tabla 11. Datos obtenidos en prueba Sensor Giróscopo.

Fuente: Autor.

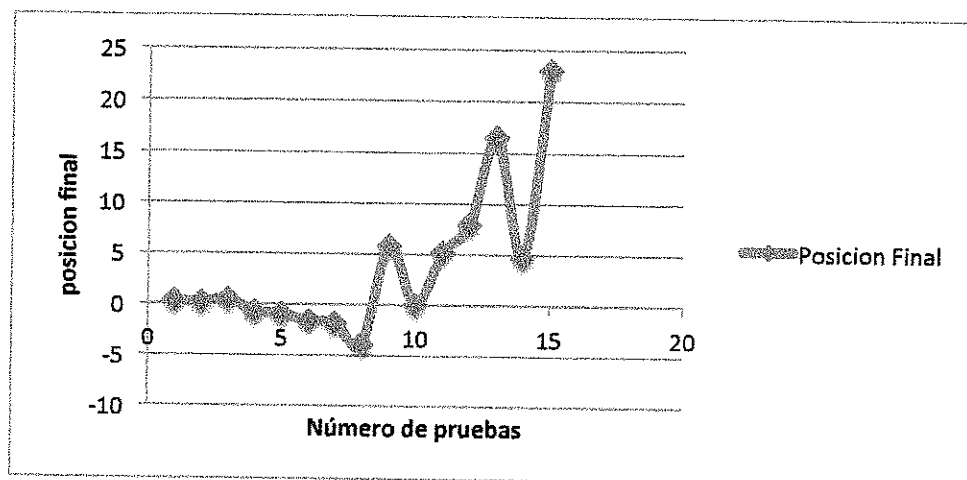


Imagen 47. Deriva de los Datos del Sensor Giróscopo.

Fuente: Autor.

En la Imagen 47, se muestra la notable desviación de los datos de posición leídos, esta desviación es comúnmente llamada deriva (Drift), que hace referencia a la acumulación de errores a través del tiempo lo que indica que luego de un lapso de tiempo el error acumulado al integrar las velocidades angulares es de varios grados.

Por esta razón se descartó la posibilidad de manejar el prototipo con el solo sensor giróscopo.

Pero es necesario tener en cuenta que este sensor tiene la virtud de que lo afectan muy poco los movimientos lineales o vibraciones.

Continuando se vio la necesidad de utilizar otra fuente de información que permitiera corregir la deriva, el proceso de corrección de la deriva es el siguiente:

- Realizar la lectura de los datos de los sensores.
- Calcular el error.
- Tomar el error y operarlo para corregir los desfases y datos erróneos.

Los sensores seleccionados para corregir la deriva fueron un Magnetómetro de tres ejes y un Acelerómetro de tres ejes explicados con detalle anteriormente.

Los magnetómetros son una alternativa perfecta para la corrección de la deriva en el eje Z, está basado en la medición del Campo Magnético Terrestre.

Los acelerómetros son capaces de dar un vector de dirección, fundamentado en la fuerza de gravedad y es perfecto para corregir la deriva de los ejes X y Y .

Luego de las consideraciones anteriores se procedió a desarrollar el algoritmo para conocer los ángulos de vuelo.

Para realizar la interpretación de los datos se trabajo sobre un algoritmo basado en el sistema de referencia de actitud y rumbo AHRS (Attitude and Heading Reference Systems), está formado por un giróscopo, un acelerómetro y un magnetómetro tridimensionales, entregan información de actitud, rumbo y guiñada del prototipo donde estén ubicados.

el objetivo del sistema AHRS es conseguir los ángulos de Euler, estos ángulos conforman un conjunto de tres coordenadas angulares que sirven para especificar la orientación de un sistema de referencia de eles ortogonales, normalmente móvil, respecto a otro sistema de referencia de ejes ortogonales normalmente fijos.

Inicialmente se leen los datos del magnetómetro de 3 ejes.

Enseguida se calculo la inclinación "compensada" X y Y de su componente magnético

(formulación estándar):

$$CMx = mag_x * \cos(pitch) + mag_y * \sin(roll) \sin(pitch) + mag_z * \cos(roll) \sin(pitch) \quad CMy = mag_y * \cos(roll) - mag_z * \sin(roll).$$

Luego es calculada la dirección magnética compensando los componentes:

$$MAG_Heading = \text{Atan}(CMy/CMx)$$

Como resultado *MAG_Heading* era como el indicador de referencia, para poder utilizar las fórmulas de corrección de deriva de guiñado.

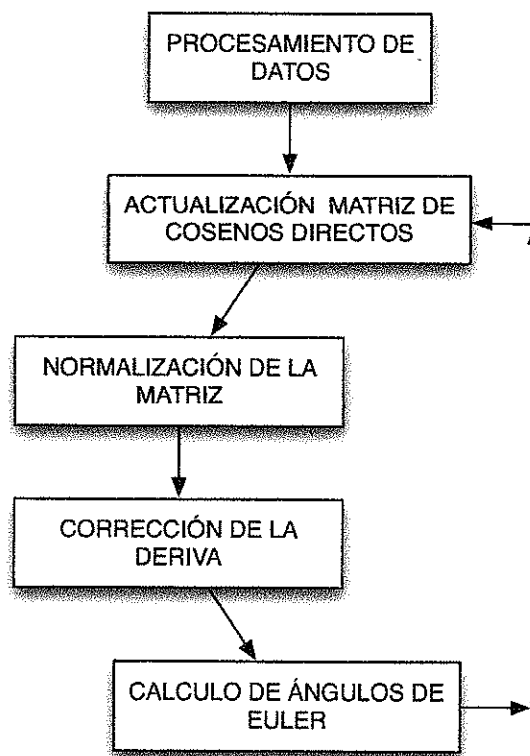


Imagen 48. Flujo del algoritmo para la obtención de los Ángulos de Euler.

Fuente: Autor.

1. ACTUALIZACIÓN MATRIZ DE COSENOS DIRECTOS (DCM)

Es común que para definir una magnitud vectorial se indican sus componentes respecto a algún sistema de referencia, en un espacio tridimensional son usados 3 ejes ortogonales para la ubicación. En el control de aeronaves se pueden utilizar dos sistemas de referencia:

Sistema que utiliza como referencia el suelo y sistema que utiliza como referencia el mismo prototipo.

El primer sistema nombrado se referencia en el suelo lo cual indica que los ejes X y Y forman en el plano la horizontal y el eje z coincide con la fuerza de gravedad.

El segundo sistema utiliza como referencia el mismo prototipo tal y como se muestra en la Imagen 49.

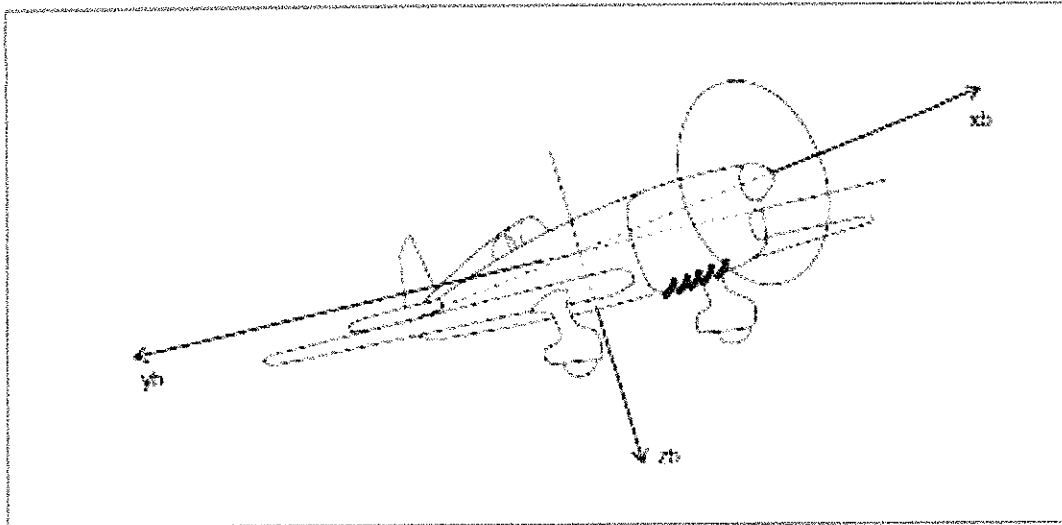


Imagen 49. Sistema de referencia Prototipo.

Fuente: sites.google.com/site/kuadricoptero³⁶

Al estudiar la dinámica de los sensores se tiene que algunos sensores utilizan como referencia el centro del mismo sensor como los son los sensores de magnitudes angulares y otros utilizan como referencia la superficie de la tierra teniendo como eje Z la misma dirección de la fuerza de gravedad.

Por esta razón es necesario utilizar un sistema de referencia común para todos los sensores, el cual establezca que todos se referencien de la misma manera.

La matriz de cosenos directos (DCM) es una matriz de 3×3 corresponde a las componentes de los vectores unitarios del sistema Referencia la Tierra en el sistema Referencia Prototipo. Es decir: los cosenos de los ángulos formados por cada eje del sistema Referencia la Tierra con los ejes del sistema Referencia Prototipo.

Y a la inversa cada columna corresponde a las componentes de los vectores unitarios del

³⁶ <https://sites.google.com/site/kuadricoptero/home/4-dcm>

sistema Referencia la Tierra en el sistema Referencia Prototipo, la matriz DCM tiene la propiedad de que permite transformar cualquier vector del sistema Referencia Prototipo al sistema Referencia la Tierra simplemente con una multiplicación matricial.

En sentido contrario se pueden transformar vectores del sistema sistema Referencia la Tierra al sistema sistema Referencia Prototipo multiplicándolos por la “matriz transpuesta de DCM” (La transpuesta se obtiene cambiando filas por columnas).

Aunque en la matriz DCM se tiene 9 elementos en realidad solo hay 3 variables independientes, por la condición de ortogonalidad y que el módulo de los vectores es 1.

A la matriz DCM también se le denomina Matriz de Rotación, dado que permite transformar vectores entre dos sistemas de referencia que están rotados, una de las propiedades interesantes de las Matrices de Rotación (DCM) es que permiten agrupar varias rotaciones consecutivas en una única multiplicando las matrices de rotación individuales.

Resumiendo el proceso matemático se tiene que en los dos sistemas de referencia coincide la matriz de rotación inicial.

$$R(0) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (35)$$

Teniendo esta información ya es posible actualizar la matriz de cosenos directos con los datos de los sensores previamente guardados en un *vector* [3].

En el proceso de actualización de la matriz de cosenos directos consistió en guardar directamente en vectores las variables de entrada del giróscopo multiplicadas por la constante del conversor ADC del sensor, luego se guardo directamente en vectores las variables correspondientes al acelerómetro.

Como paso siguiente se sumó la constante omega al vector de posición X y se guardo un vector con el fin de asignarle al mismo un término proporcional, posteriormente se realizó el proceso de actualización de la matriz multiplicando esta misma matriz que contiene previamente los datos de los sensores por un tiempo definido como tiempo de integración.

Como etapa final dentro de esta sección se multiplicó la matriz resultante con la matriz DCM (Direct Cosene Matrix) y esta se guardó en una matriz temporal.

2. NORMALIZACIÓN DE LA MATRIZ

Para la normalización de la matriz se crearon tres variables; un vector temporal y dos variables tipo flotante, donde el error es igual al resultado del promedio de las matrices inicial y la matriz final. Teniendo estos parámetros se hizo la normalización de la matriz realizando operaciones matemáticas entre vectores para tener como resultado las matrices renormalizadas.

3. CORRECCIÓN DE LA DERIVA

El proceso más importante y complejo del tratamiento de los datos de los sensores fue la corrección de la deriva, se inició con la declaración de variables como las referencias del sensor magnético en el eje X y el eje Y , una variable para almacenar el error del sistema, y unas variables para realizar la compensación de la deriva de Roll, Pitch y Yaw.

Para realizar la respectiva corrección de la deriva de Roll y Pitch se inicio con el cálculo de la magnitud del vector que contiene la información del acelerómetro, posteriormente se escalizó teniendo en cuenta el factor de gravedad que en el caso corresponde al $1G$ en la tasa de cambio del acelerómetro.

Teniendo la magnitud compensada del vector de aceleración con la gravedad, se realizo el producto cruz entre el vector de aceleración y la matriz de cosenos del giróscopo obteniendo así el error de Pitch y Roll. Luego se realizo el producto del error con el factor de gravedad asignado al acelerómetro. Y finalmente se realiza la actualización de la variable corrección integral nombrada anteriormente.

Para la corrección de la deriva en Yaw se baso en el los datos entregados por el magnetómetro debido a que este se guía por el polo magnético de la tierra estableciendo una referencia en cuanto a dirección. Primero fue necesario identificar el error de la dirección entregada por el magnetómetro operando los componentes X y Y de la matriz por las direcciones en estos mismos ejes entregadas por el magnetómetro obteniendo así el error de las mediciones y para luego aplicar la corrección generada por Yaw en la rotación de los tres ejes, dependiendo de la posición que se encontrara el prototipo.

Finalmente se sumó el error obtenido en este procedimiento al anterior para así ir aplicando la corrección progresivamente es decir a medida que haya un error, es guardado para luego sumarlo

al error presente y no perder las anteriores correcciones en la deriva.

4. CALCULO DE ÁNGULOS DE EULER

Para el cálculo de los ángulos de Euler se realizó las respectivas operaciones para cada eje:

$$Pitch = \sin^{-1}(\text{Matriz DCM [2] [0]}) \quad (36)$$

$$Roll = \tan^{-1}(\text{Matriz DCM [2] [1]}, \text{Matriz DCM [2][2]}) \quad (37)$$

$$Yaw = \tan^{-1}(\text{Matriz DCM [1] [0]}, \text{Matriz DCM [0][0]}) \quad (38)$$

y así se obtuvo las posición del prototipo Pitch, Roll y Yaw.

B. RECEPCIÓN DE DATOS DE RADIO CONTROL

El proceso de recepción de datos de radio control se realizó por etapas iniciando con la identificación los datos que llegan al receptor del radio control situado en el prototipo.

Los radio controles (RC) transmiten la posición de los stick/controles mediante trenes de impulsos, este tren está compuesto por señales que varían su longitud de acuerdo a la posición de cada stick o control dispuesto en el mando enlazado a un canal. Un canal contiene la información correspondiente a su mando y esta información es de tipo PWM, los radio controles como el seleccionado para el desarrollo del proyecto envían impulsos con una frecuencia de 50Hz, es decir cada 20 ms el receptor captura estas señales dividiéndolas por canales y situándolas de tal manera que en el receptor se encuentren en paralelo para ser tomada la señal deseada por separado cabe mencionar que esta frecuencia es la ideal para el control de servomotores. El receptor se puede conectar directamente a los pines de 5v del Arduino.

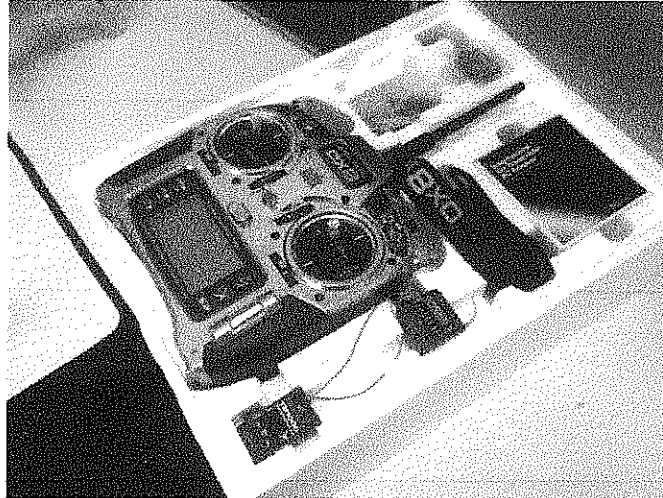


Imagen 50. Radio Control Spektrum DX8, Adquirido Para el Proyecto.

Fuente: Autor.

Para interpretar estos datos con el microprocesador Arduino solo basta medir el ancho de pulso de cada canal y así interpretar la señal recibida, a través de varias pruebas en el laboratorio se identificó el pulso haciendo uso de un osciloscopio el cual confirmó la información y agregó que el voltaje de salida del receptor es de *3.3 voltios*, además se observó que el ancho de pulso mínimo es de *1000 μ S* y el ancho de pulso máximo es de *1920 μ S*, suficiente información para comenzar a trabajar con el microprocesador.

Luego de revisar en las librerías y funciones de Arduino que se encontraban disponibles en la red, se encontró dentro de la configuración de entradas y salidas especializadas, la función *pulsein()*³⁷.

La función *pulsein()* lee un pulso ya sea alto o bajo en un pin; ideal para el proyecto, si se configura para leer un tren de pulsos HIGH la función espera a que la entrada sea HIGH y empieza a cronometrar, cuando el pin cambie a LOW detiene el cronómetro con la medida de tiempo en la cual el pulso estaba en HIGH y entrega el valor de tiempo medido en microsegundos.

Teniendo esta información se realizó la conexión del Arduino con el receptor del radio control.

³⁷ <http://arduino.cc/es/Reference/HomePage>

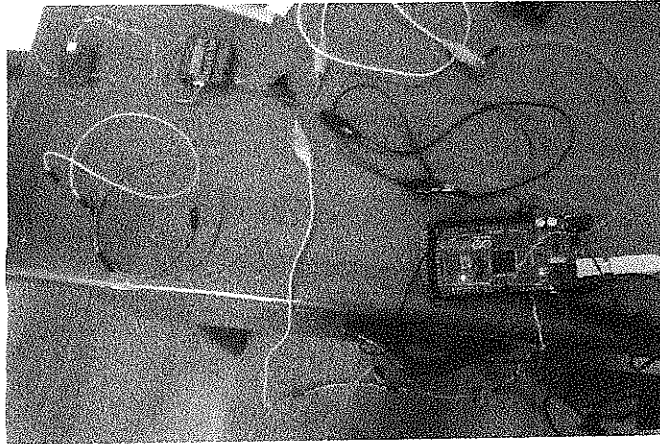


Imagen 51. Conexión Física del Receptor al Microprocesador Arduino.

Fuente: Autor.

Luego de las pruebas realizadas no se presentó ningún inconveniente en la recepción de datos, se obtuvo que el rango de operación de los sticks estaba entre los $1010 \mu S$ y los $1903 \mu S$.

Debido a que la información no podía variar se decidió hacer un recorte de los datos, es decir indicarle al microprocesador que solo leyera un rango de $1100 \mu S$ hasta los $1880 \mu S$, esto se hizo mediante la función *constrain()* la cual restringe una variable a un rango determinado retornando:

x: si x está entre a y b

a: si x es menor que a

b: si x es mayor que b

Posteriormente, para obtener un margen cómodo se realizó el mapeo del mismo dato asignando a la variable una nueva escala, de 0 a 256 mediante la función *map()* la cual re-mapea un número desde un rango hacia otro. Es decir cambia los límites máximos y mínimos a la escala deseada.

Como paso final de este proceso, se incluyó todo en una subrutina llamada *leercontrol()* retornando los datos leídos del receptor, elevación, giro, alabeo y cabeceo.

C. MANEJO DE ACTUADORES

En la etapa de manejo de actuadores es necesario tener clara la manera de controlar los motores.

Dados los actuadores seleccionados se tiene que el control no se ejerce directamente sobre estos, sino que se aplica el control a los inversores ESC (Electronic Speed Control) que se manejan por medio de pulsos PWM los cuales tienen una respuesta directamente proporcional dependiendo del ancho de pulso que le sean aplicado, de tal manera que al aumentar el ancho de pulso al máximo, la velocidad de giro del motor sería la máxima.

Los ESC necesitan una inicialización que es la manera de configurar el motor y el tipo de batería presentes, consiste en recibir un impulso cada 20 milisegundos durante un tiempo cercano a los 5 segundos, cuando el ESC recibe estas señales realiza una indicación sonora la cual manifiesta al usuario que está perfectamente inicializado, la indicación sonora está directamente relacionada con la clave Morse, si la inicialización se llevo a cabo el ESC realiza la siguiente combinación de sonidos: raya, punto, punto, punto(- . . .) la configuración de los punto del código indica el numero de celdas con las cuales está trabajando el variador (ESC).

Considerando que para el Arduino hay disponibles una gran cantidad de librerías era de suponerse que habría una que permitiera realizar el manejo de los actuadores, al revisar las características de funcionamiento de los ESC se observo que se podían manejar con PWM 50Hz de frecuencia que es similar al obtenido del receptor del radio control. Dentro de la referencia de librerías de Arduino se encontró la librería *servo.h*³⁸, diseñada para el manejo de servomotores con la característica que trabaja con la misma frecuencia con la que se quería operar los ESC.

Gracias a que las librerías son de libre manejo se decidió investigar más a fondo las propiedades de esta librería y se dio la posibilidad de editar propiedades de la misma facilitando la creación de una nueva librería *servo1.h* que surgió de modificar los parámetros de funcionamiento de la librería *servo.h*.

Dentro de la librería se encontraban parámetros de funcionamiento como número de motores soportados, frecuencia de funcionamiento, temporizadores utilizados, mínimo y máximo del ancho de pulso entregado y pulso predeterminado.

Se decidió acondicionar esta librería para que funcionara con los parámetros de control de los ESC es decir se modificó el ancho de pulso mínimo y el máximo estableciendo el rango usado anteriormente por la función *constrain()* que era de 1100 uS hasta los 1880 uS, además se estableció que el rango de datos con el cual se iba a manejar la librería no sería de 0 a 179 como

³⁸ <http://arduino.cc/es/Reference/Servo>

es el manejo clásico de los servomotores indicando por cada unidad de aumento se daba un grado más de giro del motor, sino se estableció de 0 a 256 permitiendo manejar los ESC directamente con los datos recibidos del radio control.

Luego de tener lista la librería y la entrada de datos del receptor se probó su funcionamiento.

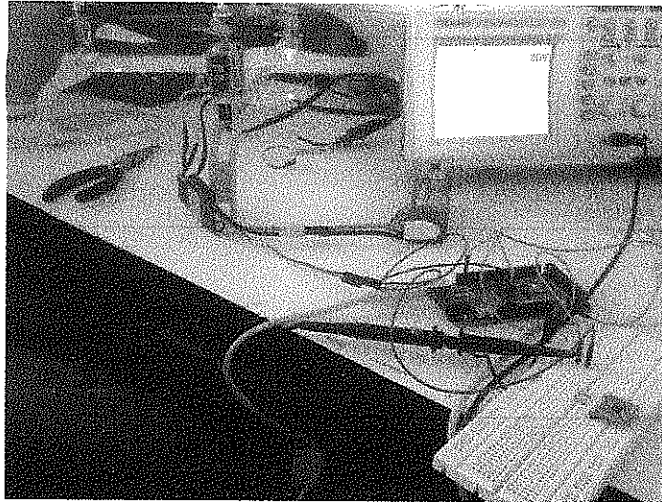


Imagen 52. Primera Prueba Librería Servo1.h con Motor.

Fuente: Autor.

La Imagen 52, muestra la conexión del Arduino con el ESC y éste conectado al motor y a la batería. En el osciloscopio se observa la señal mas baja que entrega el Arduino con la librería *servo1.h*.

La prueba resulto exitosa, el ESC se inicializó y dio como resultado el manejo completo del motor controlado desde el IDE de Arduino entregando 0 RPM con un pulso de $1100 \mu S$ y su máxima Velocidad con $1900 \mu S$.

D. ALGORITMO DE CONTROL

El diseño del algoritmo de control es la unificación de los algoritmos anteriormente desarrollados con la particularidad que debía interactuar en tiempo real con el prototipo, basado en la información proveniente de los sensores era necesario que el microprocesador tomara las mejores decisiones de control sobre los actuadores para que el prototipo se mantuviera en vuelo.

Para establecer el control del sistema se selecciono el método de lógica difusa.

1. LÓGICA DIFUSA

Lógica difusa³⁹ es una lógica alternativa a la lógica clásica que pretende introducir un grado de vaguedad en las cosas que evalúa. Actualmente existe mucho conocimiento ambiguo e impreciso por naturaleza. El razonamiento humano con frecuencia actúa con este tipo de información. *La lógica difusa fue diseñada precisamente para imitar el comportamiento del ser humano.*

La lógica difusa se inició en 1965 por Lotfi A. Zadeh, profesor de la Universidad de California en Berkeley. Surgió como una herramienta importante para el control de sistemas y procesos industriales complejos, así como también para la electrónica de entretenimiento y hogar, sistemas de diagnóstico y otros sistemas expertos.

