



**UNIVERSIDAD POLITÉCNICA SALESIANA**  
**SEDE QUITO**  
**CARRERA DE INGENIERÍA ELECTRÓNICA**

**CONTROL RETROALIMENTADO PARA UN SISTEMA DE ATERRIZAJE  
Y DESPEGUE VERTICAL (VTOL) Y VISUALIZACIÓN DE  
PARAMETROS DESDE LA NUBE**

Trabajo de titulación previo a la obtención del  
Título de Ingeniero Electrónico

AUTORES: Dennis Adrian Mendoza Loya  
Jefferson Rolando Zapata Peña

TUTOR: Danilo Sebastián Sigcha Tipán

Quito - Ecuador

2022

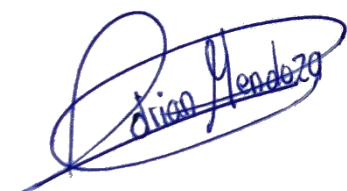
**CERTIFICADO DE RESPONSABILIDAD Y  
AUTORÍA DEL TRABAJO DE TITULACIÓN**

Nosotros, Mendoza Loya Dennis Adrian, con documento de identificación N° 1718488156 y Zapata Peña Jefferson Rolando con documento de identificación N° 1720949260; manifestamos que:

Somos los autores y responsables del presente trabajo; y, autorizamos a que sin fines de lucro la Universidad Politécnica Salesiana pueda usar, difundir, reproducir o publicar de manera total o parcial el presente trabajo de titulación.

Quito, 30 de marzo del año 2022

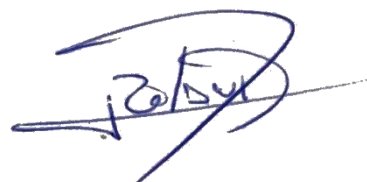
Atentamente,



---

Dennis Adrián Mendoza Loya

1718488156



---

Jefferson Rolando Zapata Peña

1720949260

**CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL  
TRABAJO DE TITULACIÓN A LA UNIVERSIDAD POLITÉCNICA  
SALESIANA**

Nosotros, Dennis Adrián Mendoza Loya, con documento de identificación N° 1718488156 y Jefferson Rolando Zapata Peña con documento de identificación N° 1720949260 , expresamos nuestra voluntad y cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos los autores del trabajo de titulación: “ Control retroalimentado para un sistema de aterrizaje y despegue vertical (VTOL) y visualización de parámetros desde la nube.”, el cual ha sido desarrollado para optar por el título de: Ingeniero Electrónico , en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente de los derechos cedidos anteriormente.

En concordancia con lo manifestado, suscribimos este documento en el momento que hacemos la entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana

Quito, 30 de marzo del año 2022

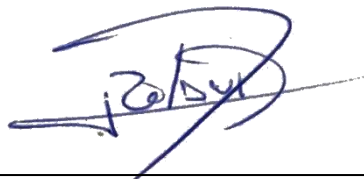
Atentamente,



---

Dennis Adrián Mendoza Loya

1718488156



---

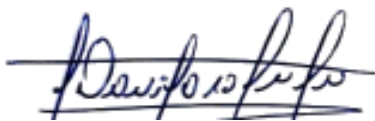
Jefferson Rolando Zapata Peña

1720949260

## CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN

Yo Danilo Sebastián Sigcha Tipán con documento de identificación N° 1721770343, docente de la Universidad Politécnica Salesiana declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: CONTROL RETROALIMENTADO PARA UN SISTEMA DE ATERRIZAJE Y DESPEGUE VERTICAL (VTOL) Y VISUALIZACIÓN DE PARAMETROS DESDE LA NUBE, realizado por Dennis Adrián Mendoza Loya, con documento de identificación N°1718488156 y por Jefferson Rolando Zapata Peña, con documento de identificación N°1720949260 obteniendo como resultado final el trabajo de titulación bajo la opción Proyecto Técnico que cumple con todos los requisitos determinados por la Universidad Politécnica Salesiana.