La lógica difusa en comparación con la lógica convencional permite trabajar con información que no es exacta para poder definir evaluaciones convencionales, contrario con la lógica tradicional que permite trabajar con información definida y precisa.

La lógica difusa se puede aplicar en procesos demasiado complejos cuando no es efectivo un modelo de solución simple o un modelo matemático preciso. Es útil también cuando se necesite usar el conocimiento de un experto que utiliza conceptos ambiguos o imprecisos. De la misma manera se puede aplicar cuando ciertas partes de un sistema a controlar son desconocidas y no pueden medirse de forma confiable y cuando el ajuste de una variable puede producir el desajuste de otras.

Actualmente la lógica difusa tiene un sin número de aplicaciones que afectan nuestra vida cotidiana de alguna u otra manera, pero en ocasiones no es detectable. La lógica difusa se ha desarrollado en diferentes áreas y a continuación se mencionan algunas:

Control de sistemas: Control de tráfico, control de vehículos, control de compuertas en plantas hidroeléctricas, centrales térmicas, control en máquinas lavadoras, control de metros (mejora de su conducción, precisión en las paradas y ahorro de energía), ascensores, etc.

Predicción de terremotos, optimización de horarios.

³⁹ http://catarina.udlap.mx/u_dl_a/tales/documentos/lmt/ramirez_r_o/capitulo3.pdf

Reconocimiento de patrones y Visión por ordenador: Seguimiento de objetos con cámara, reconocimiento de escritura manuscrita, reconocimiento de objetos, compensación de vibraciones en la cámara, sistemas de enfoque automático

Sistemas de información o conocimiento: Bases de datos, sistemas expertos

a. TEORÍA DE CONJUNTOS DIFUSOS

La lógica difusa permite tratar con información que no es exacta o con un alto grado de imprecisión a diferencia de la lógica convencional la cual trabaja con información precisa. El problema principal surge de la poca capacidad de expresión de la lógica clásica.

b. CONJUNTOS CLÁSICOS

Los conjuntos clásicos surgen por la necesidad del ser humano de clasificar objetos y conceptos. Estos conjuntos pueden definirse como un conjunto bien definido de elementos o mediante una función de pertenencia μ que toma valores de 0 ó 1 de un universo en discurso para todos los elementos que pueden o no pertenecer al conjunto. Un conjunto clásico se puede definir con la función de pertenencia mostrada en la ecuación:

$$\mu_A(x) \begin{cases} 0 & \text{si } x \notin A \\ 1 & \text{si } x \in A \end{cases} \quad (39)$$

c. CONJUNTOS DIFUSOS

La necesidad de trabajar con conjuntos difusos surge del hecho que existen conceptos que no tienen límites claros. Un conjunto difuso se encuentra asociado por un valor lingüístico que está definido por una palabra, etiqueta lingüística o adjetivo. En los conjuntos difusos la función de pertenencia puede tomar valores del intervalo entre 0 y 1, y la transición del valor entre cero y uno es gradual y no cambia de manera instantánea como pasa con los conjuntos clásicos. Un conjunto difuso en un universo en discurso puede definirse como lo muestra la ecuación:

$$A = \{ (x, \mu_A(x)) \mid x \in U \} \quad (40)$$

Donde $\mu_A(x)$ es la función de pertenencia de la variable x , y U es el universo en discurso. Cuando mas cerca este la pertenencia del conjunto A al valor de 1, mayor será la pertenencia de la variable x al conjunto A , esto se puede ver en la Imagen 53.

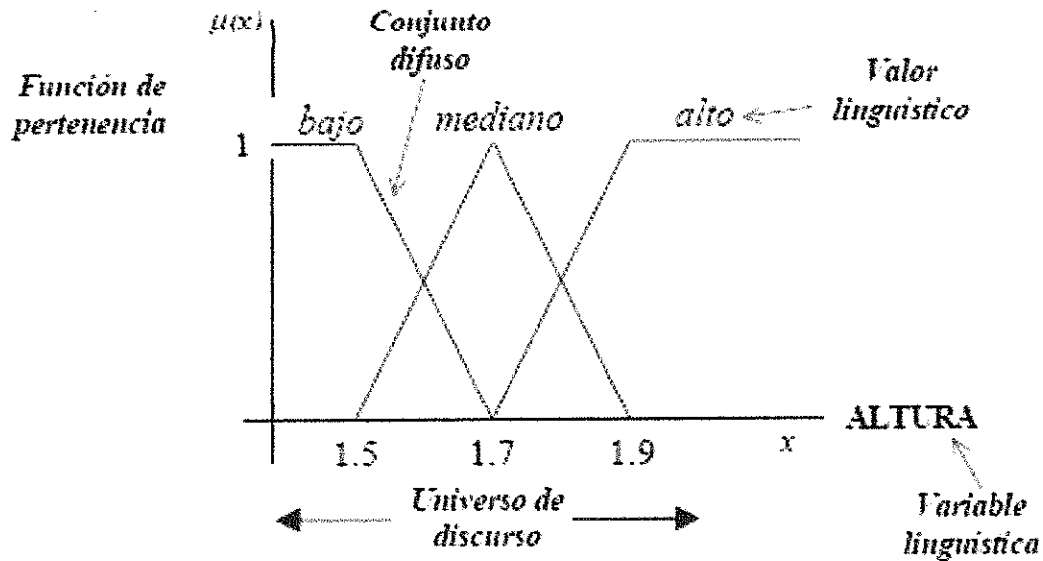


Imagen 53, Conjuntos Lógica Difusa.

Tomado de: Simulación en simmechanics de un sistema de control difuso para el robot udlap, Osvaldo Ramírez Ramos. Universidad de las Américas Puebla.

d. FUNCIONES DE PERTENENCIA

Aun cuando cualquier función puede ser válida para definir un conjunto difuso, existen ciertas funciones que son más comúnmente utilizadas por su simplicidad matemática, entre éstas se encuentran las funciones de tipo triangular, mostrado en la Imagen 54, y tipo trapezoidal como se muestra en la Imagen 55, gaussiana, etc.

Función Triangular.

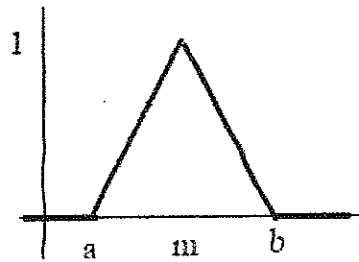


Imagen 54. Función de Pertinencia Triangular.

Tomado de: Simulación en simmechanics de un sistema de control difuso para el robot udlap,
Oswaldo Ramírez Ramos. Universidad de las Américas Puebla.

Cuya función esta expresada por:

$$\text{trapezio}(x; a, b, c) = \begin{cases} 0, & x \leq a \\ \frac{x-a}{m-a}, & a \leq x \leq m \\ \frac{c-x}{c-m}, & m \leq x \leq c \\ 0, & c \leq x \end{cases} \quad (41)$$

Función Trapezoidal.

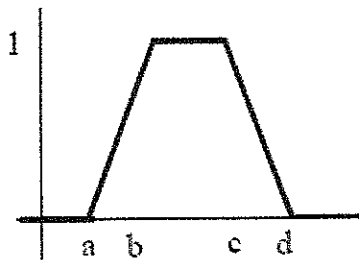


Imagen 55. Función de Pertinencia Trapezoidal.

Tomado de: Simulación en simmechanics de un sistema de control difuso para el robot udlap,
Oswaldo Ramírez Ramos. Universidad de las Américas Puebla.

Cuya función esta expresada por:

$$\text{trapecio}(x; a, b, c, d) = \begin{cases} 0, & x \leq a \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ 1, & a \leq x \leq b \\ \frac{d-x}{b-c}, & c \leq x \leq d \\ 0, & d \leq x \end{cases} \quad (42)$$

2. CONTROL LÓGICO DIFUSO

La estrategia de control utilizada es la que se muestra en la Imagen 56.

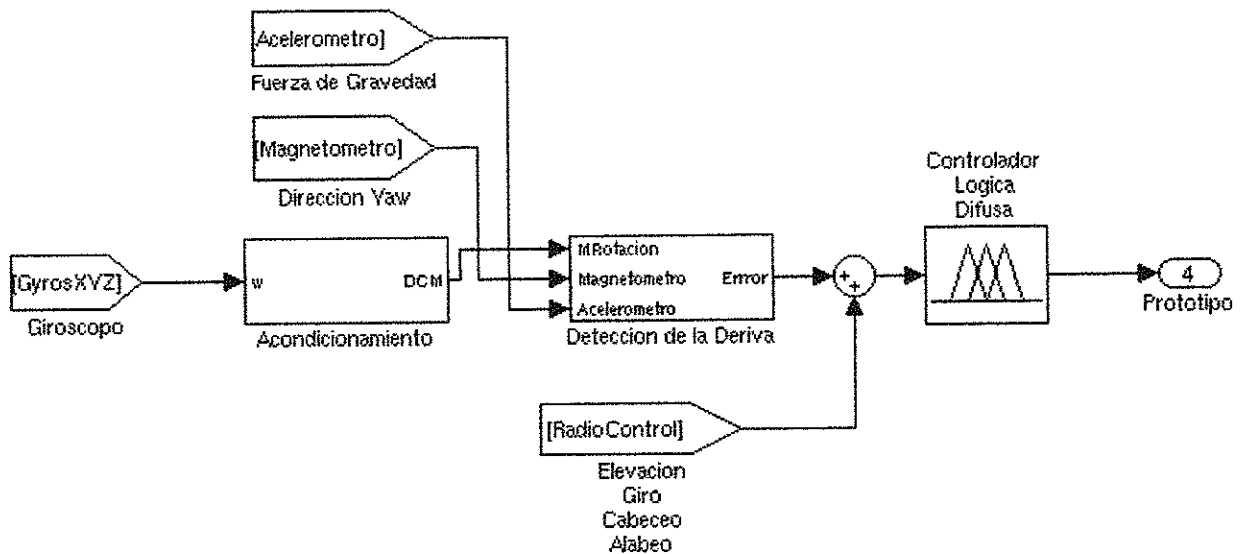


Imagen 56. Estrategia de Control.

Fuente: Autor.

El proceso de control está formado por la etapa de adquisición y tratamiento de datos y la etapa de control donde se reúne toda esta información y se ejecuta una acción para el manejo de los actuadores.

Teniendo esta información clara se realizó el algoritmo planteando los conjuntos de entradas - salidas en el Software Matlab® (ver Imagen 57).

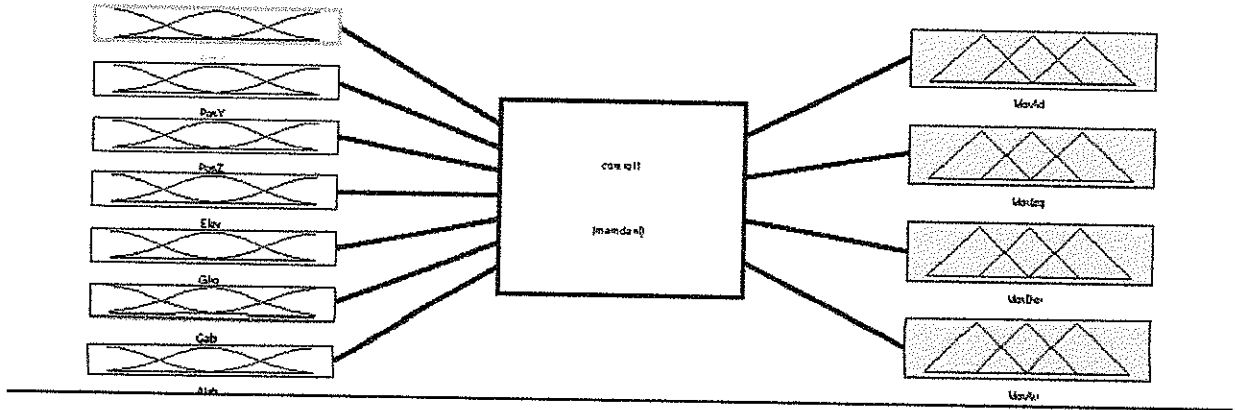


Imagen 57. Control en Toolbox Fuzzy Matlab®.

Fuente: Autor.

Como se puede ver en la Imagen 57, se definieron las entradas salidas del controlador de tal manera que el controlador se basara en 7 entradas y 4 salidas.

Current Variable		Current Membership Function (click on edit to select)	
Name	PosX	Name	trapm
Type	input	Type	trapm
Range	[-25 25]	Params	[35 -25 -20 -5]
Display Range	[-25 25]	<input type="button" value="Edit"/> <input type="button" value="Close"/>	

Selected variable "PosX"

Imagen 58. Conjuntos de Entrada y Salida.

Fuente: Autor.

Como se puede observar en la Imagen 58, se definen los conjuntos de entrada y salida; los conjuntos de entrada son los marcados de color amarillo, en la parte izquierda de la imagen se definieron 7 conjuntos de entrada los cuales pertenecen a las tres entradas de posición X, Y y Z

(Ángulos de Euler), dadas por los sensores y cuatro entradas pertenecientes a las señales recibidas desde el radio control (Elevación, Giro, Cabeceo y Alabeo). También se aprecia en la Imagen 58, las 4 salidas que fueron definidas, correspondientes a los cuatro motores del prototipo.

En los conjuntos difusos se tiene la característica de que la función de pertenencia puede adquirir valores en el rango de 0 a 1, es así como se introduce el concepto de conjunto difuso, el cual se encuentra asociado con un determinado valor lingüístico que está definido por una etiqueta, palabra o adjetivo.

A cada conjunto se le asignó una etiqueta generalmente las 3 o 4 primeras letras de la variable de control; por ejemplo, a la variable de elevación proveniente del stick de elevación del radio control se le llamó “ELEV”, luego de tener definidos los conjuntos se empezó a asignarle a cada conjunto las funciones de membresía.

Para definir este tipo de funciones se identificó el tipo a seleccionar. Para el proyecto se eligió la función trapezoidal, y dentro de cada conjunto se usó 3 funciones de pertenencia definiendo las áreas de acción de cada variable (en caso de las entradas) y la potencia suministrada a los motores (en caso de las salidas).

Los rangos definidos para los conjuntos de entradas eran dos: el primero que contemplaba las entradas correspondientes a los datos del sensor (posición X, Y y Z) cuyos conjuntos fueron llamados PosX, PosY y PosZ, para el conjunto PosX se definió el rango de $(-25, 25)$ tal y como se muestra en la Imagen 59.

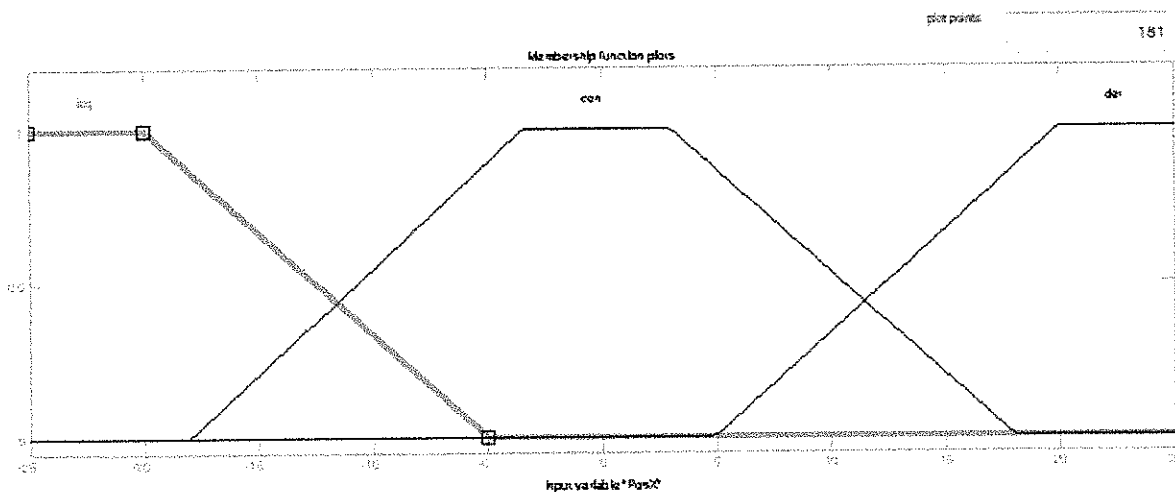


Imagen 59. Funciones de pertenencia PosX.

Fuente: Autor.

En la Imagen 59, se observa como se definieron las funciones de pertenencia del conjunto *PosX*, las etiquetas definidas fueron *izq*, *cen* y *der*, haciendo referencia a la inclinación indicada por el sensor en el eje *X*, si el prototipo se inclina hacia la izquierda, el dato del sensor sería menor a -5, por lo tanto dentro del conjunto *PosX* el valor numérico sería correspondiente a la función de pertenencia *izq*.

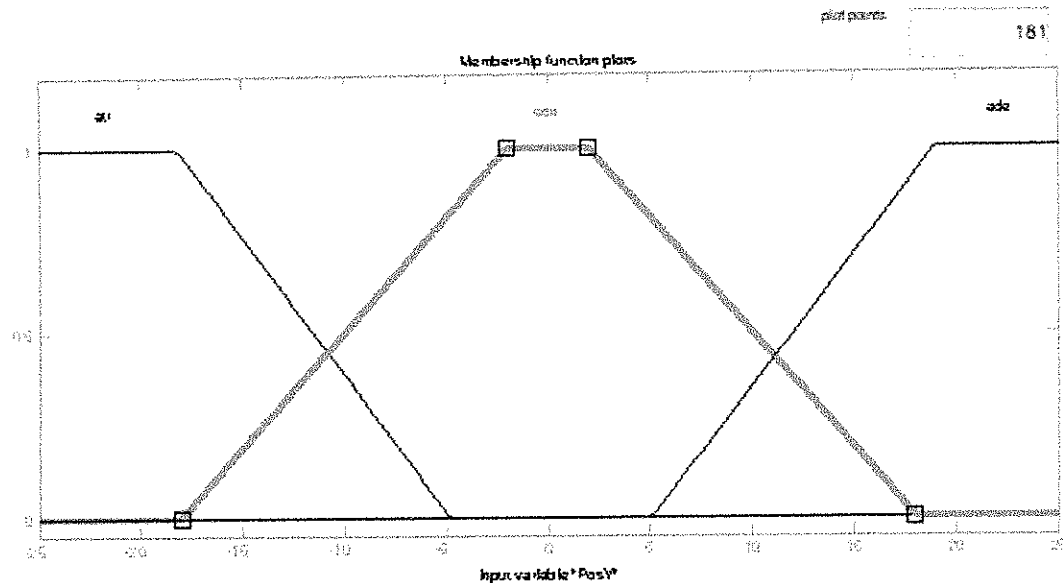


Imagen 60. Funciones de pertenencia *PosY*.

Fuente: Autor.

En la Imagen 60, se observa que para el conjunto *PosY*, se definió el rango de $(-25,25)$, así como las funciones de pertenencia, con etiquetas *atr*, *cen* y *ade*, haciendo referencia a la inclinación indicada por el sensor en el eje *Y*; si el prototipo se inclina hacia adelante, el dato del sensor sería mayor a 5 por lo tanto dentro del conjunto *PosY* el valor numérico sería correspondiente a la función de pertenencia *ade*.

Para el conjunto *PosZ* se definió el rango de $(-180,180)$ indicando la rotación de 360° grados del prototipo, así como las funciones de pertenencia, con etiquetas *izq*, *cen* y *der*, las cuales hacen referencia al giro del sensor en el eje *z*; si el prototipo gira hacia la derecha, el dato del sensor sería menor a 50 por lo tanto dentro del conjunto *PosZ* el valor numérico sería correspondiente a la función de pertenencia *izq*.

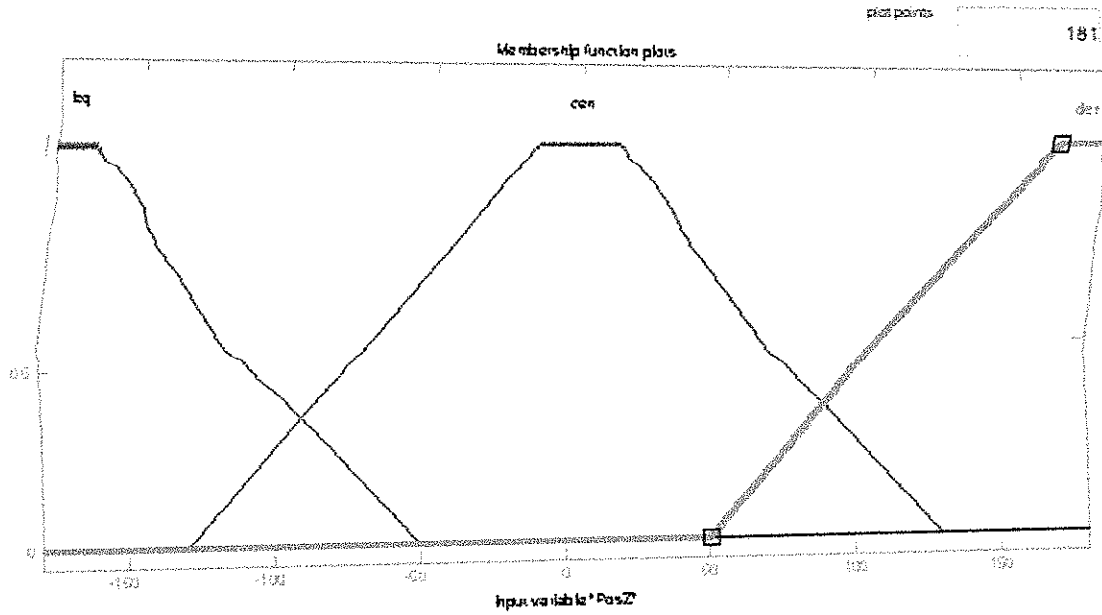


Imagen 61. Funciones de pertenencia *PosZ*.

Fuente: Autor.

Para los siguientes 4 conjuntos de entrada correspondientes a las variables recibidas del radio control, se definió el rango de (0, 255) debido a que los datos fueron procesados, especificando esta escala como el rango de salida.

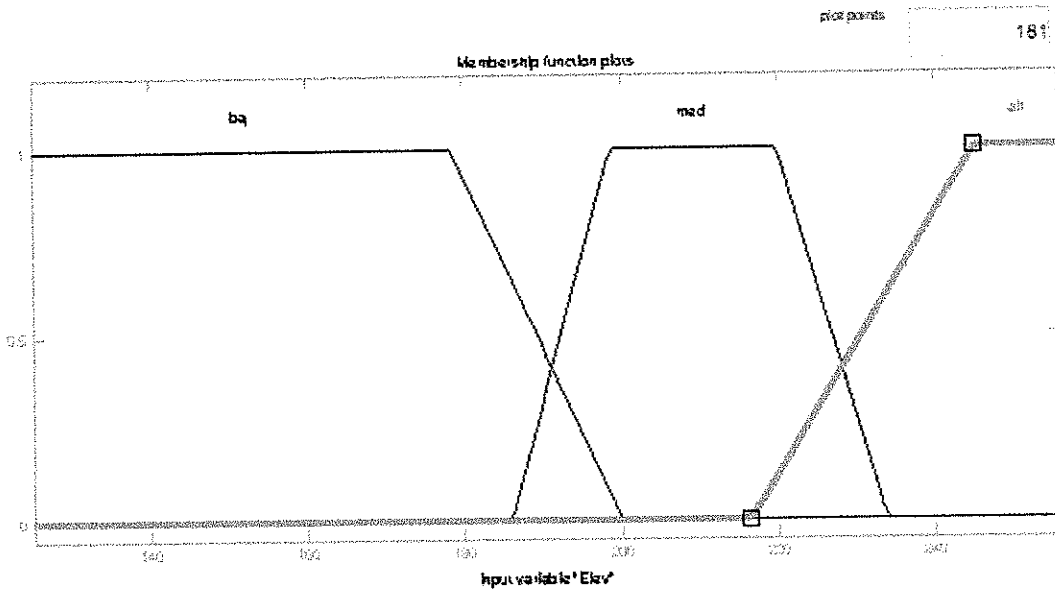


Imagen 62. Funciones de pertenencia *elev*.

Fuente: Autor.

El conjunto *elev* es particular debido a que está directamente relacionado con el stick de elevación del radio control. Se definieron las funciones de pertenencia *baj*, *med* y *alt*, que hacen referencia a la aceleración baja, media y alta reflejadas en los motores. En la Imagen 62, se observa que la función de pertenencia *baj* tiene un intervalo mayor, ya que es necesario que el stick de elevación de aceleración demuestre cambios mínimos en la iniciación de los motores, es decir, en el momento de acelerar, la variación de la potencia de los motores sea baja, con el fin de evitar accidentes reduciendo la posibilidad de que se inicie un vuelo brusco fuera de control.

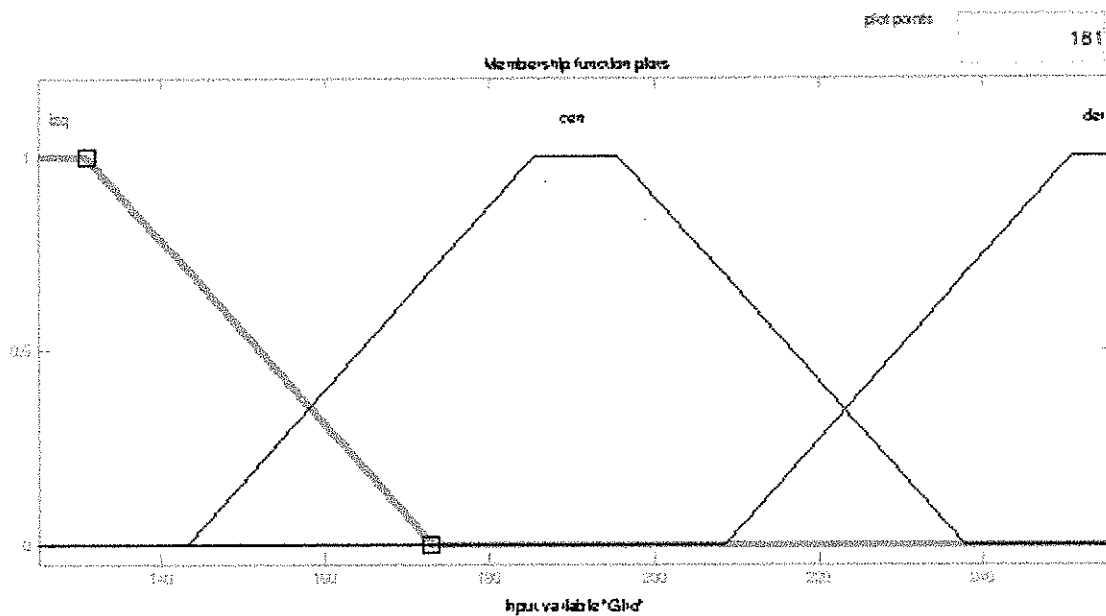


Imagen 63. Funciones de pertenencia *giro*.

Fuente: Autor.

En la Imagen 63, se muestran las funciones de pertenencia del conjunto *giro* relacionadas directamente con el stick de giro del radio control, las etiquetas de las funciones son *izq*, *cen* y *der*, que indican la posición hacia la cual el prototipo debe apuntar, ya que el conjunto *giro* depende de la información de rumbo proporcionada por el eje Z del sensor. La definición de intervalos es proporcional en las 3 funciones de pertenencia, el controlador se guía por el punto medio del intervalo general, el cual corresponde a la función de pertenencia *cen*.

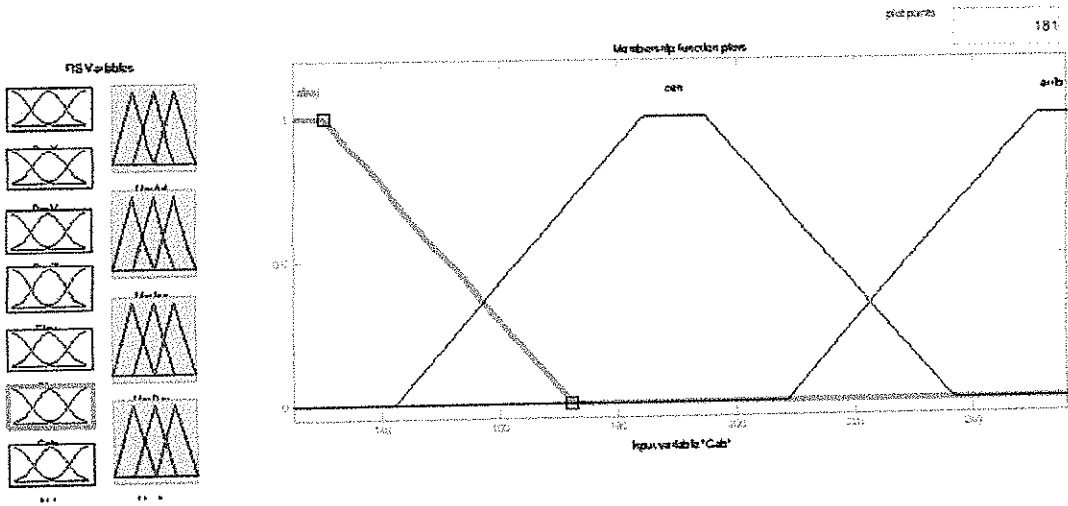


Imagen 64. Funciones de Pertenencia *cab*.

Fuente: Autor.

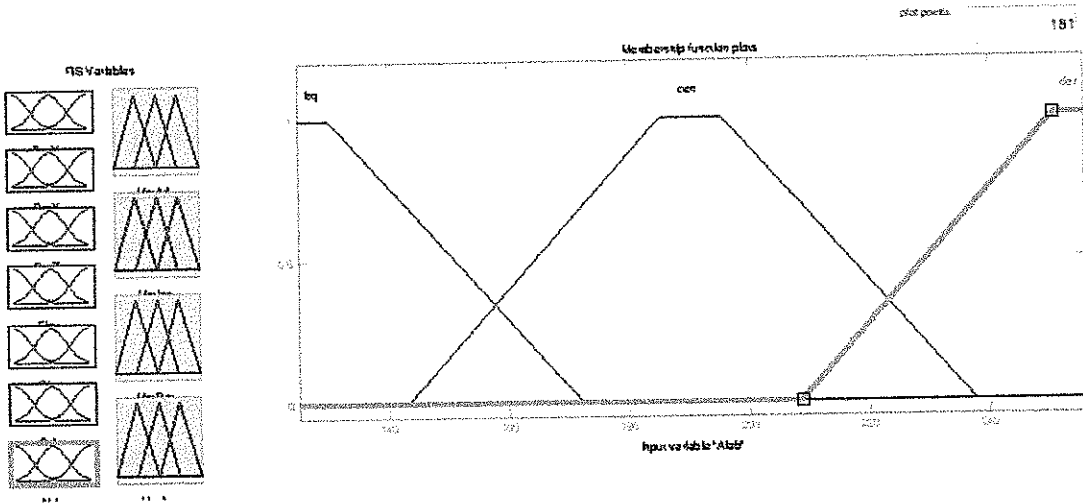


Imagen 65. Funciones de Pertenencia *alab*.

Fuente: Autor.

En las Imágenes 64 y 65, se observan las funciones de pertenencia *alab* y *cab*, correspondientes al alabeo y cabeceo del prototipo con un rango definido de $(0,255)$, estas funciones solo difieren en el concepto del movimiento ya que en el alabeo los movimientos son *der* (derecha) e *izq* (izquierda), en la parte central o nivelado se llama *cent* (centro) y en el cabeceo los movimientos y sus funciones de pertenencia son *ade* (adelante), *cen*(centro) y *atr* (atrás).

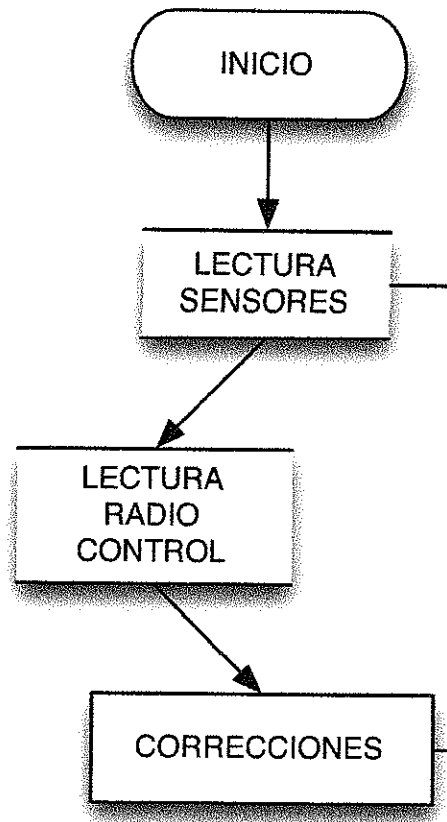


Imagen 67. Flujo de Control del Prototipo.

Fuente: Autor.

En la etapa de correcciones, el controlador envía un incremento o un decremento mínimo del PWM que controla los actuadores y luego revisa si todavía está presente el error, si el error está presente, el controlador realiza nuevamente la acción de control.

La definición de los conjuntos en el microcontrolador se realizó mediante decisiones de tipo IF como se observa a continuación.

```

if ((Promediox==0) && (Promedioy==0)){ // función de lógica difusa
  myservo1.write(elev + n);           //
  myservo2.write(elev + n);           //
  myservo3.write(elev + n);           //
  myservo4.write(elev + n);           //
}
  
```

```
}

```

mediante la función IF se definió la potencia que se debe entregar a los motores en caso de que la posición en X y la posición en Y sean cero, la variable elev (entrada directa del stick de aceleración) fue el punto de referencia de la velocidad de los motores por lo tanto sí la posición del prototipo es estable la potencia de los motores es directamente proporcional al movimiento del stick.

Los conjuntos se definieron por intervalos de posición de tal manera que solo se aplicaron tres conjuntos en este caso; el conjunto de los valores de posición menores que cero, los mayores que cero y los iguales que cero.

La segunda condición asigna la potencia a los motores teniendo en cuenta, si la posición en Y es mayor a cero; es decir si el prototipo se encuentra inclinado hacia la derecha la potencia del motor contrario va a disminuir.

```
if ((PromedioX==0) && (PromedioY>0)){
  myservo1.write(elev + n);//
  myservo2.write(elev + n);
  myservo3.write(elev + n);//
  myservo4.write(elev - n);
}
```

La tercera condición indica que el prototipo esta inclinado hacia la izquierda por lo tanto el motor de la derecha disminuye su velocidad para llegar a una estabilidad.

```
if ((PromedioX==0) && (PromedioY<0)){
  myservo1.write(elev + n);//
  myservo2.write(elev - n);
  myservo3.write(elev + n);//
  myservo4.write(elev + n);
}
```

La cuarta y quinta condición están en función al eje X, si el prototipo se encuentra inclinado hacia delante el motor de atrás disminuye la potencia.

```
if ((PromedioX>0) && (PromedioY==0)){  
  myservo1.write(elev + n);  
  myservo2.write(elev + n);//  
  myservo3.write(elev - n);  
  myservo4.write(elev + n);//  
}
```

Si el prototipo se encuentra inclinado hacia delante el motor de atrás disminuye la potencia.

```
if ((PromedioX<0) && (PromedioY==0)){  
  myservo1.write(elev - n);  
  myservo2.write(elev + n);//  
  myservo3.write(elev + n);  
  myservo4.write(elev + n);//  
}
```

Si el prototipo se encuentra inclinado hacia la diagonal inferior izquierda, los motores de su diagonal opuesta disminuyen la velocidad.

```
if ((PromedioX<0) && (PromedioY<0)){  
  myservo1.write(elev - n);  
  myservo2.write(elev - n);  
  myservo3.write(elev + n);  
  myservo4.write(elev + n);  
}
```

Si el prototipo se encuentra inclinado hacia la diagonal inferior derecha, los motores de su diagonal opuesta disminuyen la velocidad.

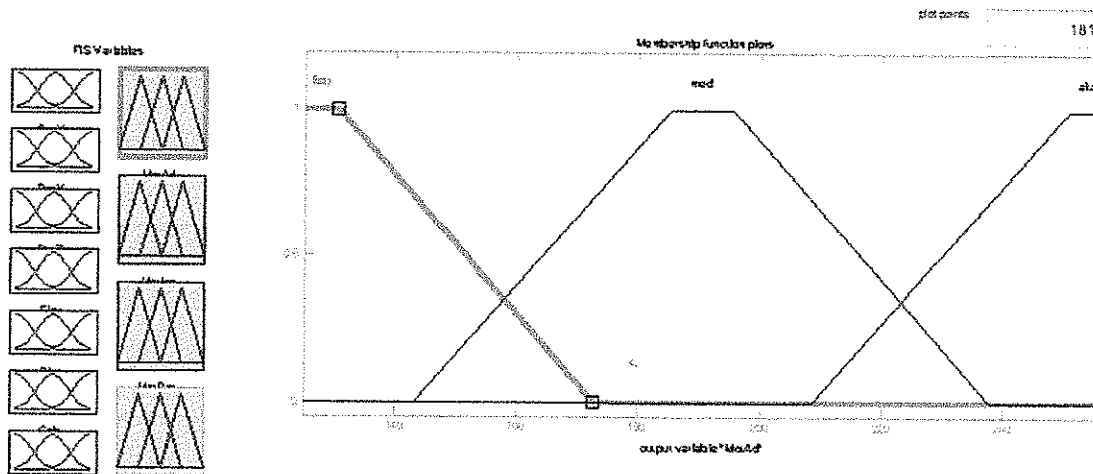


Imagen 66. Funciones de Pertinencia *MotAd*.

Fuente: Autor.

Como los motores son 4, a cada uno se le asignó un conjunto de salida para realizar el control de potencia por separado, los rangos definidos son iguales en los 4 motores de (0,255), por esta razón solo se profundiza en uno (*MotAd*), el cual hace referencia al motor situado en la parte delantera del prototipo, los motores de la parte izquierda y derecha se llamaron *MotIzq* y *MotDer* respectivamente y el motor trasero se denominó *MotAtr*. Las funciones de pertenencia se llamaron (*baj*, *med* y *alt*).

Debido a que los conjuntos de entrada son 7 y los conjuntos de salida 4, cada uno teniendo tres estados posibles, resultó demasiado extenso el desarrollo de las condiciones del controlador por esta razón se decidió tomar solo las decisiones más importantes, modificando el controlador para que actuara de tal manera.

Las reglas o condiciones más importantes vinculan a un solo stick del radio control que es el de Elevación, indicándole al controlador el punto mínimo de potencia que se le va a suministrar a los actuadores. Esto quiere decir que el controlador actúa de la manera mostrada en la Imagen 67.

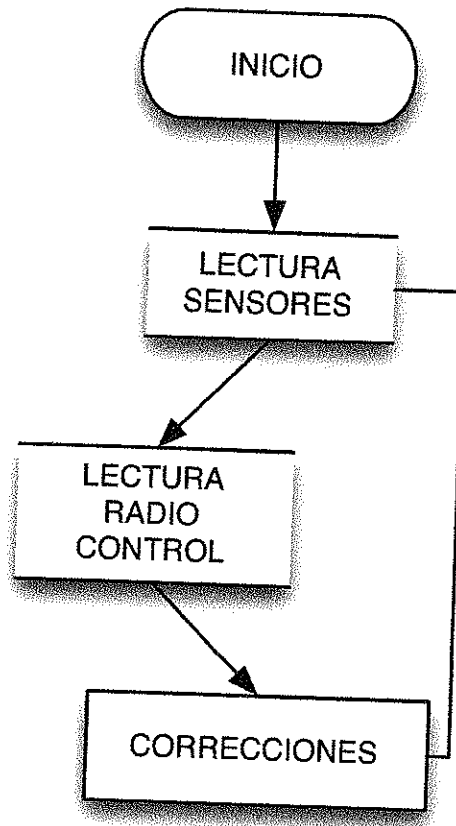


Imagen 67. Flujo de Control del Prototipo.

Fuente: Autor.

En la etapa de correcciones, el controlador envía un incremento o un decremento mínimo del PWM que controla los actuadores y luego revisa si todavía está presente el error, si el error está presente, el controlador realiza nuevamente la acción de control.

La definición de los conjuntos en el microcontrolador se realizó mediante decisiones de tipo IF como se observa a continuación.

```

if ((Promediox==0) && (Promedioy==0)) { // función de lógica difusa
  myservo1.write(elev + n);           //
  myservo2.write(elev + n);           //
  myservo3.write(elev + n);           //
  myservo4.write(elev + n);           //
}
  
```

```
}

```

mediante la función IF se definió la potencia que se debe entregar a los motores en caso de que la posición en X y la posición en Y sean cero, la variable elev (entrada directa del stick de aceleración) fue el punto de referencia de la velocidad de los motores por lo tanto si la posición del prototipo es estable la potencia de los motores es directamente proporcional al movimiento del stick.

Los conjuntos se definieron por intervalos de posición de tal manera que solo se aplicaron tres conjuntos en este caso; el conjunto de los valores de posición menores que cero, los mayores que cero y los iguales que cero.

La segunda condición asigna la potencia a los motores teniendo en cuenta, si la posición en Y es mayor a cero; es decir si el prototipo se encuentra inclinado hacia la derecha la potencia del motor contrario va a disminuir.

```
if ((PromedioX==0) && (PromedioY>0)){
  myservo1.write(elev + n);//
  myservo2.write(elev + n);
  myservo3.write(elev + n);//
  myservo4.write(elev - n);
}
```

La tercera condición indica que el prototipo esta inclinado hacia la izquierda por lo tanto el motor de la derecha disminuye su velocidad para llegar a una estabilidad.

```
if ((PromedioX==0) && (PromedioY<0)){
  myservo1.write(elev + n);//
  myservo2.write(elev - n);
  myservo3.write(elev + n);//
  myservo4.write(elev + n);
}
```

3. CONSTRUCCIÓN DEL PROTOTIPO Y PRUEBAS DE FUNCIONAMIENTO

La construcción del prototipo se inicio simultáneamente con el manejo de actuadores ya que era necesario realizar pruebas con los motores estables, sujetos a una plataforma.

El prototipo se construyó con dos perfiles de aluminio de 50 cm de largo, superpuestos en forma de cruz dando ángulos de 90° entre ellos (ver Imagen 68).

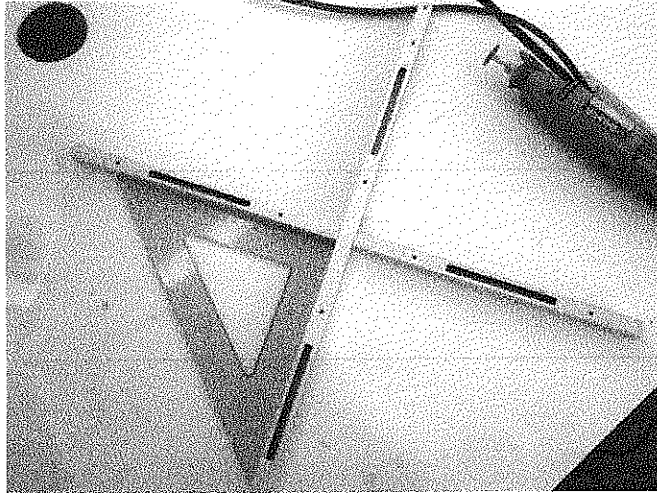


Imagen 68. Perfiles de Aluminio Estructura Quadcopter.

Fuente: Autor.

Para realizar la unión de los perfiles fue necesario construir un empalme a manera de canal para que la superficie de los estos fuera uniforme.

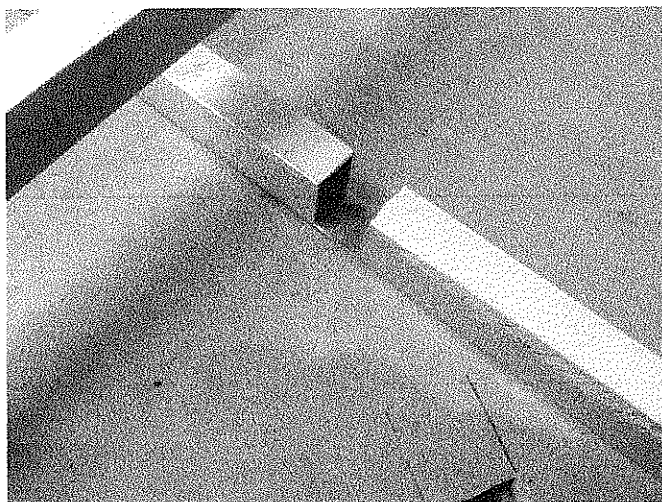


Imagen 69. Empalme de Perfiles de Aluminio.

Fuente: Autor.

El ensamble de los motores se realizó de la manera que indica el fabricante, acoplando con tornillo milimétrico las bases para quedar sujeto y anclado a la base.

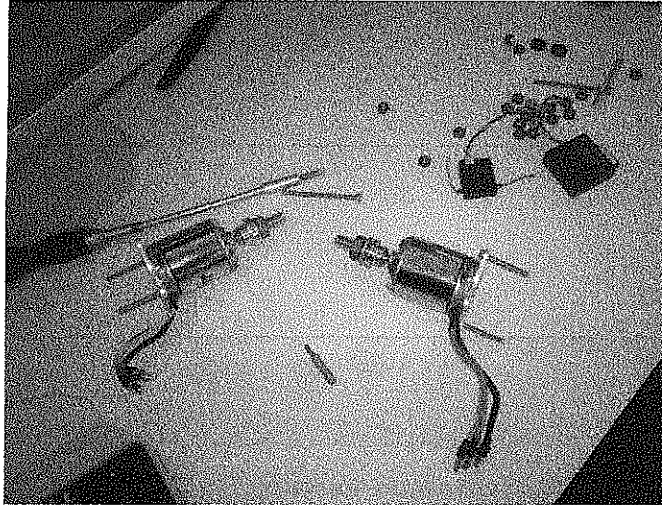


Imagen 70. Armado de los Motores.

Fuente: Autor.

En el centro de la estructura se colocó una lámina de $15\text{ cm} \times 15\text{ cm}$. De acrílico de 3 mm para sujetar los dos perfiles de aluminio garantizando con los tornillos de sujeción la perfecta perpendicularidad entre estos.

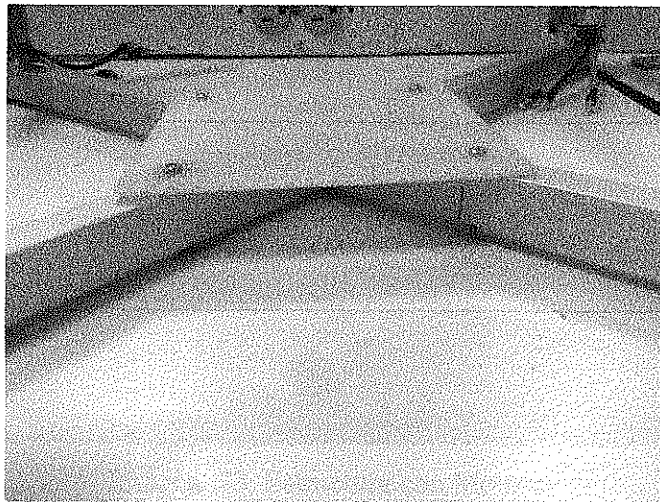


Imagen 71. Lámina de Acrílico Sujeta a la Estructura.

Fuente: Autor.

En los extremos se colocaron los cuatro (4) motores que se sujetan a los perfiles por medio de tornillos para que tuvieran consistencia, debajo de los motores se encuentran los soportes que le

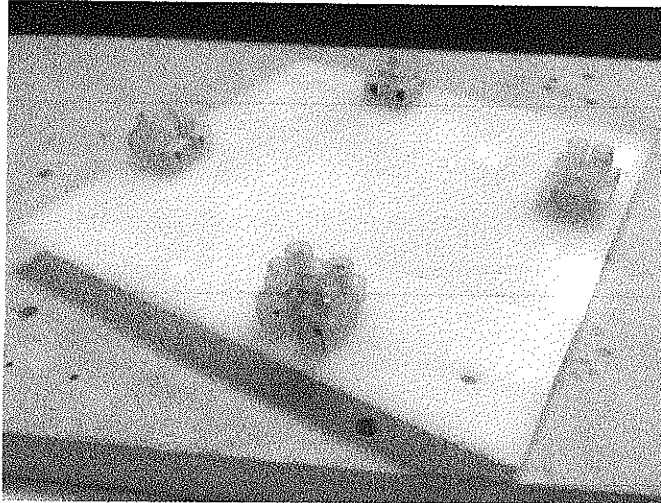


Imagen 76. Conectores de Regleta 40A.