Quito, 30 de marzo del año 2022



---

Ing. Danilo Sebastián Sigcha Tipán, Mtr

1721770343

## DEDICATORIA

El presente trabajo se le dedico a las mujeres más importantes en mi vida: a mi madre Ana Loya, quien con su amor, esfuerzo y paciencia me ha permitido cumplir hoy uno de mis sueños, por ser el pilar más importante en mi vida. A mi abuelita Isabel, a pesar de ya no estar conmigo, siempre confió en mí y nunca me abandono en el proceso de mi formación. A mi hermana Johanna Mendoza, por llenar mi vida con sus valiosos consejos y siempre creer en mí y nunca dejarme caer en los momentos más difíciles.

*Adrián Mendoza*

El presente trabajo se lo dedico a mis padres por todo el apoyo que me han sabido brindar y guiarme para cumplir hoy día un logro más en la vida. Y en general a toda mi familia que directa e indirectamente han sabido apoyando y estar ahí cuando he necesitado de su apoyo

*Jefferson Zapata*

## **AGRADECIMIENTO**

Agradezco a Dios por brindarme una gran familia donde nunca falta el amor y el apoyo de cada uno de ellos, por unos buenos abuelos que siempre quisieron lo mejor para mí y a mis tíos que también me aconsejaron en momentos difíciles, A mi persona favorita July por siempre creer en mí y los consejos y palabras de apoyo que nunca faltaron en la elaboración de este proyecto, a todos mis amigos que estuvieron desde el comienzo de este gran sueño.

Agradezco al Ing. Danilo Sigcha por brindarnos y compartimos sus conocimientos para orientarnos en el desarrollo de este proyecto.

*Adrian Mendoza*

Agradezco a Dios por todas las personas que en el transcurso de mi carrera fueron pasando y fueron de gran apoyo, a todos los ingenieros que supieron compartir su conocimiento y consejos. A mis padres y hermano que siempre me extendieron una mano y lo seguirán haciendo. Al Ing Danilo Sigcha por saber guiarnos y compartir sus conocimientos en este proyecto de culminación de estudios y por último agradecer a mi compañero de tesis por sus aportes en el desarrollo de este proyecto

*Jefferson Zapata*

## INDICE

CESIÓN DE DERECHOS DE AUTOR .....	i
DECLARATORIA DE COAUTORÍA DEL DOCENTE TUTOR .....	ii
DEDICATORIA.....	iii
AGRADECIMIENTO .....	iv
ÍNDICE DE FIGURAS .....	viii
ÍNDICE DE TABLAS.....	x
RESUMEN .....	xi
ABSTRACT .....	xii
CAPÍTULO 1 ANTECEDENTES .....	1
1.1 Planteamiento .....	del
problema.....	1
1.2 Justificación.....	2
1.3 Objetivos .....	3
1.3.1 Objetivo General .....	3
1.3.2 Objetivos Específicos.....	3
1.4 Metodología .....	4
1.4.1 Metodología Inductiva .....	4
1.4.2 Metodología Seccional y Longitudinal .....	4
1.4.3 Metodología Analítica.....	4
1.4.4 Metodología Experimental.....	4
CAPITULO 2 FUNDAMENTACIÒN TEÒRICA .....	5
2.1 Vertical take-off and landing (VTOL) .....	5
2.2 Dashboard .....	6
2.3 Control PID .....	7

2.4 Lógica difusa.....	7
2.5 Interfaz hombre-maquina (HMI).....	8
2.6 Telemetría .....	9
2.7 Motor brushless.....	9
2.8 Hélice .....	9
2.9 Electronic speed control ( ESC 30 A ).....	10
2.10 Arduino mega 2560.....	11
2.11 Tarjeta ESP8266 NodeMCU.....	12
2.12 Sensor HC-SR04.....	12
2.13 Sensor ACS712.....	13
2.14 Fuente de poder ac-dc 12v 30a 360w.....	13
2.15 Power Analyzer prova WM-02 .....	14
CAPÍTULO 3 DISEÑO E IMPLEMETACIÓN .....	16
3.1 Descripción del hardware del sistema de aterrizaje y despegue vertical (VTOL).....	16
3.2 Diseño de la función de transferencia del motor brushless.....	17
3.2.1 Diseño del controlador pid.....	19
3.3 Diseño del controlador fuzzy .....	20
3.3.1 Diseño de los conjuntos difusos.....	21
3.3.2 Fusificación en la entrada.....	23
3.3.3 Controlador difuso en tiempo real.....	23
3.3.4 Visualización remota.....	24
3.3.5 Programación del módulo ESP8266MOD con el servidor ThingSpeak.....	25
3.3.6 Interfaz Hombre-Maquina (HMI) .....	25



CAPÍTULO 4 .....	32
ANALISIS DE RESULTADOS .....	32
4.1 Controlador PID y FUZZY sin masa .....	32
4.2 Controlador PID Y FUZZY con masa = 20 gr.....	34
4.3 Controlador PID Y FUZZY con masa = 40 gr.....	34
4.5 Consumo de energía del procesador .....	36
CONCLUSIONES .....	37
RECOMENDACIONES .....	38
REFERENCIAS BIBLIOGRÁFICAS .....	39
ANEXOS .....	42

## ÍNDICE DE FIGURAS

Figura.2.1-1 Sistema VTOL.....	5
Figura.2.2-1 Dashboard en la plataforma thinkspeak.....	6
Figura.2.3-1 Estructura de un controlador PID con retroalimentación.....	7
Figura.2.4-1 Creación de los conjuntos difusos del controlador fuzzy.....	8
Figura.2.7-1 Motor Brushless.....	9
Figura.2.8-1 Hélices ft 1045.....	10
Figura.2.9-1 Driver ESC 30 A.....	10
Figura.2.10-1 Arduino Mega 2560 .....	11
Figura.2.11-1 Módulo ESP8266MOD.....	12
Figura.2.12-1 Sensor HC-SR04.....	12
Figura.2.13-1 Sensor ACS712.....	13
Figura.2.14-1 Fuente de poder AC/DC.....	14
Figura.2.15-1 Power analyzer prova WM-02.....	14
Figura.3.1-1 Diagrama de bloques sistema VTOL.....	15
Figura.3.2-1 Sistema en lazo abierto.....	16
Figura.3.2-2 Importación de datos en entradas y salidas.....	16
Figura.3.2-3 Estimación de datos del motor brushless.....	17
Figura.3.2-4 Ejecución de la variable en el command window.....	17
Figura.3.2-5 Grafica de la función de transferencia para la obtención de las ganancias.....	18
Figura.3.2.1-1 Estructura de un controlador PID.....	19
Figura.3.2.1-2 Implementacion del controlador PID.....	19
Figura.3.3-1 Proceso de un controlador difuso.....	20
Figura.3.3.1-1 Conjuntos difusos de la entrada altura.....	21
Figura.3.3.1-2 Conjuntos difusos de la salida RPM.....	21
Figura.3.3.2-1 Fusificación de la entrada del controlador fuzzy.....	22
Figura.3.3.3-1 Controlador difuso en tiempo real.....	23
Figura.3.3.4-1 Dashboard en thingspeak.....	24
Figura.3.3.6-1 Interfaz gráfica del sistema VTOL.....	25
Figura.3.3.6-2 Interfaz gráfica en tiempo real del sistema VTOL.....	26
Figura.3.3.6-3 Interfaz gráfica en simulación sistema VTOL.....	26
Figura.3.3.6-4 Interfaz gráfica de la ficha técnica.....	26
Figura.3.3.6-5 Interfaz gráfica del controlador FUZZY tiempo real.....	27
Figura.3.3.6-6 Comportamiento del controlador fuzzy.....	28
Figura.3.3.6-7 Datos del controlador FUZZY en thingSpeak en tiempo real.....	28