Fuente: Autor.

Luego de tener las conexiones de potencia se ensambló la electrónica usando como base de los sensores un protoboard tamaño pequeño y las conexiones de los mismos se realizaron con cables de protoboard tipo macho-macho; en la Imagen 77, se observa un poco desordenado pero era necesario realizar pruebas antes de situar los componentes permanentemente.

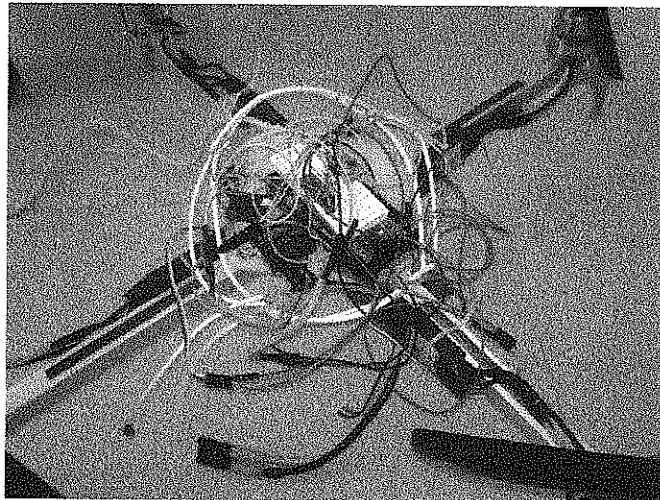


Imagen 77, Ensamble de Circuitería.

Fuente: Autor.

La primera prueba real del prototipo completamente armando se realizó sobre una estructura de balsa sujetando un eje del prototipo y dejando libre el otro, en el algoritmo de esta prueba solo se habilitó un eje tanto de sensado como de actuadores.

En esta prueba se empezó a identificar las características fundamentales del prototipo como su peso que fue inicialmente de *1100g*, se observó que la estructura tenía una alta vibración generada por los motores, adicionalmente se presentó que las láminas de acrílico se rompieron en sus esquinas debido a la vibración por lo tanto fue necesario cambiar la manera de articular la estructura agregando rigidez para disminuir la vibración, también se decidió disminuir el peso de la estructura eliminando componentes estéticamente necesarios pero funcionalmente innecesarios, se eliminó una parte de los parales de *12 cm* diseñados para que la estructura se apoyara sobre ellos disminuyendo el contacto con el suelo, fuera de lo anterior se abrieron orificios en los perfiles en el área cercana a los motores reduciendo aún más el peso de la estructura.

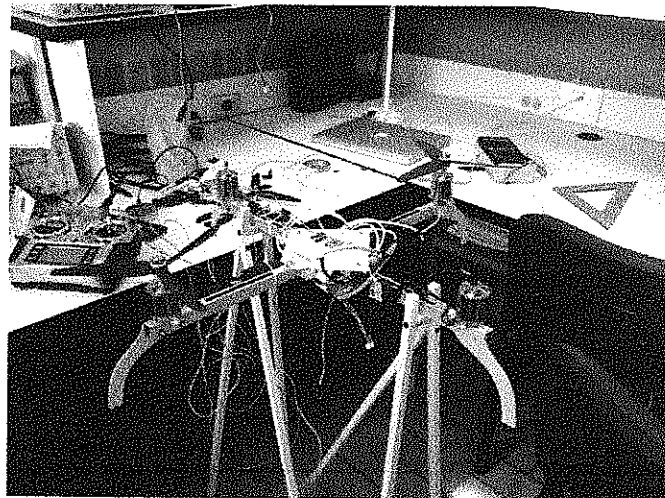


Imagen 78. Primera prueba Eje X.

Fuente: Autor.

Para solucionar el problema de la rigidez del prototipo se investigaron varios materiales que cumplieran con el requerimiento que no fueran muy pesados pero sí bastante rígidos, como resultado se consiguieron platinas de aluminio de *4 mm* de grosor que ofrecían la rigidez necesaria y el peso que se incrementaba era mínimo.

Las platinas se pusieron reemplazando las laminas de acrílico tal y como se muestra en la Imagen 70.

duplicando así el espacio disponible para el Arduino, batería, receptor, sistema de potencia y electrónica del prototipo.

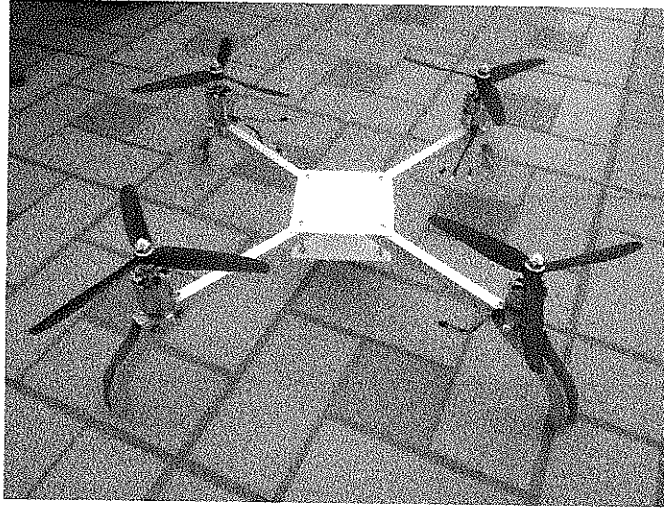


Imagen 74. Estructura con Doble Lámina de Acrílico.

Fuente: Autor.

Terminada la estructura se comenzó a situar los componentes del prototipo como el Arduino, receptor, motores, variadores (ESC) todo esto adherido temporalmente con cinta doble cara 3M para realizar pruebas de funcionamiento.

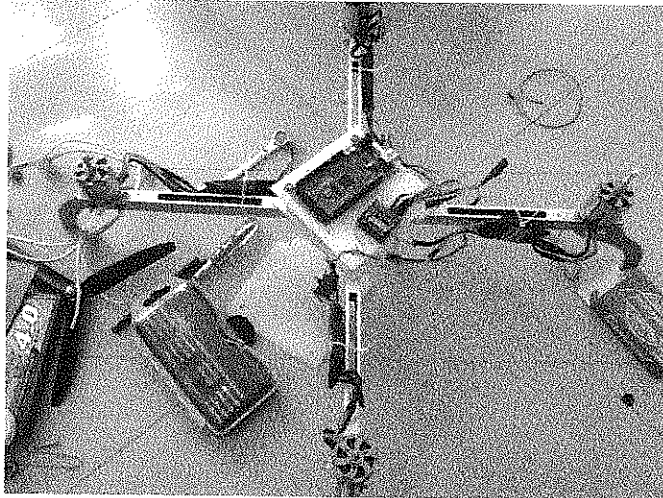


Imagen 75. Ubicación de Dispositivos en el prototipo.

Fuente: Autor.

Para el manejo y distribución de las altas corrientes se colocaron conectores de regleta de 40A en la lámina de acrílico inferior, de esta manera era posible situar los Variadores (ESC) en paralelo y realizar la distribución de corriente de manera simétrica en el prototipo.

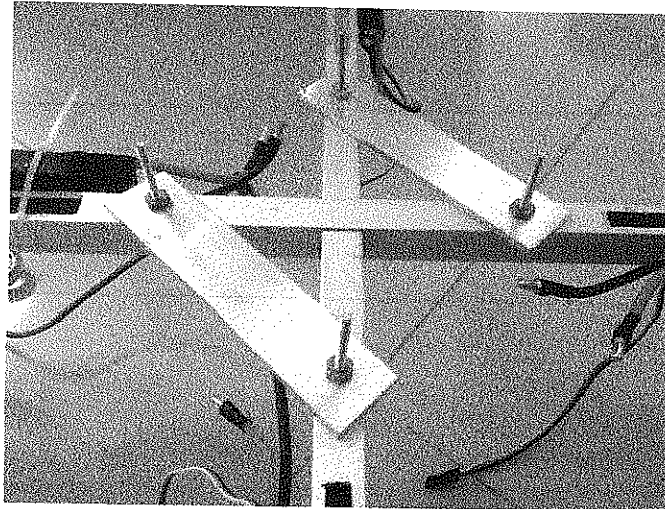


Imagen 79. Ensamble de las Platinas de Aluminio.

Fuente: Autor.

Los tornillos largos que sobresalen sobre las platinas se pusieron con el fin de hacer un aislamiento de los sensores es decir reducir al máximo el contacto directo entre los sensores y la estructura del prototipo tal y como se muestra en la Imagen 80.

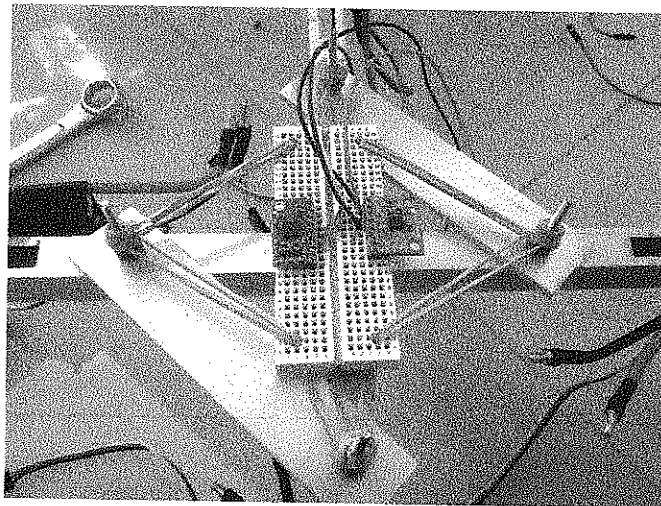


Imagen 80. Ubicación Sensores con Amortiguación.

Fuente: Autor.

Con la disposición física de sensores de la manera de la Imagen 80. Se supuso la eliminación de la vibración.



Imagen 81. Ensamble de Dispositivos en el Prototipo.

Fuente: Autor.

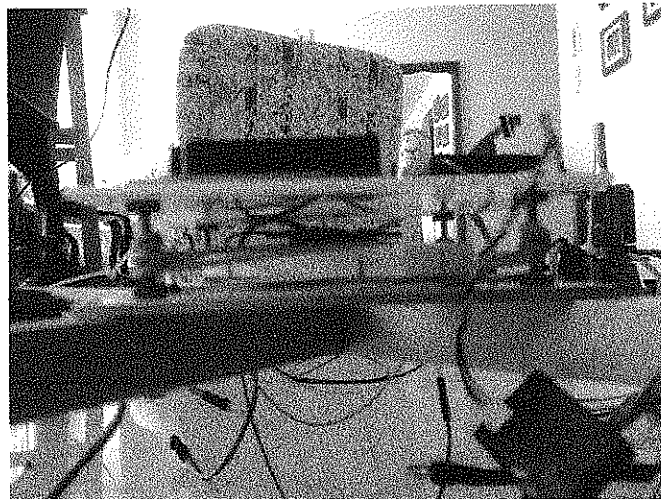


Imagen 82. Detalle de la Ubicación de los Sensores.

Fuente: Autor.

El las Imágenes 81 y 82. Se observan detalles del ensamble del prototipo indicando la posición del sensor aislado por medio de las caucheras de la estructura del prototipo.

Al ajustar los sensores se esperaba que resolvería el inconveniente de la vibración excesiva pero luego de hacer mediciones mediante el software StampPlot V4, se observó que la vibración no había disminuido.

StampPlot® es un programa de interfaz serial para adquisición de datos, es versátil y tiene una gran cantidad de herramientas las cuales no son objeto de estudio.

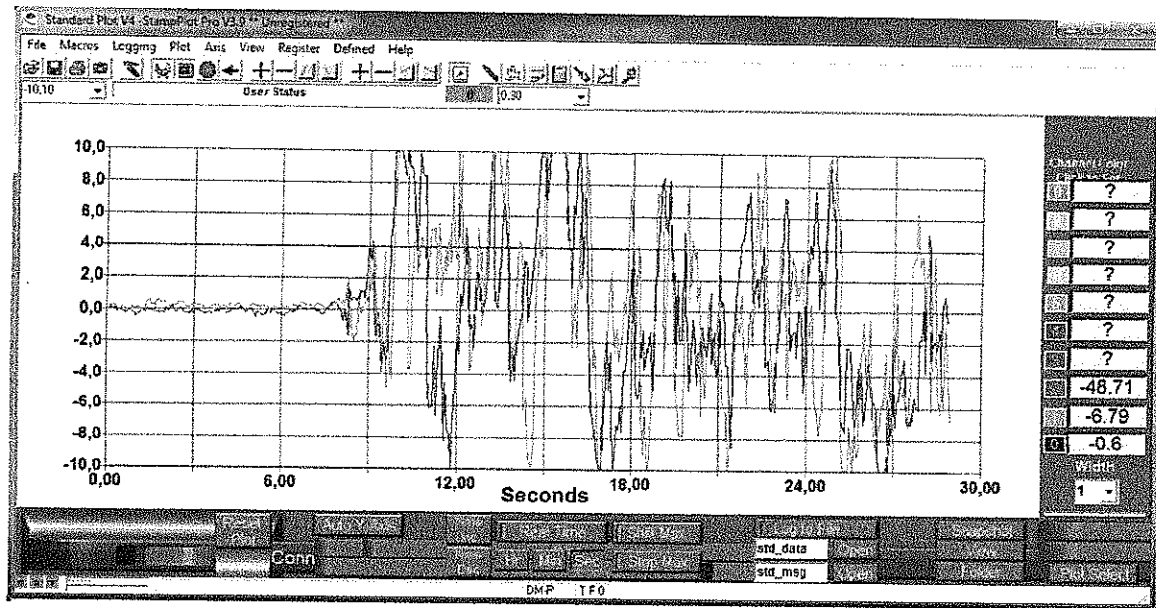


Imagen 83. Lectura de Datos Posición 1.

Fuente: Autor.

En la Imagen 83, se muestra la gráfica de la posición entregada por los sensores a través del tiempo, se puede observar que las variaciones están en un rango totalmente diferente al esperado puesto que esta prueba se realizó con el prototipo puesto en el suelo, por esta razón los datos entregados por los sensores se consideran una medida errónea ya que se esperaba que los rangos de cambio fueran mínimos.

Esta eventualidad obligó a realizar más pruebas para identificar el origen de los cambios bruscos de posición.

La prueba siguiente se hizo para verificar el origen de la vibración debido a que era posible que los motores por medio de sus piezas magnéticas afectaran con ruido eléctrico las mediciones de los sensores.

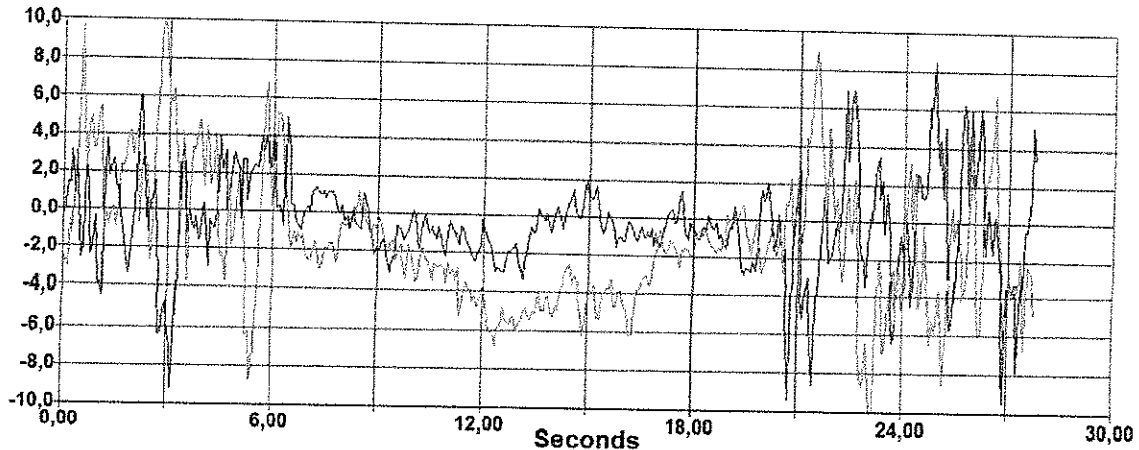


Imagen 84. Prueba Aislado los Sensores.

Fuente: Autor.

El objetivo de esta prueba era determinar si el sensor estaba afectado por la vibración mecánica del prototipo o por el ruido eléctrico, la prueba se inicio con el sensor sujeto al prototipo, luego de un tiempo de 6 segundos, el sensor se retiró del prototipo con la mano durante *14 segundos* y luego se volvió a poner en el prototipo, los resultados se muestran en la Imagen 84.

Analizando la señal recibida (Imagen 84) se observa que en el intervalo de *6 a 20 segundos* durante el cual el sensor estuvo apartado del prototipo la vibración disminuyo considerablemente afectando su amplitud y frecuencia, lo cual indica que la vibración registrada en ese tiempo es solamente física en este caso el movimiento de la mano.

Luego, teniendo la certeza que el origen de la vibración era solamente físico se pensó en que el prototipo de pruebas requería de una mejor presentación estética con cables y elementos sin sujeción alguna, entonces se decidió realizar el PCB (Printed Circuit Board) de la etapa de potencia y alimentación de circuitos además montar en un PCB los sensores para que no hayan piezas móviles que influyan en la vibración.

Para asignar un orden y ubicar los componentes permanentemente se diseñaron los PCB de potencia y de los sensores.

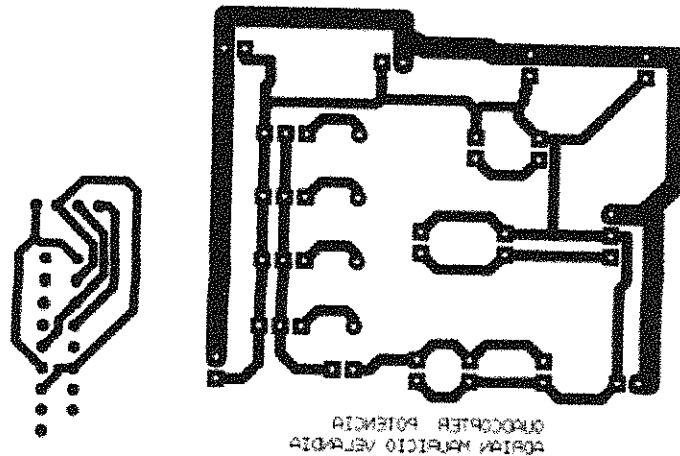


Imagen 85. PCB Sensores y Potencia.

Fuente: Autor.

En la Imagen 85, se muestra los diseños de los PCB de potencia y sensores los cuales están diseñados para cumplir con la necesidad de alimentación de los circuitos.

En el PCB de potencia se dispuso un regulador Lm7809 de $9V$ con el fin de utilizar la misma alimentación de la batería LiPo para el Arduino, sensores y receptor y como agregado se pusieron conectores en el PCB los cuales proveen de tensiones de $12V$ directamente de la batería, $9V$ regulados, $5V$ y $3.3V$ regulados provenientes del Arduino.

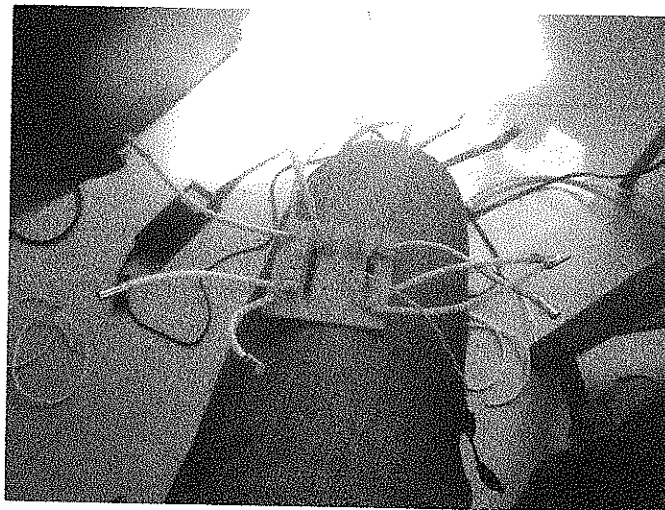


Imagen 86. PCB Implementado.

Fuente: Autor.

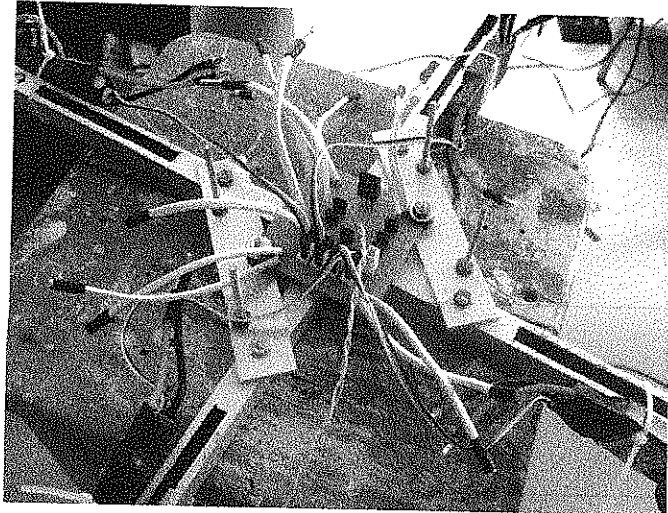


Imagen 87. PCB de Potencia Puesto en el Prototipo.

Fuente: Autor.

En la Imagen 87, se observa el PCB ya puesto en el prototipo listo para conectar y poner en marcha.

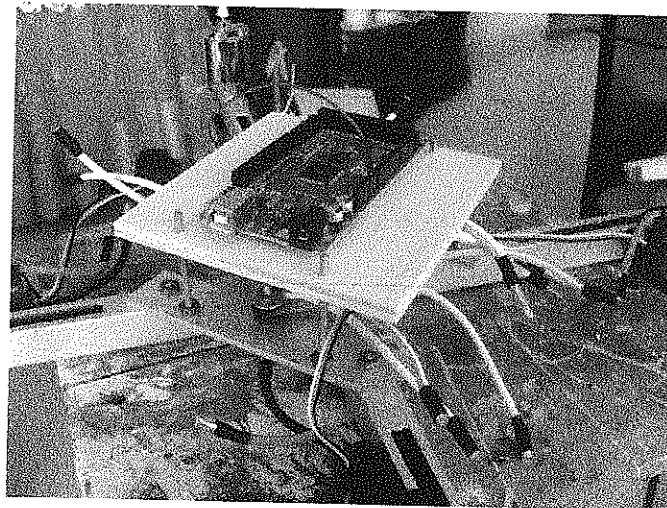


Imagen 88. Ubicación del Microprocesador en el Prototipo.

Fuente: Autor.

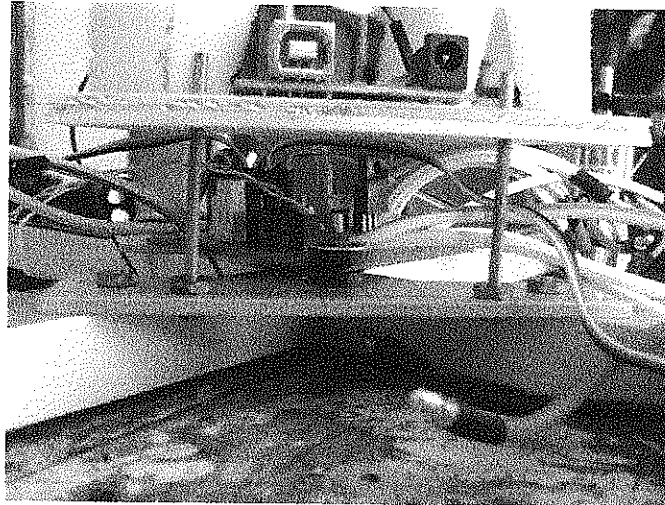


Imagen 89. Detalle de la Ubicación del PCB en el Prototipo.

Fuente: Autor.

Luego de ubicar el procesador se puso el sensor sobre este y se realizó una prueba de funcionamiento llevando a máxima potencia los motores con el radio control, funcionando perfectamente, la respuesta en cuanto a la vibración seguía permaneciendo igual.

Otra solución pensada fue la de realizar una caja de balsa aislando el sensor de las afectación del viento revolucionado por el giro de las hélices de los motores.