## ÍNDICE DE TABLAS

Tabla.3.2-1 Valores de ganancias del controlador PID.....	18
Tabla.3.3.1-1 Reglas del controlador difuso.....	21
Tabla.3.3.7-1 Parámetros de Arduino.....	29
Tabla.3.3.8-1 Parámetros del módulo WIFI.....	29
Tabla.3.3.9-1 Parámetros del sensor de corriente.....	30
Tabla.3.3.10-1 Parámetros del sensor de distancia.....	30
Tabla.4.1-1 Tiempo de subida, consumo de corriente y potencia de los controladores PID y FUZZY sin masa.....	31
Tabla.4.2-1 Consumo de corriente con masa = 20gr.....	32
Tabla.4.3-1 Consumo de potencia de los controladores PID y FUZZY con masa....	33
Tabla.4.5-1 Datos de potencia del CPU.....	35

## RESUMEN

Se presenta un sistema VTOL (vertical take-off landing) para interactuar y analizar el comportamiento del control retroalimentado para un sistema de aterrizaje y despegue vertical VTOL, en el cual se implementa dos controles: el control clásico PID (proporcional integral derivativo) y el control FUZZY. El diseño propuesto consta de un conjunto de módulos ESC (electronic speed controller ) y ARDUINO, los cuales permitirán el control de altura y estabilidad del motor brushless mediante el software de Matlab y el Simulink. Para visualizar las gráficas del módulo se diseñó una interfaz HMI, para que a través de ella se pueda manipular el set point y observar los cambios que se producen en los controlares PID y FUZZY, al tener construido el sistema VTOL con su correcto funcionamiento, se procedió a crear el sistema remoto donde se realizó la programación, entre el módulo ESP8266MOD y el servidor ThingSpeak, en el cual nos permitirá visualizar el comportamiento de los controladores en tiempo real, a través de graficas que nos indicaran: altura, corriente y potencia. Adicional se incorporó al sistema remoto una cámara para observar en tiempo real el funcionamiento del módulo VTOL, el cual beneficiara a los estudiantes de la carrera de ingeniería Electrónica y Automatización de la Universidad Politécnica Salesiana Sede Quito Campus Sur.

## **ABSTRACT**

A VTOL (vertical takeoff landing) system is presented to interact and analyze the behavior of the feedback control for a VTOL vertical takeoff and landing system, in which two controls are implemented: the classic PID (proportional integral derivative) control and the FUZZY control. . The proposed design consists of a set of ESC (electronic speed controller) and ARDUINO modules, which will allow the height and stability control of the brushless motor through Matlab and Simulink software. To visualize the graphs of the module, an HMI interface was designed, so that through it the set point can be manipulated and the changes that occur in the PID and FUZZY controls can be observed, having built the VTOL system with its correct operation, proceeded to create the remote system where the programming was carried out, between the ESP8266MOD module and the ThingSpeak server, in which it will allow us to visualize the behavior of the controllers in real time, through graphs that will indicate: height, current and power. Additionally, a camera was incorporated into the remote system to observe in real time the operation of the VTOL module, which will benefit the students of the Electronic Engineering and Automation career of the Salesian Polytechnic University, Quito Campus Sur.

## INTRODUCCIÓN

Hoy en día, los diseños de aeronaves dependen en gran medida de sistemas automatizados que permiten controlar y monitorear muchos subsistemas. Hay varios factores importantes que deben abordarse para el equilibrio de la aeronave, entre ellos el peso, la forma, el diseño del controlador, entre otros. (Fatima, Idrisi, & Mourad, 2020)

El desarrollo del CONTROL RETROALIMENTADO PARA UN SISTEMA DE ATERRIZAJE Y DESPEGUE VERTICAL (VTOL) Y VISUALIZACION DE PARAMETROS DESDE LA NUBE para el laboratorio de teoría de control de la UNIVERSIDAD POLITECNICA SALESIANA, tiene como objetivo brindar a los estudiantes de ingeniería electrónica el desarrollo de ensayos de controles de forma vertical, permitiendo que se pueda poner en práctica las materias de teoría de control I, II, III e informática industrial, en las cuales se imparten las bases para el desarrollo de estos controladores de modelamiento e implementación de este sistema, cabe recalcar que se creó una interfaz gráfica mediante el software guide de Matlab, donde se visualizara un panel que contiene los controladores en tiempo real y simulación, también abarca una hoja técnica de cada componente que conforma el módulo VTOL

Por situaciones sanitarias que atraviesa el mundo y el país, se llevó a cabo la implementación de un sistema remoto al módulo VTOL, el cual permitirá al estudiante realizar, ensayos en los controladores desde la comodidad de su hogar y observar el comportamiento de la planta a través de la plataforma ThingSpeak, en donde se podrá visualizar datos de: Altura, corriente y potencia que genera cada uno de los controladores en tiempo real.

# CAPÍTULO 1

## ANTECEDENTES

### 1.1 Planteamiento del problema

La Universidad Politécnica Salesiana en la carrera de Ingeniería Electrónica, en el laboratorio de Teoría de Control cuenta con equipos: TRMS, Péndulo Invertido, Tanque de nivel, Ball and Plate donde se implementan sistemas de control: PID, PI, PD, en dicho laboratorio no se dispone con un sistema VTOL, el cual permitirá el análisis de diferentes parámetros de forma vertical, el factor de amortiguamiento, perturbaciones internas y externas, las ganancias proporcional, integral y derivativa ( $K_p$ ,  $K_i$  y  $K_d$ ) y el tiempo de establecimiento. De la misma manera se tendrá el análisis de la lógica difusa que consiste en conjuntos difusos, operaciones de conjuntos, reglas de la lógica difusa y reglas de inferencia

De lo antes expuesto surge la necesidad de desarrollar un sistema VTOL para la Universidad Politécnica Salesiana, la misma que requiere de módulos para el laboratorio que permitan analizar y estudiar un control estático, visualizando los ángulos de orientación, posicionamiento horizontal y un control híbrido.

Por ende, el módulo permitirá a los estudiantes de teoría de control e informática industrial realizar un análisis crítico que tipo de controlador es más eficiente al momento de aplicarlo en una planta.

Este módulo se desarrolla con el fin de crear un aprendizaje amigable con los usuarios en específico, de modo que permita a los estudiantes de la asignatura de teoría de control realizar estudios prácticos de manera presencial o de manera remota transformándose en un módulo híbrido para el estudio de los controladores entre PID y fuzzy, basándose en las leyes que rigen cada sistema y la evaluación de los compensadores obtenidos

## **1.2 Justificación**

El proyecto tiene como objetivo desarrollar un módulo de control retroalimentado para un sistema de aterrizaje y despegue vertical VTOL que permita al estudiante de ingeniería realizar ensayos de control. El módulo posee dos tipos de controles el clásico PID y el controlador FUZZY, los cuales serán manipulados mediante MATLAB y SIMULINK permitiendo el control de la planta. Observando un comportamiento de un sistema real en menor escala de un sistema como: los helicópteros, drones u otros tipos de aeronaves de despegue vertical.

El módulo VTOL permite controlar la altura deseada, la cual se observará su comportamiento mediante un análisis de gráficas que se obtendrá al momento de realizar las pruebas de los dos modelos de controladores, en la cual se realiza una comparación de las técnicas de control, seleccionando el controlador adecuado o más idóneo para el sistema.



## **1.3 Objetivos**

### **1.3.1 Objetivo General**

Desarrollar un sistema de control para la estabilidad de un modelo VTOL mediante lógica difusa, clásico PID y la visualización de parámetros desde un HMI y dashboard de nube, para el laboratorio de teoría de control de la Universidad Politécnica Salesiana

### **1.3.2 Objetivos Específicos**

Revisar bibliográficamente sobre los controladores PID y lógica difusa, y además de los diferentes parámetros mecánicos y eléctricos inversos en un sistema VTOL, mediante base de datos de artículos científicos y trabajos de grado similares propuestos a este estudio.

Construir un sistema VTOL para la simulación de vuelo mediante el diseño de un software especializado CAD.