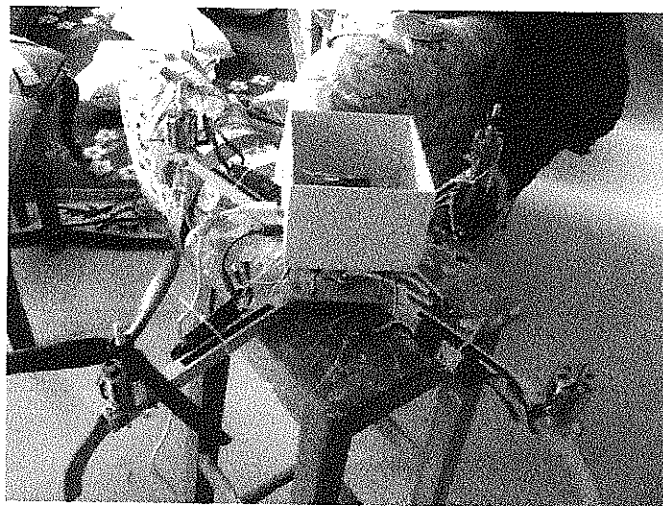


Imagen 90. Ensamble de Caja de Balsa Prototipo.

Fuente: Autor.

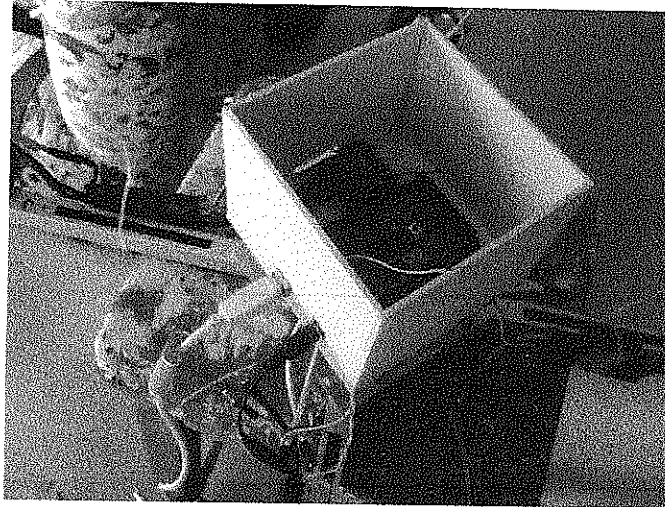


Imagen 91. Detalle Caja para Sensor y Procesador.

Fuente: Autor.

La siguiente prueba realizada dio como resultado que la vibración seguía estando presente en el sensor, por lo tanto se exploró y se buscó materiales para aislar la vibración mecánica, que no transmitieran la vibración como las caucheras inicialmente instaladas.

Los materiales encontrados fueron: espuma de diferentes densidades, algodón, amortiguadores de los radios de carro y guata.

Se inicio probando cada material, primero se probó la espuma (Imagen 92) situándola dentro de la caja diseñada para el sensor.

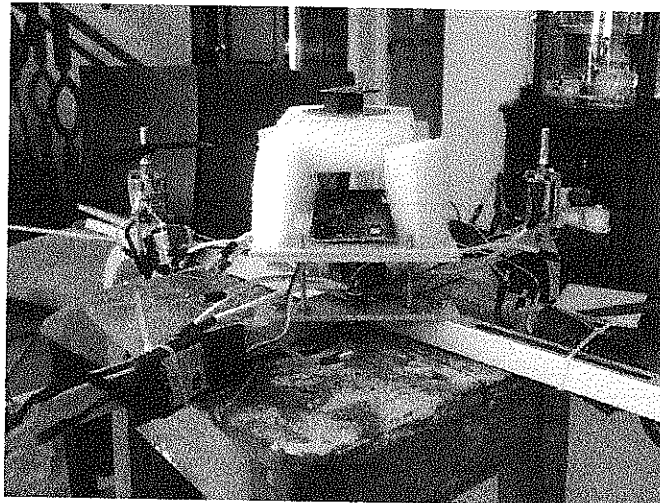


Imagen 92. Diseño y Ubicación del Sensor con Espuma.

Fuente: Autor

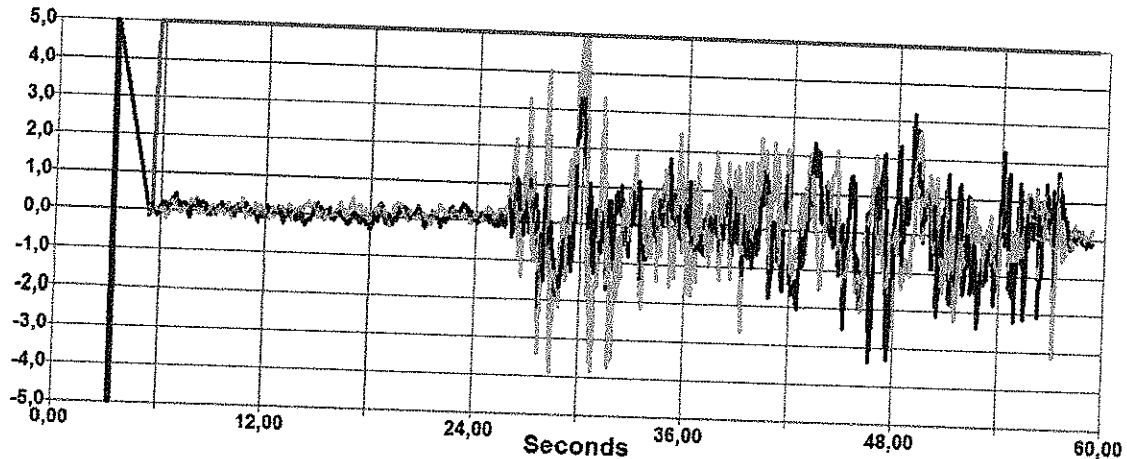


Imagen 93. Prueba con Espuma.

Fuente: Autor.

La prueba con espuma como amortiguador dió como resultado la reducción de la vibración; reduciendo la amplitud máxima de la señal de ruido, pero todavía no era suficiente ya que se espera tener una vibración mínima, con una amplitud máxima de *1,4 grados* debido a que las variables del proceso de control son enteros y los cambios de *-1 a 1* son mínimos dentro del controlador.

Luego se probó con algodón realizando dos pruebas en las cuales se obtuvo los resultados mostrados en las imágenes 94 y 95.

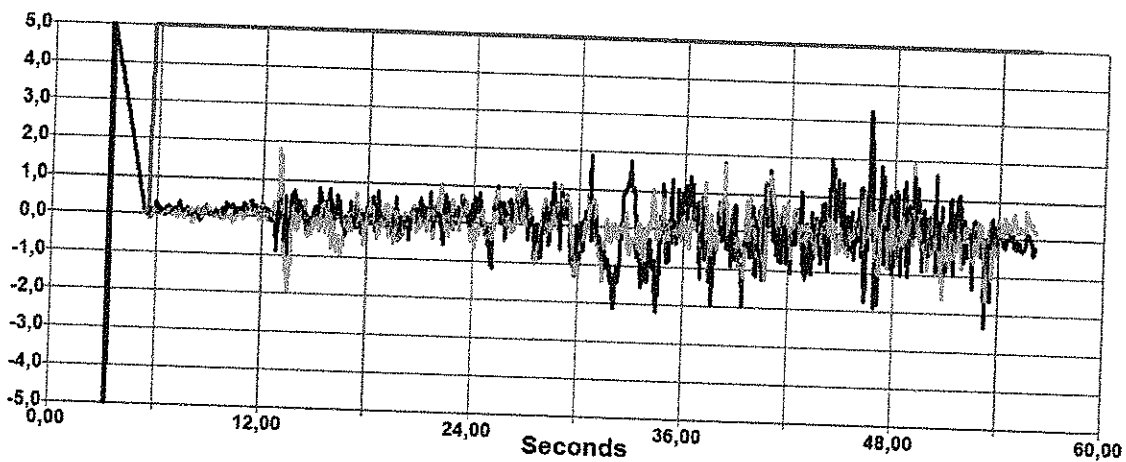


Imagen 94. Primera Prueba Algodón.

Fuente: Autor.

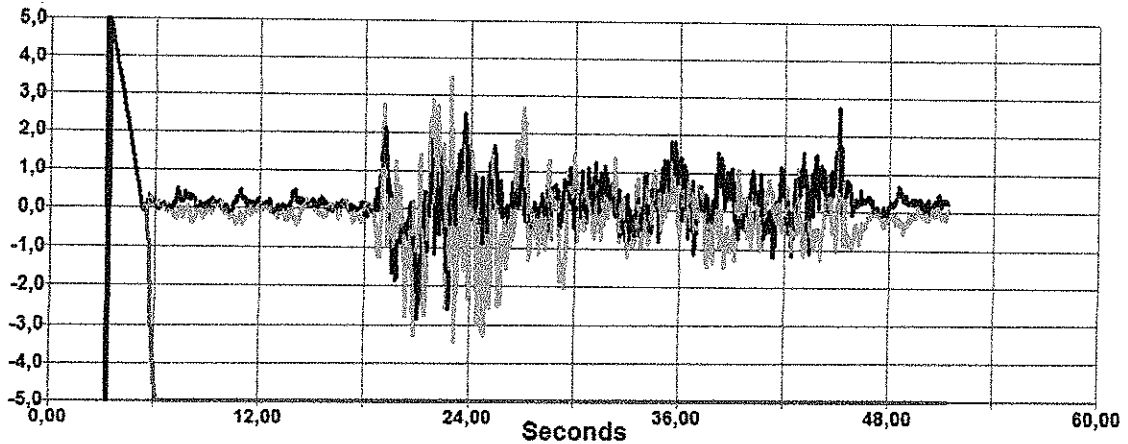


Imagen 95, Segunda Prueba Algodón.

Fuente: Autor.

Las pruebas con el algodón como material amortiguador, redujo un poco la vibración, pero las amplitudes máximas de la vibración son de *3.6 grados*, lo cual no es bueno para el controlador por que va a manejar datos erróneos interpretando esta vibración como la posición real del prototipo ejerciendo un control incorrecto.

Posteriormente se probó con guata siendo un material más denso que el algodón pero con más espacio entre sus fibras ofreciendo la posibilidad de amortiguar la vibración.

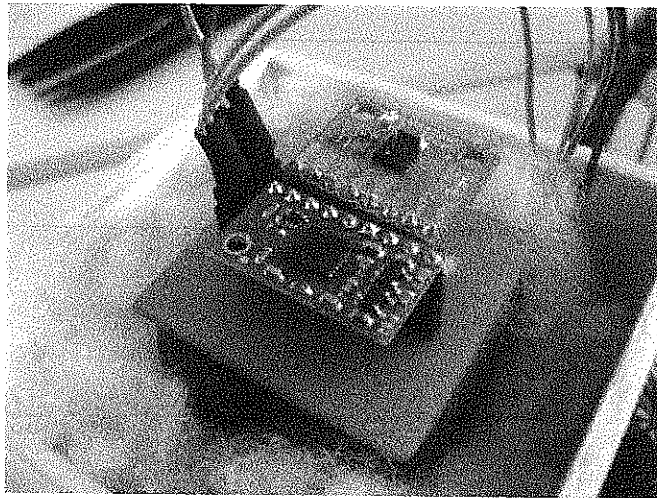


Imagen 96. Sensor Amortiguado por Guata.

Fuente: Autor.

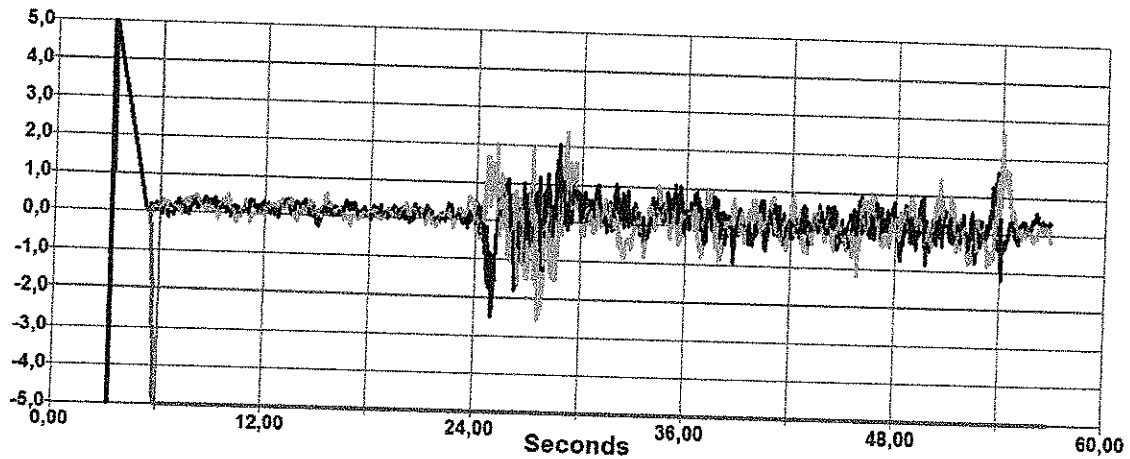


Imagen 97. Prueba con Guata.

Fuente: Autor.

En la Imagen 97, se muestra los resultados obtenidos con guata como material de amortiguación, se puede observar que en el intervalo de tiempo entre 24 y 30 segundos que es intervalo donde los motores giran a bajas revoluciones aumenta su vibración y posterior a esto la vibración se estabiliza cerca de los rangos deseados.

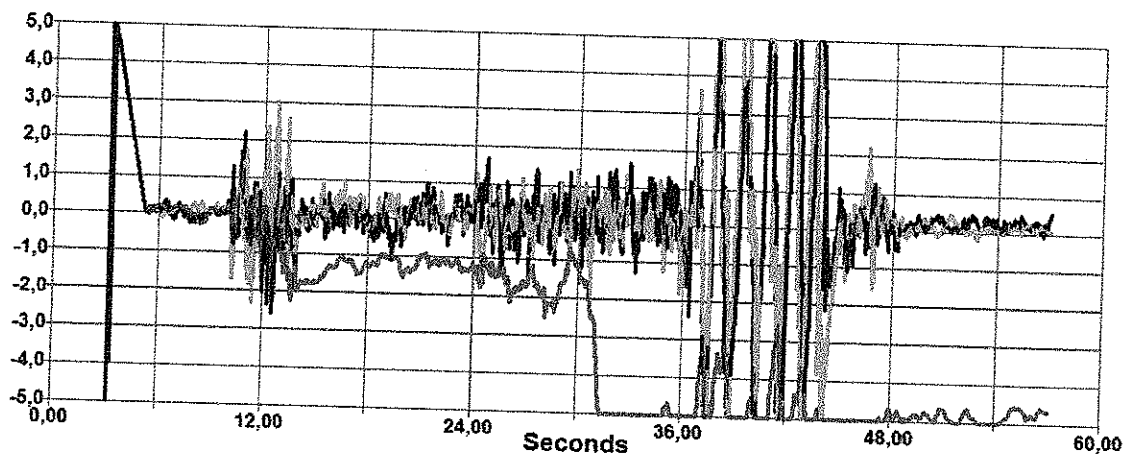


Imagen 98. Prueba con Guata, Elevación.

Fuente: Autor.

En la Imagen 98, se observa la segunda prueba con guata la cual tiene el resultado esperado en el tiempo en el cual el prototipo se encuentra en el suelo, luego de los 36 segundos se le dió más potencia a los motores y el prototipo se elevó, el sensor comenzó a vibrar demasiado obligando a detener la prueba los 48 segundos.

Debido a que se creía posible que la estructura construida estuviera generando esta vibración se decidió adquirir la estructura de fibra de carbono nombrada anteriormente.

El prototipo se ensambla en la nueva estructura (ver Imágenes 99, 100 y 101)

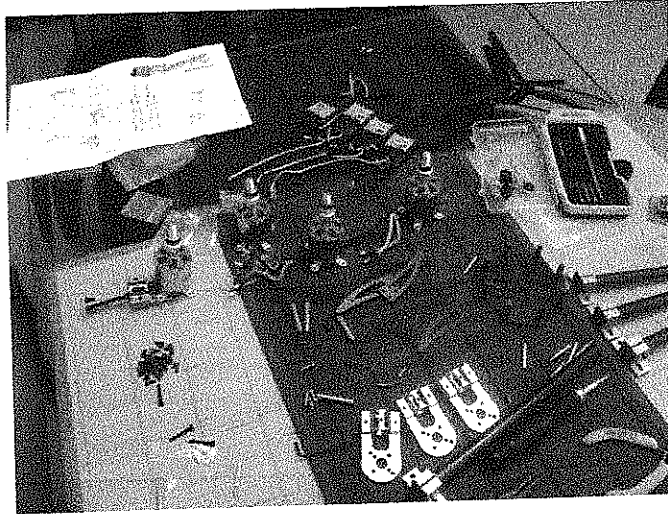


Imagen 99. Despiece de la Estructura de Fibra de Carbono.

Fuente: Autor.

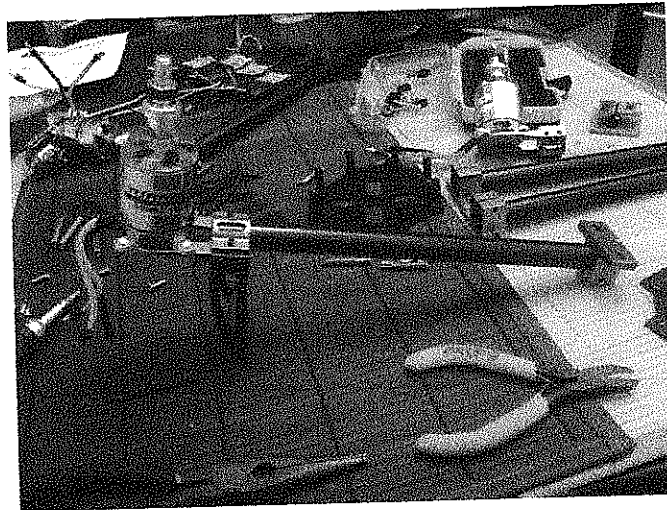


Imagen 100. Detalle Ensamble del Motor en un Eje de la Estructura de Fibra de Carbono.

Fuente: Autor.

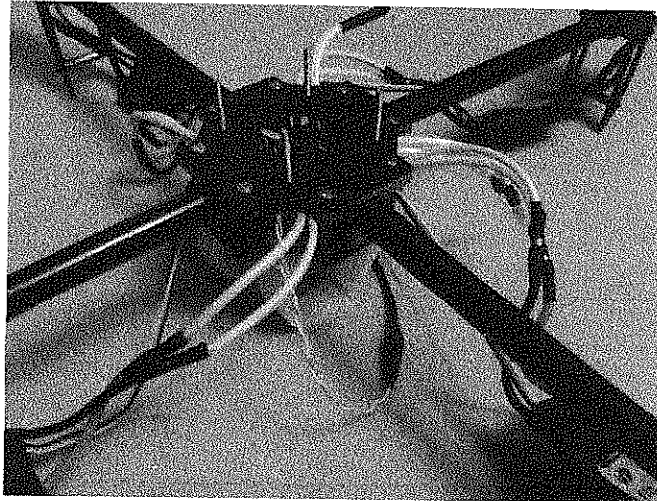


Imagen 101. Detalle Estructura Armada con PCB de Potencia.

Fuente: Autor.

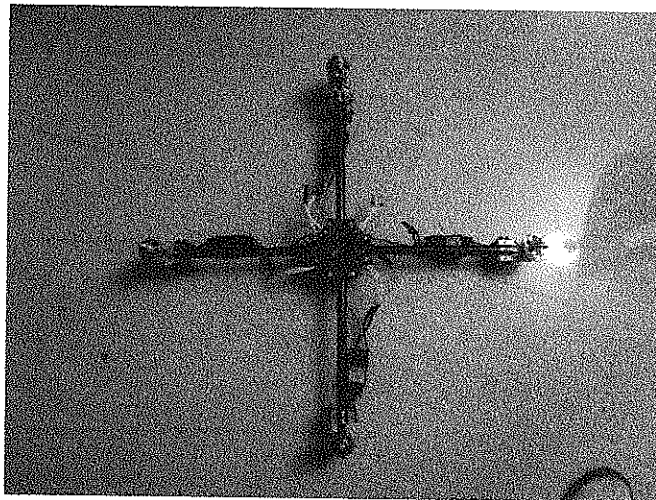


Imagen 102. Prototipo Estructura Fibra de Carbono Perfil Superior.

Fuente: Autor.

Gracias a que la estructura inicial de aluminio contenía los dispositivos electrónicos y demás, debido a que el proceso fue: armar estructura de Fibra de Carbono, desmontar dispositivos de la estructura de aluminio y luego montarlos en la estructura de Fibra de Carbono, se logró realizar la comparación física de las dos estructuras (ver Imagen 103).

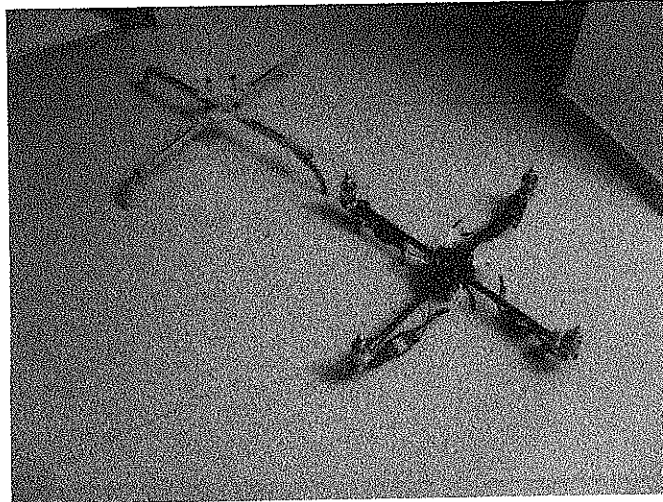


Imagen 103. Comparación Estructuras Quadcopter.

Fuente: Autor.

Al montar la electrónica en la estructura de Fibra de Carbono se observó el prototipo más ordenado, como se muestra en la Imagen 104, el Arduino se situó en el nivel intermedio entre la caja de el sensor y la batería junto al receptor.



Imagen 104. Ubicación Dispositivos Estructura Fibra de Carbono.

Fuente: Autor.

En la Imagen 105, se observa el detalle del ensamble de los motores junto con las hélices.

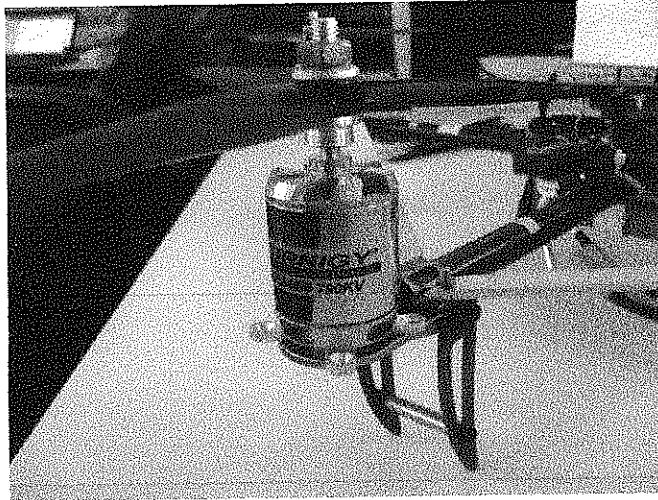


Imagen 105. Detalle Motor Estructura Fibra de Carbono.

Fuente: Autor.

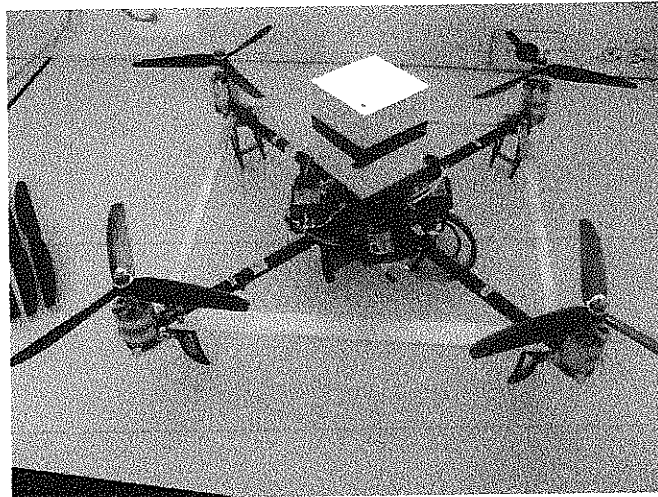


Imagen 106. Perfil Superior Estructura de Fibra de Carbono Lista.