Desarrollar un sistema de control en lazo cerrado para el sistema el VTOL, el cual permitirá la estabilización del vuelo, a través de un software especializado.

Crear una interfaz inalámbrica que permita la visualización de los índices de desempeño de la planta VTOL mediante dashboard en thingspeak.

Implementar el sistema de control en la planta VTOL para el análisis de estudio en la estabilidad de vuelo mediante parámetros de control desde un HMI y una plataforma en nube.

Comparar el sistema de control lógica difusa con el controlador clásico PID para la validación de su funcionamiento mediante índices de desempeño en un entorno controlado.

## **1.4 Metodología**

### **1.4.1 Metodología Inductiva**

Se aplicará en el diseño de planos con las normas que rige un modelo VTOL que determina la infraestructura que cumpla los parámetros adecuados del mismo.

### **1.4.2 Metodología Seccional y Longitudinal**

Realizar la estructura y análisis que debe tener el controlador difuso y controlador PID de un modelo VTOL, ya que será para el beneficio de los estudiantes de la Universidad Politécnica Salesiana para el laboratorio de control.

### **1.4.3 Metodología Analítica**

Se analizará y comparará los controladores con el fin de determinar un mejor desempeño del control dentro de sistema VTOL mediante la evolución de tiempos de estabilidad, tiempos de subida y su eficiencia energética.

### **1.4.4 Metodología Experimental**

Mediante la observación experimental se verificará el comportamiento de los controladores Difuso y PID dentro de un modelo VTOL propuesto. Para identificar el efecto que se produce en: el tiempo de subida, tiempo de establecimiento y su rendimiento energético, con ayuda del software Matlab y Simulink donde se realiza varias pruebas en tiempo real y simulado de la planta.

## CAPITULO 2

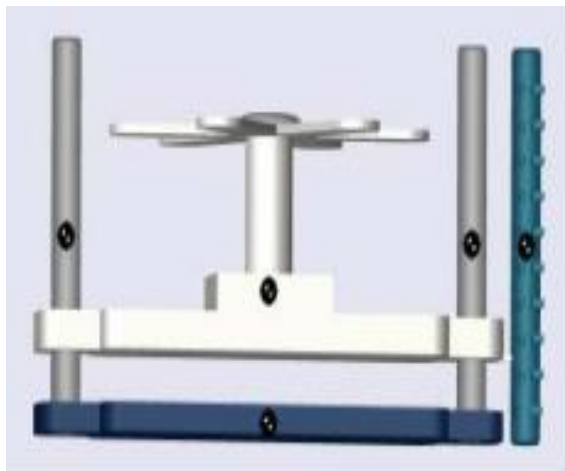
### FUNDAMENTACIÒN TEÒRICA

En el siguiente capitulo se describe los elementos que conforman el sistema VTOL, el software de desarrollo de los controladores PID y FUZZY, la interfaz gráfica y dashboard.

#### 2.1 Vertical take-off and landing (VTOL)

Los modelos VTOL (vertical take-off and landing) como se muestra en la figura.2.1-1 son sistemas o mecanismos que poseen gran versatilidad, ya que pueden desplazarse de manera vertical u horizontal, con la finalidad de conseguir la estabilidad para su funcionamiento eficaz, por lo tanto, el sistema VTOL debe tener precisión en respuesta y minimización de vértigo. (Marco Carpio-Alemán, 2016).