Fuente: Autor.

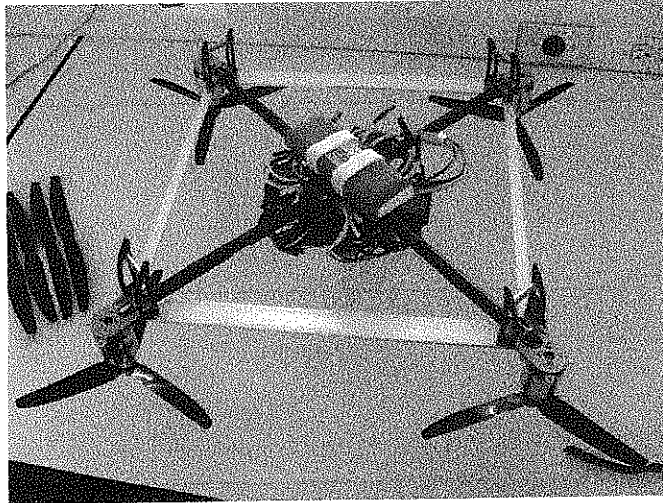


Imagen 107. Perfil inferior Estructura de Fibra de Carbono Lista.

Fuente: Autor.

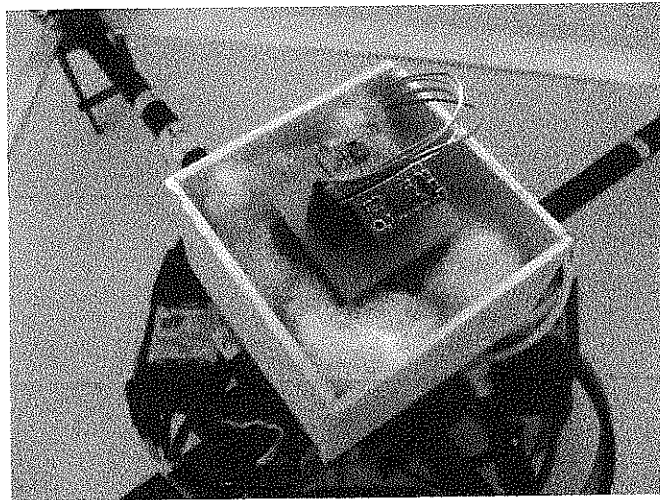


Imagen 108. Ubicación de los Sensores en la Estructura de Fibra de Carbono.

Fuente: Autor.

Luego de la instalación completa de toda la electrónica y los dispositivos del prototipo se realizaron pruebas del sistema integrado. (ver Imagen 109).

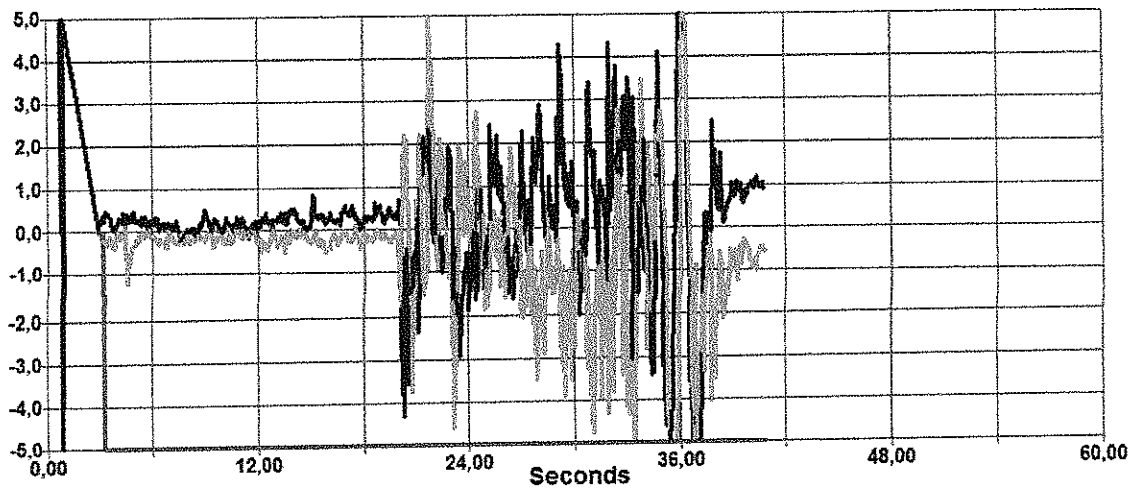


Imagen 109. Prueba Estructura de fibra de Carbono.

Fuente: Autor.

En la imagen 109 se observa que la vibración aumentó con el cambio de la estructura. Como nueva solución se pensó en modificar los registros y parámetros de configuración del sensor giróscopo.

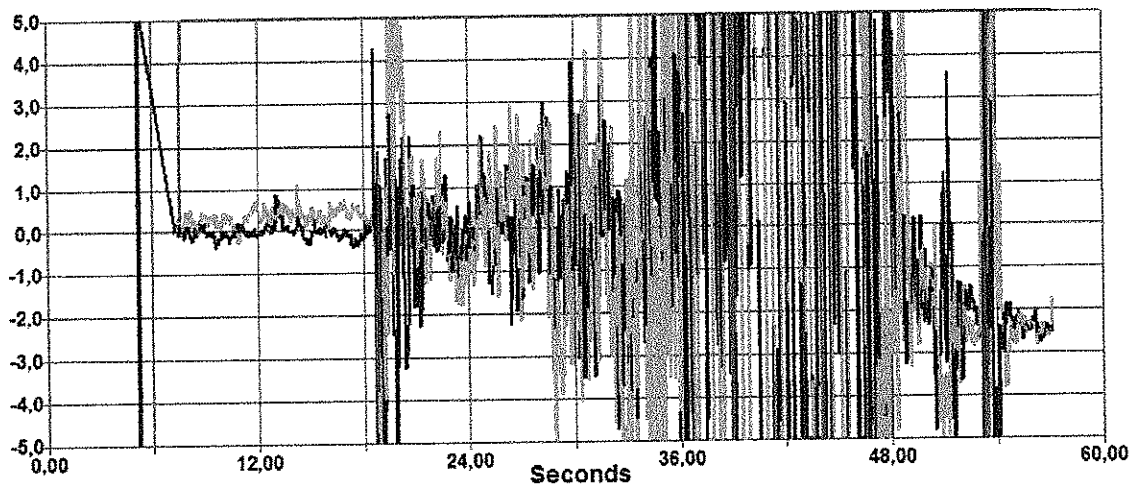


Imagen 110. Prueba Configurando Sensibilidad a 250dps.

Fuente: Autor.

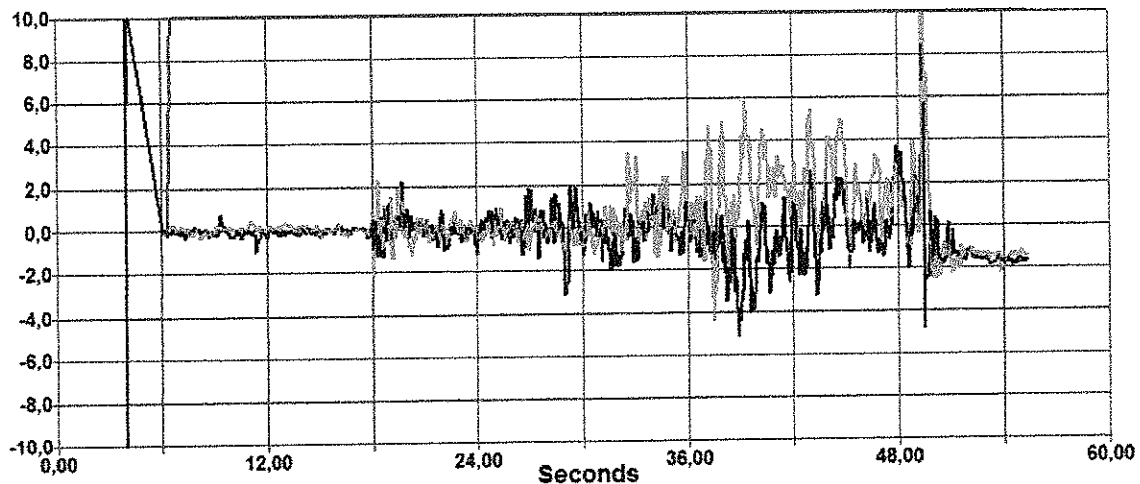


Imagen 111. Prueba Configurando Sensibilidad a 2000dps.

Fuente: Autor.

Como resultados de las pruebas de las Imágenes 110 y 111, se observó que reducir la sensibilidad solo produjo que la vibración aumentara desmedidamente debido a que al tener menos sensibilidad el sensor registró datos erróneos de los cuales la moda era un dato totalmente lejano de la posición real del prototipo.

Continuando con la investigación se analizó la posibilidad de que las hélices se encontraran desbalanceadas, cuando una hélice se encuentra desbalanceada significa que el peso de sus aspas no es el mismo, esto conlleva a que la fuerza rotatoria ejerza más fuerza en una dirección por lo tanto cuando el motor gira genera una vibración cada vez mayor proporcional al nivel de desbalance.

Dependiendo de la calidad de la hélice, requerirá ser balanceada para que el motor pueda rendir a su máxima eficiencia. El desbalance de una hélice trae consigo factores negativos que se aplican directamente al modelo y al motor. Los factores negativos son causados por la vibración. Desde el punto de vista de un motor la vibración es su enemigo mortal por que acorta rápidamente su vida, especialmente sobre los bujes y rodamientos; también la vibración hace que el motor no proporcione las RPM que él puede dar y dependiendo de la cantidad de vibración, se ha podido observar en varias ocasiones que el motor presenta fallas de funcionamiento. Desde el punto de vista del modelo, la vibración es enemiga de las partes que no han sido ajustadas, también es

enemiga de las partes electrónicas; sin embargo el mayor daño será percibido por el motor del modelo⁴⁰.

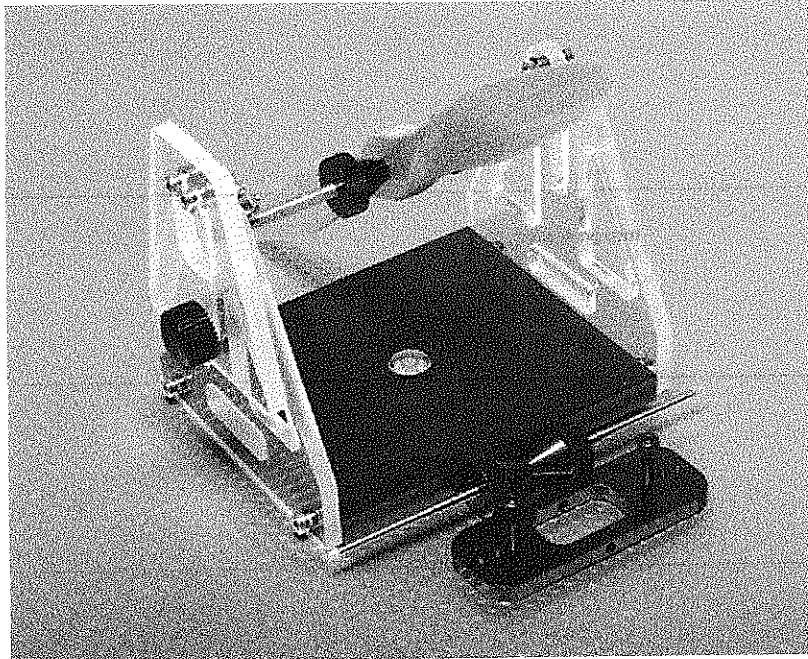


Imagen 112. Balanceador Magnético Adquirido para el Proyecto.

Fuente: Autor.

El balanceo de las hélices se realizó agregando capas de cinta en las aspas menos pesadas para agregar peso a la pala con el fin de nivelarla.

Luego del proceso de balanceo se realizó la lectura de datos para comprobar los resultados del mismo.

En la imagen 113, se muestra la respuesta del sensor ante las modificaciones realizadas, se tomó como prueba exitosa.

Luego de realizar un promedio de los últimos 10 datos se realizó el envío de este dato para así procesarlo en el algoritmo de control, el promedio actúa a manera de un filtro digital implementado en el microcontrolador.

⁴⁰ http://www.todohobbies.com.uy/sitios/taller/balance_de_la_helice.htm

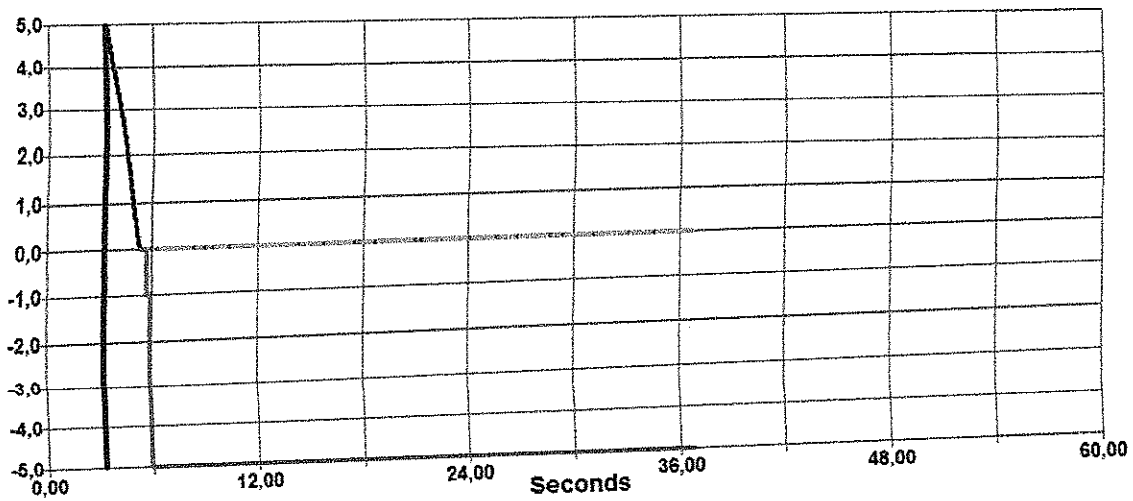


Imagen 113. Prueba luego del Balanceo y Promedio con 10 datos.

Fuente: Autor.

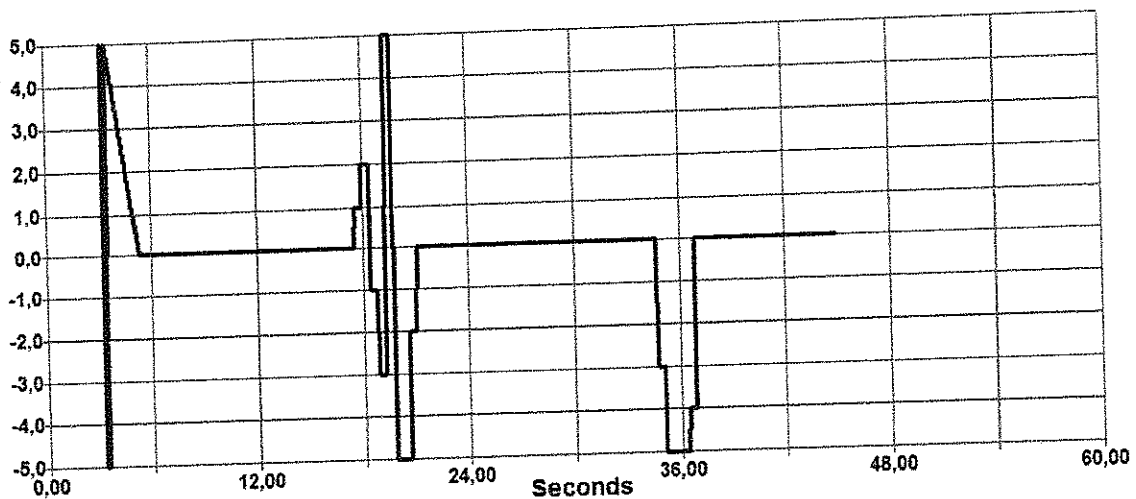


Imagen 114. Prueba luego del Balanceo y Promedio con 5 Datos eje X.

Fuente: Autor.

En la Imagen 114 se muestra el resultado del procesamiento final de los datos, se observa que en el tiempo entre 6 y 18 segundos el prototipo se encuentra estable luego se inclina de tal manera que el sensor registra esta inclinación para volver a equilibrarse alrededor de los 22 segundos, a los 35 segundos se aplica otra perturbación a la cual el controlador responde efectivamente guiando al prototipo a la posición inicial.

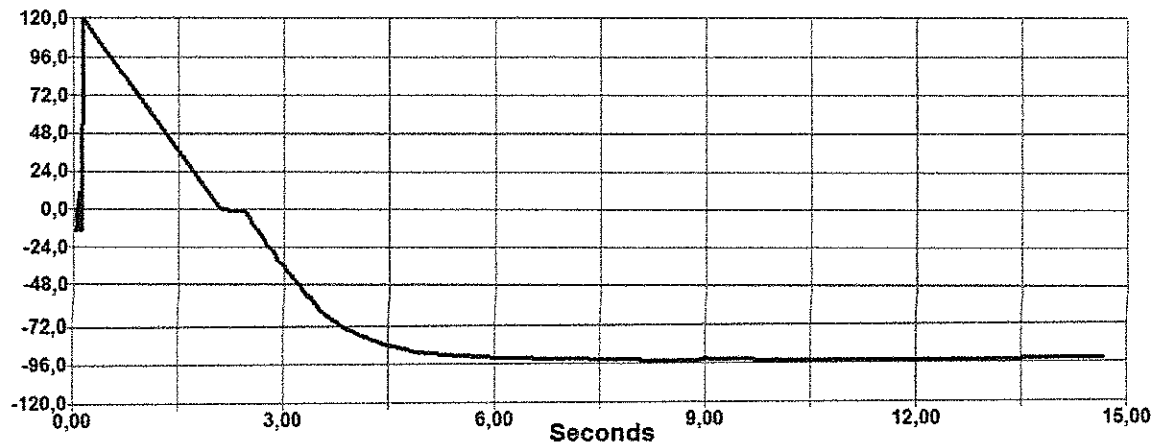


Imagen 115. Prueba luego del Balanceo y Promedio con 5 datos eje Z.

Fuente: Autor.

En la Imagen 115, se muestra la curva de calibración de orientación realizada por el algoritmo de control, guiado por el magnetómetro el algoritmo toma un tiempo para establecerse referenciado por el norte magnético de la tierra, para finalmente permanecer constante indicando que el prototipo se encuentra sin presencia de rotación.

VIII. COSTOS TOTALES DEL PROYECTO

Cantidad	Componente	Precio Unitario		Total
		Dólares	Pesos	
4	Turnigy D3548/6 790KV Brushless Outrunner M	80	144000	576000
1	USBasp AVR Programming Device for ATMEL	15	27000	27000
4	Hobbyking SS Series 40-50A ESC	24	43200	172800
1	Turnigy Accucel-6 50W 6A Balancer/Charger w/	60	108000	108000
1	2 Pin 6 x Pico Plug Charging Set Eflight comp	18	32400	32400
4	GWS HD9050 3 Blade Prop Counter Rotate (2pc)	15	27000	108000
4	GWS HD9050 3 Blade Prop (2pc pack)	15	27000	108000
1	TURNIGY BESSC Programming Card	10	18000	18000
4	HXT900 9g / 1.6kg / .12sec Micro Servo	8	14400	57600
1	900MHZ 1500mW Tx/Rx & 1/3-inch CCD Cam.	120	216000	216000
2	4mm Gold Connectors 10 pairs (20pc)	6	10800	21600
2	HXT 3.5mm Gold Connector w/ Protector (10pcs)	6	10800	21600
1	HobbyKing Multi-Rotor Control Board V2.1 (At	15	27000	27000
2	PolyMax 3.5mm Gold Connectors 10 PAIRS	10	18000	36000
6	10X6 Propellers (Standard and Counter Rot.	12	21600	129600
1	TURNIGY TRUST 55A SBEC Brushless Speed	22	39600	39600
1	Lithium Polymer Charge Pack 25x33cm JUMBO	28	50400	50400
1	ZIPPY Flightmax 2200mAh 3S1P 40C	18	32400	32400
2	Turnigy 4000mAh 3S 20C Lipo Pack	25	45000	90000
1	Balanceador Hélices	72,2	130000	130000
1	Arduino Mega 2560	77,8	140000	140000
2	Conectores Arduino Tipo Macho-Macho	6,7	12000	24000
1	Cinta 3M	12,2	22000	22000
1	Amortiguadores	38,9	70000	70000
1	Estructura Fibra de Carbono	166,7	300000	300000
1	Elementos Construcción Prototipo	83,3	150000	150000
1	Sensores	133,3	240000	240000

1	Elementos Amortiguación	22,2	40000	40000
1	Balso	16,7	30000	30000
1	Fuente PC	11,1	20000	20000
1	Costo Envíos Hong Kong	400	720000	720000
1	Radio Control	500	900000	900000
	Otros Gastos (Transportes, Comunicaciones)	277,8	500000	500000
			Total Gastos	5158000

Tabla 12. Costos Totales del Proyecto.

Fuente: Autor.

IX. CONCLUSIONES

Durante el proceso de selección de dispositivos no solamente se debe hacer la elección basada en parámetros técnicos, sino también se debe contar con información acerca de la tendencia de uso, las experiencias de las personas que han tenido contacto con estos dispositivos y las recomendaciones del fabricante debido a que son máquinas en las cuales los dispositivos deben funcionar completamente sincronizados; es necesario contar con dispositivos que se complementen, como caso puntual se cita la selección de las hélices, el costo de las hélices seleccionadas para el prototipo fue muy cómodo y se ajustaban a los requerimientos de los motores, sin embargo, la calidad de las mismas era mínima lo cual condujo a la aparición de inconvenientes en el momento de evaluar el rendimiento del controlador y de los demás dispositivos.

En el diseño de sistemas de control se realizaron pruebas de escritorio teniendo en cuenta posibles perturbaciones externas. En el transcurso del desarrollo del proyecto se observó que cuando se implementa un prototipo puede haber perturbaciones que se salen de los parámetros de diseño, perturbaciones que simplemente son causadas por el hardware que no es posible controlar de una manera digital. En el desarrollo del proyecto uno de los principales inconvenientes fue la vibración mecánica generada por las hélices, esta dificultad se solucionó balanceando cada una de ellas, con base en la experiencia brindada. Por esta situación, se recomienda contemplar todos los parámetros de hardware y posibles fallas en él, dentro de un diseño electrónico.

El diseño y la implementación de máquinas voladoras es un campo de desarrollo e investigación muy amplio que presenta demasiadas variables ya que son máquinas que no solo tienen la influencia de su arquitectura sino se ven afectadas por entornos cambiantes donde cualquier tipo de variación puede influir negativamente en el funcionamiento. En una prueba de vuelo hecha con el prototipo ocurrió que una semilla de un árbol cayó en una de sus hélices mientras este estaba elevándose, la semilla rompió la hélice y el prototipo cayó de manera repentina cerca al operador del radio control, por esta razón se recomienda hacer uso de todas las medidas de precaución y seguridad para evitar daños irreparables en el personal cercano a los vuelos.

Se ha demostrado que con el uso de la plataforma Arduino y los microcontroladores de la familia Atmel, es posible culminar proyectos robustos y complejos a muy bajo costo, el software de Arduino es fácilmente escalable y se adapta a cualquier tipo de hardware.

Todos los objetivos planteados se cumplieron de manera satisfactoria, pero la dificultad del proyecto ha sido mayor de la prevista debido a que este desarrollo tuvo un carácter multidisciplinario, las problemáticas y soluciones plasmadas en este documento solo son una muestra de muchas que se presentaron en el transcurso del proyecto, por esto se recomienda; en el momento de acondicionar un sistema electrónico en una plataforma mecánica es necesario realizar pruebas desde el primer momento, verificar los resultados y obtener un registro del comportamiento durante todo el proceso de adaptación del sistema, de este modo la detección de fallas es mucho mas rápida y efectiva.