Figura.2.1-1 Sistema VTOL



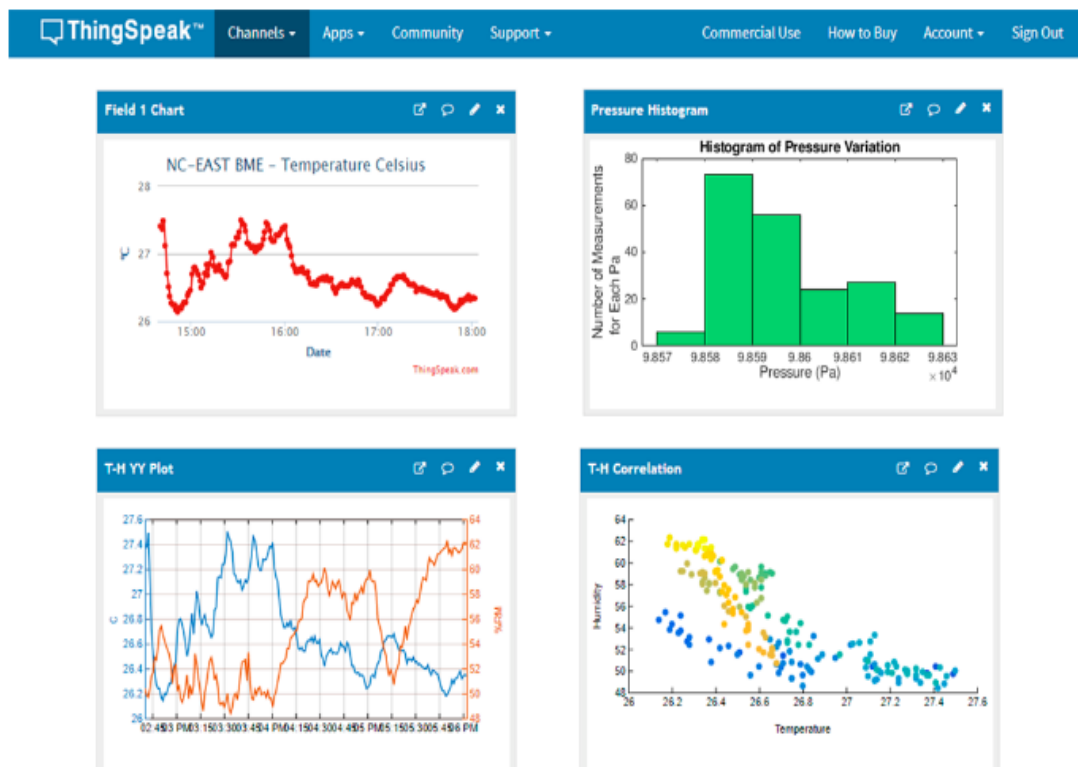
Autor: (Marco Carpio-Alemán, 2016).

El diseño de un sistema VTOL es importante para el estudio de sistemas de control, ya que permite idear su dinámica y comportamiento ante diferentes circunstancias, entradas, perturbaciones, ruido, cualquier fenómeno que pueda afectar de manera directa al normal funcionamiento de la planta (Castaño Rivillas Andres Camilo, 2018)

## 2.2 Dashboard

La dashboard es una herramienta de información que almacena datos en tiempo real, monitorea y muestra de manera visual los indicadores más importantes de desempeño para llevar a cabo el estado de una empresa, departamento o un proceso específico “ (Ortiz, 2021), la mayoría de los dashboards se encuentran en la nube y son fácilmente accesibles para varios miembros de un mismo equipo a la vez se encuentran en distintas partes del mundo, el objetivo de un dashboard es representar métricas de forma clara y concisa. De esta forma, cualquier persona podrá entender dichas gráficas. Como se puede observar en la Figura.2.2-1 (Vaquero, 2021)

Figura.2.2-1 Dashboard en la plataforma thingspeak



Autor: (Scharler, 2018)

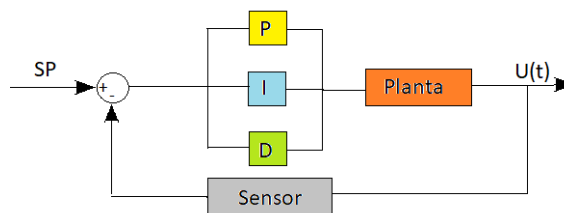
EL crecimiento de la comunidad de las ciudades inteligentes he interconectadas han inspirado mucho el área de la investigación sobre las dashboard lo que da como resultado una gran cantidad de datos provenientes de diversas fuentes, como sensores, señales eléctricas, dispositivos móviles, datos sociales. Esta gran cantidad de datos proviene de una gran variedad de aplicaciones que están sumergidas en diferentes áreas tales como: agricultura, infraestructura civil, mitigación y respuestas a desastres,