Los sensores tipo MEMS son una herramienta efectiva y comúnmente utilizada en muchos sectores como la industria, medicina, y gran cantidad de procesos en los cuales es necesario realizar una lectura e interpretación de señales del entorno, dentro del proyecto se trabajo con dos sensores de este tipo suficientes para entregar los resultados esperados

El controlador lógico difuso permite tratar con información que contiene un alto grado de imprecisión contrario a la lógica tradicional que trabaja con información definida y precisa, la efectividad del controlador se evalúa en el momento de la implementación, si la información presente en las entradas es desacertada, el controlador actuara de forma errónea, en este caso gracias al procesamiento realizado y el buen desempeño de los sensores, el controlador seleccionado actuó de manera efectiva.

Sí es posible soñar algo, es posible hacerlo realidad. Solo hace falta una meta tan alta como se pueda imaginar.

X. DESARROLLOS FUTUROS

- Implementación control multidireccional de vuelo.
- Alertas de eventos como bajo nivel de batería y presencia de obstáculos.
- Transmisión de video en tiempo real.
- Implementación de sensores de proximidad para evitar choques.
- Implementación de GPS para realizar recorridos programados.
- Hardware y Software para realizar vuelos autónomos identificando objetos.
- Implementación de sistema vuelta a casa.
- Vuelo para largas distancias.
- Sistema de vision artificial.

XI. RECOMENDACIONES

Este proyecto se realizó con la intención de que fuese la plataforma y fundamento teórico-práctico de proyectos similares, así mismo, la información esta disponible para toda la comunidad académica de la Universidad Santo Tomás Seccional Tunja, brindando la posibilidad de que otras personas continúen con el desarrollo propuesto ya que se cuenta con la base documental necesaria para introducirse al mundo de las maquinas a un nivel de alta Ingeniería, se sugiere continuar con la proyección de desarrollos planteada dentro del documento.

En cuanto a recomendaciones del prototipo se recomienda:

- El sistema físico debe encontrarse en perfectas condiciones, las hélices deben estar balanceadas y el peso bien distribuido, todo esto reduce considerablemente las vibraciones que se generan el prototipo afectando el sistema electrónico.
- La implementación de un filtro de Kalman para reducir la vibración por este método y ayudar con el rendimiento del controlador.
- Cumplir con los desarrollos futuros y si es así, hacer uso de varios microprocesadores o uno multiproceso, con el fin de realizar tareas simultaneas.
- Tener en cuenta el tiempo de duración de la batería de realizando el procedimiento matemático sugerido por los fabricantes.
- Disminuir al máximo el peso del prototipo.

XII. REFERENCIAS

- [1] Quadruino.com. *Motores Brushless*. Recuperado el 2 de Abril de 2012, de <http://www.quadruino.com/guia-2/materiales-necesarios-1/motores-brushless>.
- [2] *Control de un motor Brushless DC con frenado regenerativo*. Recuperado el 3 de Abril de 2012, de http://www.infopl.net/files/documentacion/motion_control/infopl_net_tesis89.pdf
- [3] *Motores Brushless*. Recuperado el 5 de Abril de 2012, de <http://www.brushlessmotor.com.ar/>.
- [4] E-Aeromodelismo. *Construcción de un motor Brushless en madera*. Recuperado el 5 de Abril de 2012, de http://www.e-aeromodelismo.com.ar/Notas/brushless/brushless_1.htm.
- [5] E-Radio Control Circuitos Electrónicos de Radio Control. *Motores Brushless*. Recuperado el 6 de Abril de 2012, de http://www.e-radiocontrol.com.ar/?Motores_Brushless.
- [6] Avión microservos. *¿Sabias qué es el paso de la hélice?* Recuperado el 8 de Abril de 2012, de <http://avion.microservos.com/sabias/sabias-que-es-el-paso-de-la-helice.html>.
- [7] *AJUSTE DE UN MOTOR GLOW DE 2 TIEMPOS PARA AEROMODELISMO*. Recuperado el 8 de Abril de 2012, de <http://www.aeromasos.galeon.com/castella/ajustes.htm>.
- [8] El Laboratorio Loco. *Los motores Glow*. Recuperado el 8 de Abril de 2012, de <http://ellaboratorioloco.wordpress.com/2006/03/30/los-motores-glow-i/>.
- [9] *El servo motor*. Recuperado el 12 de Abril de 2012, de <http://www.todorobot.com.ar/documentos/servomotor.pdf>.

- [10] NitroyLitio. *Baterías Lipo*. Recuperado el 12 de Abril de 2012, de <http://www.nitroylitio.com/web/articulos/42-electronica/81-baterias-lipo>.
- [11] Quadruino.com. *Primera Prueba*. Recuperado el 15 de Abril de 2012, de <http://www.quadruino.com/guia-2/primera-prueba-brushless/primera-prueba>.
- [12] *Modelado y Control de un helicóptero Quadrotor*. Recuperado el 18 de Abril de 2012, de http://bibing.us.es/proyectos/abreproy/70017/fichero/Tesis_Master_GuilhermeRaffo.pdf.
- [13] Manual de Vuelo. *Sistemas Funcionales*. Recuperado el 18 de Abril de 2012, de <http://www.manualvuelo.com/SIF/SIF32.html>.
- [14] SPM8800 DX8. *SPM8800 con telemetría integrada*. Recuperado el 18 de Abril de 2012, de http://www.madridmodelismo.com/b8_spektrum/aero/SPM8800.html.
- [15] FUTURHOBY. *Iniciarse en aviones RC*. Recuperado el 20 de Abril de 2012, de http://www.futurhobby.com/magazine/futurhobby_magaz_000003.html.
- [16] *Recopilación de fórmulas para Aeromodelismo Eléctrico*. Recuperado el 21 de Abril de 2012, de http://www.clubalarc.com/images/formulas_erc.pdf.
- [17] Madrid Modelismo *DSM*. Recuperado el 21 de Abril de 2012, de http://www.madridmodelismo.com/f_aprende%20como/DSMX.html.
- [18] SPEKTRUM. *The Spektrum DX8*. Recuperado el 23 de Abril de 2012, de <http://www.spektrumrc.com/Products/Default.aspx?ProdID=SPM8800>.
- [19] Nurenberg Radio Insight. *Spektrum DX8*. Recuperado el 25 de Abril de 2012, de http://www.horizonhobby.co.uk/aeroonline/knews/karticles/k37_nuremberg_radio_insight/k37_radio_insight.html.

- [20] FUTURHOBY. *Programar una emisora de radio control para helicópteros RC y aviones radio control*. Recuperado el 25 de Abril de 2012, de http://www.helicopteros-radio-control.com/helicopteros_rc_000011.html.
- [21] SlideShare. *Ángulos de Euler*. Recuperado el 25 de Abril de 2012, de <http://www.slideshare.net/tito.carreras/ngulos-de-euler-1477463>.
- [22] Scribd. *Instrumentación en el sector aeronáutico*. Recuperado el 26 de Abril de 2012, de <http://es.scribd.com/doc/64504319/20/Sensores-avanzados-de-vuelo>.
- [23] *Navegación Aérea*. Recuperado el 26 de Abril de 2012, de <http://aero.us.es/na/files1112/T2NA.pdf>.
- [24] Badcock. *Motor XL*. Recuperado el 27 de Abril de 2012, de <http://www.badcock.net/MotorXL/>.
- [25] Brico Geek. *Todo sobre las baterías Lipo*. Recuperado el 28 de Abril de 2012, de <http://blog.bricogeek.com/noticias/modelismo/todo-sobre-las-baterias-lipo/>.
- [26] Todo Hobbies. *Balance de la Hélice*. Recuperado el 28 de Abril de 2012, de http://www.todohobbies.com.uy/sitios/taller/balance_de_la_helice.htm.
- [27] Mundo RC. *Balanceador de Hélices*. Recuperado el 28 de Abril de 2012, de http://www.mundorcx.com/tienda-de-aeromodelismo-bogota-colombia.html?page=shop.product_details&flypage=flypage.tpl&product_id=1233&category_id=129&keyword=balance&vmcchk=1.
- [28] *Diccionario de Términos Técnicos de Radio Control*. Recuperado el 29 de Abril de 2012, de http://www.spookycopters.com/maxrpm_00000e.html.

- [29] HobbyKing. *Motor Brushless*. Recuperado el 29 de Abril de 2012, de http://www.hobbyking.com/hobbyking/store/uh_listCategoriesAndProducts.asp?idCategory=223.
- [30] I+D Electrónica. *Tarjeta IMU acelerómetro y giróscopo 6 grados de libertad*. Recuperado el 30 de Abril de 2012, de http://www.didacticaselectronicas.com/index.php?option=com_virtuemart&page=shop.product_details&flypage=flypage.tpl&product_id=449&Itemid=6&vmcchk=1&Itemid=6.
- [31] *Electricidad básica para Aeromodelismo*. Recuperado el 30 de Abril de 2012, de http://www.webaero.net/aeromodelismo/download/ficheros/Documentacion/manual_electricidad_basica_rilos.pdf.
- [32] DynamoElectronics. *Giróscopo de tres ejes digital L3G4200D*. Recuperado el 30 de Abril de 2012, de http://dynamoelectronics.com/dynamo-tienda-virtual.html?page=shop.product_details&flypage=dynamo.tpl&product_id=937&category_id=70.
- [33] Pololu Robotics & Electronics. *CHR-6d Inertial Measurement Unit*. Recuperado el 1 de Mayo de 2012, de <http://www.pololu.com/catalog/product/1263>.
- [34] Emagister. *Teoría Motor Brushless*. Recuperado el 6 de Mayo de 2012, de http://grupos.emagister.com/documento/teoria_motor_brushless/1007-194548.
- [35] *Interfaz LabView para programar el sistema de control de Quadrotores*. Recuperado el 6 de Mayo de 2012, de <http://upcommons.upc.edu/pfc/bitstream/2099.1/10490/1/memoria.pdf>.
- [36] RC Groups. *Como seleccionar el motor eléctrico adecuado*. Recuperado el 7 de Mayo de 2012, de <http://www.rcgroups.com/forums/showthread.php?t=614629>.

- [37] MotoCalc. *The World's Best Electric Flight Calculator*. Recuperado el 10 de Mayo de 2012, de <http://www.motocalc.com/>.
- [38] NEOTEO.PLL *para motores Brushless*. Recuperado el 15 de Mayo de 2012, de <http://www.neoteo.com/pli-para-motores-brushless>.
- [39] Aparatos Voladores. *Motores Eléctricos*. Recuperado el 18 de Mayo de 2012, de <http://aparatosvoladores.mforos.com/1624922/7618637-motores-electricos/>.
- [40] *Motores Eléctricos*. Recuperado el 18 de Mayo de 2012, de <http://www.icmm.csic.es/jaalonso/velec/motor/motor.htm>.
- [41] HobbiesPR. *Brushless*. Recuperado el 20 de Mayo de 2012, de <http://www.hobbiespr.com/forums/showthread.php?t=3184>.
- [42] Die Bot Reise. *Como controlar un motor Brushless con Arduino. Tutorial*. Recuperado el 20 de Mayo de 2012, de <http://diebotreise.blogspot.com/2011/12/como-controlar-un-motor-brushless-con.html>.
- [43] Brico Geek. *Pruebas con Motores Brushless*. Recuperado el 23 de Mayo de 2012, de <http://blog.bricogeek.com/noticias/modelismo/pruebas-y-chanchullos-con-motores-brushless/>.
- [44] ARDUINO PLAYGROUND. *Timer1*. Recuperado el 24 de Mayo de 2012, de <http://arduino.cc/playground/Code/Timer1>.
- [45] ARDUINO PLAYGROUND. *Ajusting PWM Frequencies*. Recuperado el 24 de Mayo de 2012, de <http://www.arduino.cc/playground/Main/TimerPWMCheatsheet>.
- [46] ARDUINO. *Secretos del PWM en Arduino*. Recuperado el 25 de Mayo de 2012, de <http://arduino.cc/es/Tutorial/SecretsOfArduinoPWM>.

- [47] ARDUINO FORUM. *Arduino timer interrupt*. Recuperado el 25 de Mayo de 2012, de <http://arduino.cc/forum/index.php/topic,37875.0.html>.
- [48] Ardumanía. *Señales de control RC*. Recuperado el 28 de Mayo de 2012, de <http://www.ardumania.es/apendice-del-ejercicio-3-senales-de-control-rc/>.
- [49] Brico Geek. *Como leer pulsos PPM con Arduino*. Recuperado el 28 de Mayo de 2012, de <http://blog.bricogeek.com/noticias/arduino/codigo-como-leer-pulsos-ppm-con-arduino/>.
- [50] Pololu Robotics & Electronics. *LSM303DLH 3D Compass and Accelerometer Carrier with Voltage Regulators*. Recuperado el 1 de Junio de 2012, de <http://www.pololu.com/catalog/product/1250/resources>.
- [51] Hoja de datos. *LSM303DLM Sensor module:3-axis accelerometer and 3-axis magnetometer*. Recuperado el 1 de Junio de 2012, de <http://www.pololu.com/file/0J514/LSM303DLM.pdf>.
- [52] Bildr. Forum. *Arduino freezes with I2C sensor*. Recuperado el 1 de Junio de 2012, de <http://forum.bldr.org/viewtopic.php?f=20&t=323>.
- [53] ARDUINO FORUM. *New Library: LSM303DLH 3D CompassAccelerometer*. Recuperado el 5 de Junio de 2012, de <http://arduino.cc/forum/index.php/topic,38147.0.html>.
- [54] GOMEZ DE GABRIEL, J.: *Tele operación y Telerrobótica*. ALFAOMEGA. 2006.
- [55] HOLLAND, J. "Designing Autonomous Mobile Robots". ELSEVIER, 2004.
- [56] SAM GE, S. Lewis, F. L. Lewis "Autonomous Mobile Robots" CRC PRESS, 2006.

- [57] RELF, C. "ImageAcquisition and ProcessingwithLabVIEW" CRC PRESS, 2004.
- [58] SIEGWART, R. NOURBAKHSI, I. "Introduction to Autonomous Mobile Robots" THE MIT PRESS, 2004.
- [59] Performance Estimation of Interior Permanent-MagnetBrushless Motors UsingtheVoltage-Driven Flux-MMF Diagram T. J. E. Miller, *Fellow, IEEE*, MirceaPopescu, CalumCossar, and Malcolm McGilp.
- [60] Hardware-in-the-loop Simulation Framework Design For a UAV Embedded Control System Shixianjun', Songjiakun2, Lihongxing2 7-11 August, 2006, Harbin, Heilongjiang.
- [61] Integrated Mini INS Basedon MEMS Sensors for UAV Control, July 6-9 2010, Jinan, China.
- [62] A StateflowBasedSimulation of UAV Multi-mode Flight Control Liu Bo, Li Ping Institute of Industrial Process Control, Zhejiang University, Hangzhou, China(bliu2@iipc.zju.edu.cn).

DATOS DEL AUTOR

Apellidos y Nombres: Adrian Mauricio Velandia Rodríguez
Correo Electrónico: adrian.velandia@santoto.ustatunja.edu.co
Correo Electrónico alternativo: adrian.velandia1@gmail.com
Código: 3071336
No Telefónico: 312 355 0259

```

//*****PROGRAMA GENERAL*****//

///////// Eje X apuntando hacia adelante. ////////////
///////// Eje Y apuntando hacia la derecha. ////////////
///////// El eje Z apuntando hacia abajo. ////////////
///////// Positivo tono (PITCH): nariz hasta. ////////////
///////// Positivo ALABEO (roll): ala derecha hacia abajo. ////////////
///////// Positivo de guiñada (YAW): hacia la derecha. ////////////

int SENSOR_SIGN[9] = {1,1,1,-1,-1,-1,1,1,1};

///////// Eje X apuntando hacia adelante. ////////////
///////// Eje Y apuntando hacia la derecha. ////////////
///////// El eje Z apuntando hacia abajo. ////////////
///////// Positivo tono (PITCH): nariz hasta. ////////////
///////// Positivo ALABEO (roll): ala derecha hacia abajo. ////////////
///////// Positivo de guiñada (YAW): hacia la izquierda. ////////////

//LIBRERIAS//
#include <Wire.h> //COMUNICACION I2C
#include <Servo1.h> //LIBRERIA DE CONTROL DE ESC (MACKO)
#include <PID_v1.h> //LIBRERIA PID ARDUINO
#include <PID_AutoTune_v0.h> //LIBRERIA AUTOTUNE PID ARDUINO
// LSM303 accelerometer: 8 g SENSIBILIDAD
// 3.8 mg/digit; 1 g = 256
#define GRAVITY 256 //EQUIVALE A 1G EN LA TASA DE CAMBIO DEL
ACELEROMETRO

#define ToRad(x) ((x)*0.01745329252) // *pi/180
#define ToDeg(x) ((x)*57.2957795131) // *180/pi

// PARA EL GYRO L3G4200D: 2000 dps FULL ESCALA
// 70 mdps/digito; 1 dps = 0.07
#define Gyro_Gain_X 0.07 //GANANCIA EJE X
#define Gyro_Gain_Y 0.07 //GANANCIA EJE Y
#define Gyro_Gain_Z 0.07 //GANANCIA EJE Z
#define Gyro_Scaled_X(x) ((x)*ToRad(Gyro_Gain_X)) //ESCALA EL ADC DEL GIRO
#define Gyro_Scaled_Y(x) ((x)*ToRad(Gyro_Gain_Y)) //ESCALA EL ADC DEL GIRO
#define Gyro_Scaled_Z(x) ((x)*ToRad(Gyro_Gain_Z)) //ESCALA EL ADC DEL GIRO

// LSM303 CONSTANTES DE CALIBRACION DEL MAGNETOMETRO;
#define M_X_MIN -796
#define M_Y_MIN -457
#define M_Z_MIN -424
#define M_X_MAX 197

```

```

#define M_Y_MAX 535
#define M_Z_MAX 397

#define Kp_ROLLPITCH 0.02
#define Ki_ROLLPITCH 0.00002
#define Kp_YAW 1.2
#define Ki_YAW 0.00002

/*para depurar*/
//OUTPUTMODE=1 IMPRIME LOS DATOS CORRECTOS,
//OUTPUTMODE=0 IMPRIME DATOS DEL GYRO CON DERIVA
#define OUTPUTMODE 1

//#define PRINT_DCM 0 //Se imprimirá la matriz de coseno
#define PRINT_ANALOGS 0 // Se imprimirán los datos sin formato analógicos
#define PRINT_EULER 1 // Se imprimirán los angulos de Euler Roll, Pitch Yaw.
#define STATUS_LED 13 // se defien el estatus del led

float G_Dt=0.02; // TIEMPO DE INTEGRACION (ALGORITMO DCM) INTEGRACION A
50HZ**

long timer=0; //TIMER DE PROPOSITO GENERAL
long timer_old; //TIMER ANTERIOR
long timer24=0; //SEGUNDO TIMER ES USADO PARA IMPRIMIR
int AN[6]; //VECTOR QUE GUARDA LOS DATOS DEL GYRO Y DEL
ACELEROMETRO
int AN_OFFSET[6]={0,0,0,0,0,0}; // VECTOR QUE GUARDA EL OFFSET DE LOS
SENSORES

//.....VARIABLES DE LOS SENSORES.....//

int gyro_x; //VARIABLES POSICION X DEL GYRO
int gyro_y; //VARIABLES POSICION Y DEL GYRO
int gyro_z; //VARIABLES POSICION Z DEL GYRO

int accel_x; //VARIABLES POSICION X DEL GYRO
int accel_y; //VARIABLES POSICION Y DEL GYRO
int accel_z; //VARIABLES POSICION Z DEL GYRO

int magnetom_x; //VARIABLES POSICION X DEL GYRO
int magnetom_y; //VARIABLES POSICION Y DEL GYRO
int magnetom_z; //VARIABLES POSICION Z DEL GYRO

float c_magnetom_x; //CONSTANTE DE OPERACION X
float c_magnetom_y; //CONSTANTE DE OPERACION Y
float c_magnetom_z; //CONSTANTE DE OPERACION Z

```

```

float MAG_Heading; //CABECERA MAGNETOMETRO

float Accel_Vector[3]= {0,0,0}; //GUARDA LA ACELERACION EN UN VECTOR
float Gyro_Vector[3]= {0,0,0}; //GUARDA EL GYRO EN UN VECTOR
float Omega_Vector[3]= {0,0,0}; //CORRECTOS DATOS DEL VECTOR DE GYRO
float Omega_P[3]= {0,0,0}; //OMEGA PROPORCIONAL CORRECCION
float Omega_I[3]= {0,0,0}; //OMEGA INTEGRAL
float Omega[3]= {0,0,0};

//.....VARIABLES DEL RECEPTOR.....//

int canal1 = 8; //LEE CANAL1 EN EL PIN 8
int canal2 = 9; //LEE CANAL1 EN EL PIN 9
int canal3 = 10; //LEE CANAL1 EN EL PIN 10
int canal4 = 11; //LEE CANAL1 EN EL PIN 11

int ch1 = 0; //INICIALIZA LA VARIABLE CH1 EN 0
int ch2 = 0; //INICIALIZA LA VARIABLE CH2 EN 0
int ch3 = 0; //INICIALIZA LA VARIABLE CH3 EN 0
int ch4 = 0; //INICIALIZA LA VARIABLE CH4 EN 0

//.....VARIABLES DEL ALGORITMO DE ESTABILIZACION.....//

int elev=0; //INICIALIZA LA VARIABLE ELEV EN 0
int giro=0; //INICIALIZA LA VARIABLE GIRO EN 0
int cabeceo=0; //INICIALIZA LA VARIABLE CABECEO EN 0
int alabeo=0; //INICIALIZA LA VARIABLE ALABEO EN 0

float posx=0; //VARIABLE DE POSICION X//
float posy=0; //VARIABLE DE POSICION Y//
float posz=0; //VARIABLE DE POSICION Z//

float posxpro=0;
float posypro=0;
float poszpro=0;

Servo myservo1; // CREA OBJETO PARA CONTROLAR SERVO 1
Servo myservo2; // CREA OBJETO PARA CONTROLAR SERVO 2
Servo myservo3; // CREA OBJETO PARA CONTROLAR SERVO 3
Servo myservo4; // CREA OBJETO PARA CONTROLAR SERVO 4

// ANGULOS DE EULER//
float roll;
float pitch;
float yaw;

```

```

float errorRollPitch[3]= {0,0,0}; //VECTOR GUARDA ERROR DE PITCH Y ROLL
float errorYaw[3]= {0,0,0}; //VECTOR GUARDA ERROR DE YAW

unsigned int counter=0; //CONTADOR DE PROPOSITO GENERAL
byte gyro_sat=0; //+++++

float DCM_Matrix[3][3]= { //MATRIZ DE COSENOS DIRECTOS CON DIAGONAL DE UNOS
  {
    1,0,0 }
  ,{
    0,1,0 }
  ,{
    0,0,1 }
};
float Update_Matrix[3][3]={0,1,2},{3,4,5},{6,7,8}}; //MATRIZ 3 X 3 PARA GUARDAR LOS GIROS

float Temporary_Matrix[3][3]={ //MATRIZ DE PROPOSITO GENERAL
  {
    0,0,0 }
  ,{
    0,0,0 }
  ,{
    0,0,0 }
};
//VARIABLES PARA EL PROMEDIO DE LOS ANGULOS DE EULER X - Y - Z
int Lecturasx[5]; //VECTOR DE LECTURAS DEL EJE X
int ii = 0;
int Valx, Totalx = 0, Promediox = 0;

int Lecturasy[5]; //VECTOR DE LECTURAS DEL EJE Y
int iii = 0;
int Valy, Totaly = 0, Promedioy = 0;

int Lecturasz[5]; //VECTOR DE LECTURAS DEL EJE Z
int iiii = 0;
int Valz, Totalz = 0, Promedioz = 0;

void setup() //FUNCION DE CONFIGURACION
{
  Serial.begin(9600); //VELOCIDAD COMUNICACION SERIAL
  pinMode (STATUS_LED,OUTPUT); // LED STATUS
  pinMode(canal1, INPUT); // PIN 1 COMO ENTRADA
  pinMode(canal2, INPUT); // PIN 2 COMO ENTRADA
  pinMode(canal3, INPUT); // PIN 3 COMO ENTRADA

```

```

pinMode(canal4, INPUT); // PIN 4 COMO ENTRADA
myservo1.attach(40);    // DEFINICION PINES DE LOS MOTORES
myservo2.attach(42);
myservo3.attach(44);
myservo4.attach(46);
I2C_Init();            //INICIALIZACION COMUNICACION I2C

//INICIALIZACION DE VECTORES PARA PROMEDIO
for(ii=0; ii< 5; ii++)
  Lecturasx[ii] = 0;
ii=0;

for(iii=0; iii< 5; iii++)
  Lecturasy[iii] = 0;
iii=0;

for(iiii=0; iiii< 5; iiii++)
  Lecturasz[iiii] = 0;
iiii=0;

Serial.println("QUADCOPTER ARDUINO ");
Serial.println("ADRIAN MAURICION VELANDIA RODRIGUEZ ");
Serial.println("UNIVERSIDAD SANTO TOMAS ");
Serial.println("INGENIERIA ELECTRONICA ");

digitalWrite(STATUS_LED,LOW);
delay(1500);
//INICIALIZACION SENSORES
Accel_Init();
Compass_Init();
Gyro_Init();

delay(20);
// CICLO PARA EL CALIBRADO DE LOS SENSORES
for(int i=0;i<32;i++)
{
  Read_Gyro();
  Read_Accel();
  for(int y=0; y<6; y++)
    AN_OFFSET[y] += AN[y];
  delay(20);
}

for(int y=0; y<6; y++)
  AN_OFFSET[y] = AN_OFFSET[y]/32;

```

```

AN_OFFSET[5]=GRAVITY*SENSOR_SIGN[5];

for(int y=0; y<6; y++)
  Serial.println(AN_OFFSET[y]);

delay(2000);
digitalWrite(STATUS_LED,HIGH);

timer=millis();

delay(20);
counter=0;
}

void loop() //FUNCION CICLO
{
  if((millis()-timer)>=20) // CICLO A 50Hz
  {
    counter++;
    timer_old = timer;
    timer=millis();
    if (timer>timer_old)
      G_Dt = (timer-timer_old)/1000.0; // TIEMPO DE INTEGRACION DE GIROSCOPO
    else
      G_Dt = 0;

    // ADQUISICION DE DATOS PARA MATRIZ DCM
    Read_Gyro(); // LEER GIROSCOPO
    Read_Accel(); // LEER ACELEROMETRO

    if (counter > 5) // LEER COMPAS A 10Hz (5 CICLOS)
    {
      counter=0;
      Read_Compass(); // LEER MAGNETOMETRO
      Compass_Heading(); // CALCULAR NORTE MAGNETICO
    }

    // PROCESAMIENTO
    Matrix_update(); //ACTUALIZAR DCM
    Normalize(); //NORMALIZAR DCM
    Drift_correction(); //CORRECCION DE LA DERIVA (DRIFT)
    Euler_angles(); //CALCULO DE ANGULOS DE EULER
    //printdata();
    leercontrol(); //LEER RADIO CONTROL
    algoritmo(); //ALGORITOMO DE CONTROL
  }
}

```

```

    }
}

//MAGNETOMETRO

void Compass_Heading() //CALCULO RE RUMBO
{
    //DEFINICION DE CONSTANTES
    float MAG_X;
    float MAG_Y;
    float cos_roll;
    float sin_roll;
    float cos_pitch;
    float sin_pitch;

    cos_roll = cos(roll);
    sin_roll = sin(roll);
    cos_pitch = cos(pitch);
    sin_pitch = sin(pitch);

    //AJUSTE PARA EL OFFSET Y LA SENSIBILIDAD DEL COMPAS MAGNETICO

    c_magnetom_x = (float)(magnetom_x - SENSOR_SIGN[6]*M_X_MIN) / (M_X_MAX -
M_X_MIN) - SENSOR_SIGN[6]*0.5;
    c_magnetom_y = (float)(magnetom_y - SENSOR_SIGN[7]*M_Y_MIN) / (M_Y_MAX -
M_Y_MIN) - SENSOR_SIGN[7]*0.5;
    c_magnetom_z = (float)(magnetom_z - SENSOR_SIGN[8]*M_Z_MIN) / (M_Z_MAX -
M_Z_MIN) - SENSOR_SIGN[8]*0.5;

    // COMPENZACION DE INCLINACION MAGNETICA X:
    MAG_X
    c_magnetom_x*cos_pitch+c_magnetom_y*sin_roll*sin_pitch+c_magnetom_z*cos_roll*sin_pitc
h;
    // COMPENZACION DE INCLINACION MAGNETICA Y:
    MAG_Y = c_magnetom_y*cos_roll-c_magnetom_z*sin_roll;
    //RUMBO MAGNETICO
    MAG_Heading = atan2(-MAG_Y,MAG_X);
}

//FUNCIONES MATRIZ DE COSENOS DIRECTOS
/*****/
void Normalize(void) //FUNCION DE NORMALIZACION DE LA MATRIZ DCM
{

```

```

float error=0;
float temporary[3][3];
float renorm=0;

error= -Vector_Dot_Product(&DCM_Matrix[0][0],&DCM_Matrix[1][0]).5;

Vector_Scale(&temporary[0][0], &DCM_Matrix[1][0], error);
Vector_Scale(&temporary[1][0], &DCM_Matrix[0][0], error);

Vector_Add(&temporary[0][0], &temporary[0][0], &DCM_Matrix[0][0]);
Vector_Add(&temporary[1][0], &temporary[1][0], &DCM_Matrix[1][0]);

Vector_Cross_Product(&temporary[2][0],&temporary[0][0],&temporary[1][0]);

renorm= .5 *(3 - Vector_Dot_Product(&temporary[0][0],&temporary[0][0]));
Vector_Scale(&DCM_Matrix[0][0], &temporary[0][0], renorm);

renorm= .5 *(3 - Vector_Dot_Product(&temporary[1][0],&temporary[1][0]));
Vector_Scale(&DCM_Matrix[1][0], &temporary[1][0], renorm);

renorm= .5 *(3 - Vector_Dot_Product(&temporary[2][0],&temporary[2][0]));
Vector_Scale(&DCM_Matrix[2][0], &temporary[2][0], renorm);
}

/*****/
void Drift_correction(void) //COMPENZACION DE LA DERIVA DE ROLL, PITCH Y YAW
{
//DEFINICION DE VARIABLES
float mag_heading_x;
float mag_heading_y;
float errorCourse;

static float Scaled_Omega_P[3];
static float Scaled_Omega_I[3];
float Accel_magnitude;
float Accel_weight;

//*****ROLL Y PITCH*****/

// CALCULO DE LA MAGNITUD DEL VECTOR DE ACELERACION
Accel_magnitude = sqrt(Accel_Vector[0]*Accel_Vector[0] +
Accel_Vector[1]*Accel_Vector[1] + Accel_Vector[2]*Accel_Vector[2]);
Accel_magnitude = Accel_magnitude / GRAVITY; // ESCALA DE FUERZA DE
GRAVEDAD
// (<0.5G = 0.0, 1G = 1.0 , >1.5G = 0.0)
Accel_weight = constrain(1 - 2*abs(1 - Accel_magnitude),0,1); //

```

```

Vector_Cross_Product(&errorRollPitch[0],&Accel_Vector[0],&DCM_Matrix[2][0]); //adjust
the ground of reference
Vector_Scale(&Omega_P[0],&errorRollPitch[0],Kp_ROLLPITCH*Accel_weight);

Vector_Scale(&Scaled_Omega_I[0],&errorRollPitch[0],Ki_ROLLPITCH*Accel_weight);
Vector_Add(Omega_I,Omega_I,Scaled_Omega_I);

//*****YAW*****
// LA CORRECCION DE LA DRIVA DE YOW ES REALIZADA BASADO EN EL
COMPAS MAGNETICO
We make the gyro YAW drift correction based on compass magnetic heading

mag_heading_x = cos(MAG_Heading);
mag_heading_y = sin(MAG_Heading);
errorCourse=(DCM_Matrix[0][0]*mag_heading_y) - (DCM_Matrix[1][0]*mag_heading_x);
//COLCULO DEL ERROR DE YAW
Vector_Scale(errorYaw,&DCM_Matrix[2][0],errorCourse); //APLICACION DE LA
CORRECCION DE YAW PARA LA ROTACION DE X Y y Z DEL QUAD, DEPENDIENDO
DE LA POSICION
Vector_Scale(&Scaled_Omega_P[0],&errorYaw[0],Kp_YAW);//.01          CONSTANTE
PROPORCIONAL
Vector_Add(Omega_P,Omega_P,Scaled_Omega_P);//SUMA          DE          CONSTANTE
PROPORCIONAL

Vector_Scale(&Scaled_Omega_I[0],&errorYaw[0],Ki_YAW);//.00001 INTEGRADOR
Vector_Add(Omega_I,Omega_I,Scaled_Omega_I);//SUMA INTEGRADOR A Omega_I
}

void Matrix_update(void) //ACTUALIZACION DE LA MATRIZ
{
Gyro_Vector[0]=Gyro_Scaled_X(gyro_x); //gyro x roll
Gyro_Vector[1]=Gyro_Scaled_Y(gyro_y); //gyro y pitch
Gyro_Vector[2]=Gyro_Scaled_Z(gyro_z); //gyro Z yaw

Accel_Vector[0]=accel_x;
Accel_Vector[1]=accel_y;
Accel_Vector[2]=accel_z;

Vector_Add(&Omega[0], &Gyro_Vector[0], &Omega_I[0]); //SUMA DE CONSTANTE
PROPORCIONAL
Vector_Add(&Omega_Vector[0], &Omega[0], &Omega_P[0]); //SUMA CONSTANTE
INTEGRAL

#if OUTPUTMODE==1 //ACTUALIZACION DE LA MATRIZ SIN DERIVA
Update_Matrix[0][0]=0;

```

```

Update_Matrix[0][1]=-G_Dt*Omega_Vector[2];//-z
Update_Matrix[0][2]=G_Dt*Omega_Vector[1];//y
Update_Matrix[1][0]=G_Dt*Omega_Vector[2];//z
Update_Matrix[1][1]=0;
Update_Matrix[1][2]=-G_Dt*Omega_Vector[0];//-x
Update_Matrix[2][0]=-G_Dt*Omega_Vector[1];//-y
Update_Matrix[2][1]=G_Dt*Omega_Vector[0];//x
Update_Matrix[2][2]=0;
#else //ACTUALIZACION DE LA MATRIZ SIN DERIVA CON DERIVA
Update_Matrix[0][0]=0;
Update_Matrix[0][1]=-G_Dt*Gyro_Vector[2];//-z
Update_Matrix[0][2]=G_Dt*Gyro_Vector[1];//y
Update_Matrix[1][0]=G_Dt*Gyro_Vector[2];//z
Update_Matrix[1][1]=0;
Update_Matrix[1][2]=-G_Dt*Gyro_Vector[0];
Update_Matrix[2][0]=-G_Dt*Gyro_Vector[1];
Update_Matrix[2][1]=G_Dt*Gyro_Vector[0];
Update_Matrix[2][2]=0;
#endif

Matrix_Multiply(DCM_Matrix,Update_Matrix,Temporary_Matrix); //RESULTADO DE LA
MATRIZ

for(int x=0; x<3; x++) //MATRIZ ADICIONAL
{
  for(int y=0; y<3; y++)
  {
    DCM_Matrix[x][y]+=Temporary_Matrix[x][y];
  }
}

void Euler_angles(void) //CALCULO DE ANGULOS DE EULER
{
  pitch = -asin(DCM_Matrix[2][0]);
  roll = atan2(DCM_Matrix[2][1],DCM_Matrix[2][2]);
  yaw = atan2(DCM_Matrix[1][0],DCM_Matrix[0][0]);
}
//COMUNICACIÓN I2C

#include <L3G4200D.h>
#include <LSM303.h> //LIBRERIAS SENSORES

L3G4200D gyro;
LSM303 compass; //DEFINICION DE SENSORES

```

```

void I2C_Init() //FUNCION DE INICIALIZACION DE PROTOCOLO I2C
{
  Wire.begin();
}

void Gyro_Init() //INICIALIZACION DEL GIROSCOPO Y CONFIGURACION DE
REGISTROS
{
  gyro.writeReg(L3G4200D_CTRL_REG1, 0x0F); //MODO NORMAL, TODOS LOS EJES
CONECTADOS 100 Hz
  gyro.writeReg(L3G4200D_CTRL_REG4, 0x20); // 2000 dps FULL ESCALA
}

void Read_Gyro() //LECTURA Y ALMACENAMIENTO DE DATOS
{
  gyro.read();

  AN[0] = gyro.g.x;
  AN[1] = gyro.g.y;
  AN[2] = gyro.g.z;
  gyro_x = SENSOR_SIGN[0] * (AN[0] - AN_OFFSET[0]);
  gyro_y = SENSOR_SIGN[1] * (AN[1] - AN_OFFSET[1]);
  gyro_z = SENSOR_SIGN[2] * (AN[2] - AN_OFFSET[2]);
}

void Accel_Init() //INICIALIZACION DEL ACELEROMETRO Y CONFIGURACION DE
REGISTROS
{
  compass.writeAccReg(LSM303_CTRL_REG1_A, 0x27); // MODO NORMAL, TODOS LOS
EJES CONECTADOS 50 Hz

  compass.writeAccReg(LSM303_CTRL_REG4_A, 0x30); // 8 g FULL ESCALA
}

void Read_Accel() //LECTURA Y ALMACENAMIENTO DE DATOS
{
  compass.readAcc();

  AN[3] = compass.a.x;
  AN[4] = compass.a.y;
  AN[5] = compass.a.z;
  accel_x = SENSOR_SIGN[3] * (AN[3] - AN_OFFSET[3]);
  accel_y = SENSOR_SIGN[4] * (AN[4] - AN_OFFSET[4]);
  accel_z = SENSOR_SIGN[5] * (AN[5] - AN_OFFSET[5]);
}

```

```

void Compass_Init() //INICIALIZACION DEL MAGNETOMETRO Y CONFIGURACION
DE REGISTROS
{
  compass.init();
  compass.writeMagReg(LSM303_MR_REG_M, 0x00); //MODO DE CONVERSION
CONTINUA // 15 Hz default
}

void Read_Compass() //LECTURA Y ALMACENAMIENTO DE DATOS
{
  compass.readMag();

  magnetom_x = SENSOR_SIGN[6] * compass.m.x;
  magnetom_y = SENSOR_SIGN[7] * compass.m.y;
  magnetom_z = SENSOR_SIGN[8] * compass.m.z;
}

// IMPRESIÓN DE DATOS

void printdata(void) IMPRIMIR DATOS
{
  Serial.print("...");

  #if PRINT_EULER == 1 //IMPRIMIR ANGULOS DE EULER
  Serial.print("ANG:");
  Serial.print(ToDeg(roll));
  Serial.print(",");
  Serial.print(ToDeg(pitch));
  Serial.print(",");
  Serial.print(ToDeg(yaw));
  #endif
  #if PRINT_ANALOGS==1 //IMPRIMIR DATOS SIN TRATAMIENTO
  Serial.print(",AN:");
  Serial.print(AN[0]); //LECTURA DEL ADC
  Serial.print(",");
  Serial.print(AN[1]);
  Serial.print(",");
  Serial.print(AN[2]);
  Serial.print(",");
  Serial.print(AN[3]);
  Serial.print(",");
  Serial.print(AN[4]);
  Serial.print(",");
  Serial.print(AN[5]);
  Serial.print(",");

```

```

Serial.print(c_magnetom_x);
Serial.print(",");
Serial.print(c_magnetom_y);
Serial.print(",");
Serial.print(c_magnetom_z);
#endif
/*#if PRINT_DCM == 1 //IMPRIMIR RESULTADO MATRIZ DE COSENOS
DIRECTOS
Serial.print(",DCM:");
Serial.print(convert_to_dec(DCM_Matrix[0][0]));
Serial.print(",");
Serial.print(convert_to_dec(DCM_Matrix[0][1]));
Serial.print(",");
Serial.print(convert_to_dec(DCM_Matrix[0][2]));
Serial.print(",");
Serial.print(convert_to_dec(DCM_Matrix[1][0]));
Serial.print(",");
Serial.print(convert_to_dec(DCM_Matrix[1][1]));
Serial.print(",");
Serial.print(convert_to_dec(DCM_Matrix[1][2]));
Serial.print(",");
Serial.print(convert_to_dec(DCM_Matrix[2][0]));
Serial.print(",");
Serial.print(convert_to_dec(DCM_Matrix[2][1]));
Serial.print(",");
Serial.print(convert_to_dec(DCM_Matrix[2][2]));
#endif*/
Serial.println();

}

long convert_to_dec(float x) // CONVERTIR A DECIMAL
{
return x*10000000;
}

// DEFINICION DE FUNICIONES DE OPERACIONES ENTRE VECTORES//

float Vector_Dot_Product(float vector1[3],float vector2[3])//PRODUCTO PUNTO
{
float op=0;

for(int c=0; c<3; c++)
{
op+=vector1[c]*vector2[c];
}
}

```

```

return op;
}

void Vector_Cross_Product(float vectorOut[3], float v1[3],float v2[3]) //PRODUCTO CRUZ
{
vectorOut[0]= (v1[1]*v2[2]) - (v1[2]*v2[1]);
vectorOut[1]= (v1[2]*v2[0]) - (v1[0]*v2[2]);
vectorOut[2]= (v1[0]*v2[1]) - (v1[1]*v2[0]);
}

void Vector_Scale(float vectorOut[3],float vectorIn[3], float scale2) //PRODUCTO DE UN
VECTOR POR UN ESCALAR
{
for(int c=0; c<3; c++)
{
vectorOut[c]=vectorIn[c]*scale2;
}
}

void Vector_Add(float vectorOut[3],float vectorIn1[3], float vectorIn2[3]) //SUMA DE
VECTORES
{
for(int c=0; c<3; c++)
{
vectorOut[c]=vectorIn1[c]+vectorIn2[c];
}
}
//FUNCIONES DE CONTROL Y PROCESAMIENTO

void leercontrol(void){ //FUNCION DE LETURA Y ACONDICIONAMIENTO DE DATOS
DEL RADIO CONTROL

ch1=pulseIn(canal1 , HIGH); //LECTURA DE LA SEÑAL PWM PROVENIENTE DEL
RECEPTOR DEL RADIO CONTROL
ch2=pulseIn(canal2 , HIGH);
ch3=pulseIn(canal3 , HIGH);
ch4=pulseIn(canal4 , HIGH);

constrain (ch1,1100,1880); //ASIGNACION DE LIMITES DE LA LECTURA
elev=map(ch1,1100,1880,0,256); //ASIGNACION DE ESCALA A LOS DATOS LEIDOS
constrain (ch4,1075,1880);
giro=map(ch4,1075,1880,0,256);
constrain (ch3,1080,1880);
cabeceo=map(ch3,1080,1880,0,256);
constrain (ch2,1080,1880);
alabeo=map(ch2,1080,1880,0,256);
/*Serial.print("ch1:"); //IMPRESION DE DATOS PARA PRUEBAS

```

```

Serial.print(elev);
Serial.print(" ch2:");
Serial.print(alabeo);
Serial.print(" ch3:");
Serial.print(cabeceo);
Serial.print(" ch4:");
Serial.println(giro);*/

/*Serial.print("ch1:");
Serial.print(ch1);
Serial.print(" ch2:");
Serial.print(ch2);
Serial.print(" ch3:");
Serial.print(ch3);
Serial.print(" ch4:");
Serial.println(ch4);*/
}

void algoritmo(){ //ALGORITMO DE CONTROL

int n=1; //DEFINICION DE VARIABLES
posx=ToDeg(roll);
posy=ToDeg(pitch);
posz=ToDeg(yaw);

Totalx=Totalx-Lecturasx[ii];
//SUSTRAE EL TOTAL CON LA LECTURA ANTERIOR
Lecturasx[ii] = posx;
//REALIZA LA SUMATORIA ENTRE LECTURAS
Totalx = Totalx + Lecturasx[ii];
ii = ii + 1;
//CALCULA EL PROMEDIO Y ES GUARDADO EN UNA VARIABLE PARA SU MANEJO
POSTERIOR
if (ii >= 5){
ii = 0;
Promediox = Totalx / 5;

}

Totaly=Totaly-Lecturasy[iii];
//SUSTRAE EL TOTAL CON LA LECTURA ANTERIOR
Lecturasy[iii] = posy;
//REALIZA LA SUMATORIA ENTRE LECTURAS
Totaly = Totaly + Lecturasy[iii];
iii = iii + 1;
//CALCULA EL PROMEDIO Y ES GUARDADO EN UNA VARIABLE PARA SU

```

MANEJO POSTERIOR

```

if (iii >= 5){
  iii = 0;
  Promedioy = Totaly / 5;

```

```

}

```

```

Totalz=Totalz-Lecturasz[iiii];

```

```

//SUSTRAE EL TOTAL CON LA LECTURA ANTERIOR

```

```

Lecturasz[iiii] = posz;

```

```

//REALIZA LA SUMATORIA ENTRE LECTURAS

```

```

Totalz = Totalz + Lecturasz[iiii];

```

```

iiii = iii + 1;

```

```

//CALCULA EL PROMEDIO Y ES GUARDADO EN UNA VARIABLE PARA SU MANEJO

```

POSTERIOR

```

if (iiii >= 5){

```

```

  iii = 0;

```

```

  Promedioz = Totalz / 5;

```

```

}

```

```

if ((Promedioz==0) && (Promedioy==0)){ //FUNCIONES O REGLAS DE DECISION

```

LOGICO DIFUSAS

```

  myservo1.write(elev + n);

```

```

  myservo2.write(elev + n);

```

```

  myservo3.write(elev + n);

```

```

  myservo4.write(elev + n);

```

```

}

```

```

if ((Promedioz==0) && (Promedioy>0)){

```

```

  myservo1.write(elev + n);//

```

```

  myservo2.write(elev + n);

```

```

  myservo3.write(elev + n);//

```

```

  myservo4.write(elev - n);

```

```

}

```

```

if ((Promedioz==0) && (Promedioy<0)){

```

```

  myservo1.write(elev + n);//

```

```

  myservo2.write(elev - n);

```

```

  myservo3.write(elev + n);//

```

```

  myservo4.write(elev + n);

```

```

}

```

```

if ((Promedioz>0) && (Promedioy==0)){

```

```

  myservo1.write(elev + n);

```

```

  myservo2.write(elev + n);//

```

```

  myservo3.write(elev - n);

```

```

  myservo4.write(elev + n);//

```

```

}

```

```

if ((Promedioz<0) && (Promedioy==0)){

```

```

  myservo1.write(elev - n);

```

```

myservo2.write(elev + n);//
myservo3.write(elev + n);
myservo4.write(elev + n);//
}

if ((Promediox<0) && (Promedioy<0)){
  myservo1.write(elev - n);
  myservo2.write(elev - n);
  myservo3.write(elev + n);
  myservo4.write(elev + n);
}
if ((Promediox<0) && (Promedioy>0)){
  myservo1.write(elev - n);
  myservo2.write(elev + n);
  myservo3.write(elev + n);
  myservo4.write(elev - n);
}
if ((Promediox>0) && (Promedioy<0)){
  myservo1.write(elev + n);
  myservo2.write(elev - n);
  myservo3.write(elev - n);
  myservo4.write(elev + n);
}
if ((Promediox>0) && (Promedioy>0)){
  myservo1.write(elev + n);
  myservo2.write(elev + n);
  myservo3.write(elev - n);
  myservo4.write(elev - n);
}
float tiempo = millis(); //VERIFICACION DEL TIEMPO TRASCURRIDO EN EL
PROGRAMA
Serial.println (tiempo);
//Serial.print("posx:");

//Serial.print("x:"); //IMPRESION DE DATOS PARA VERIFICACION DE CODIGO
/*Serial.print(Promediox);
Serial.print(",");
Serial.print(Promedioy);
Serial.print(",");
Serial.println(Promedioz);*/
}

//DEFINICION DE LA FUNCION PRODUCTO DE MATRICES
void Matrix_Multiply(float a[3][3], float b[3][3],float mat[3][3]) //MULTIPLICACION DE
MATRICES
{

```

```
float op[3];
for(int x=0; x<3; x++)
{
  for(int y=0; y<3; y++)
  {
    for(int w=0; w<3; w++)
    {
      op[w]=a[x][w]*b[w][y];
    }
    mat[x][y]=0;
    mat[x][y]=op[0]+op[1]+op[2];

    float test=mat[x][y];
  }
}
}
```