educación y aprendizajes, energía, calidad ambiental, salud y bienestar, medicina, servicios humanos, seguridad, servicios sociales, telecomunicaciones, transporte y movilidad, planificación urbana y rural. Comprender analizar y procesar los datos son importantes, a hora en día la visualización de datos en tiempo real también es importantes incluso a hora en la actualidad llegan hacer extremadamente útil he importante para la toma de decisiones. (Han, Nesi, Pantaleo, & Paoli, 2020)

### 2.3 Control PID

El controlador PID es el dispositivo mecánico o electrónico que controla un sistema en lazo cerrado, con el fin de mantener las condiciones deseadas al inicio del proceso. El cual está formado por tres variables: proporcional, integral y derivativa (Cusidó, 2006), este controlador es utilizado en la industria ya que es fácil de implementar, controlando la temperatura, velocidad, altitud, presión y flujo mediante un set point que es una señal establecida bajo ciertos parámetros deseados . Y puede ser utilizado en todo tipo de hardware. En la figura.2.3-1 se observa el proceso del controlador PID donde calcula la diferencia entre la variable real y la variable deseada (Garcia P. , 2013).

Figura.2.3-1 Estructura de un controlador PID con retroalimentación



Autor: Jefferson Zapata, Adrián Mendoza

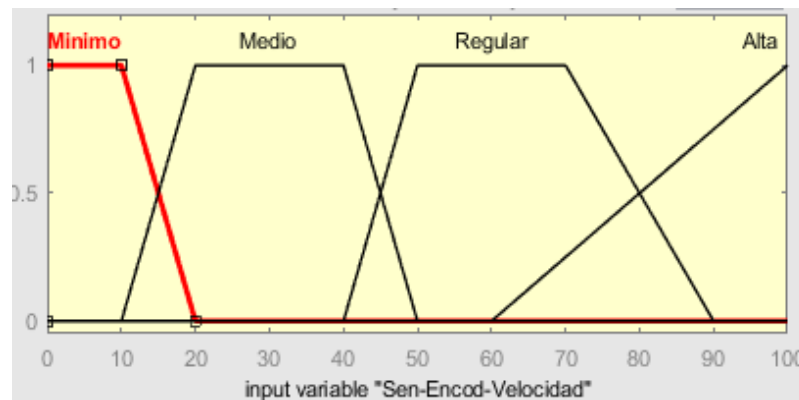
### 2.4 Lógica difusa

La lógica difusa fue inventada por Lofti Zadeh en 1965 que permite trabajar con información que no es exacta. Se refiere a los principios normativos y formales del razonamiento que la diferencia de la lógica clásica, la cual tomo más interés de investigación al momento de involucrarla con la creación de redes neuronales con su similitud a los sistemas fuzzy (Rosanna, 2005). La lógica difusa está basada entre 0

que representa hechos falsos y 1 que representa hechos ciertos, de igual importancia se procede a realizar los conjuntos como se muestra en la figura.2.4-1. Se procede a realizar la fuzzificación para proceder a crear las reglas lingüísticas y obtener una salida deseada, la idea de los conjuntos difusos es para verificar que la salida forme parte de un conjunto con un grado de pertenencia. (Morcillo, 2011)

La función primordial de la lógica difusa es emplearla en sistemas no lineales con alta complejidad, donde no existan modelos matemáticos precisos y no se tenga un conocimiento definido del sistema para solucionarlo (Morcillo, 2011)

Figura.2.4-1 Creación de los conjuntos difusos del controlador fuzzy



Autor: Jefferson Zapata, Adrián Mendoza

## 2.5 Interfaz hombre-maquina (HMI)

El HMI (Human Machine-Interface) por sus siglas en inglés, es un panel de control que permite al operador observar y verificar procesos de una planta en tiempo real. Utilizado más en procesos industriales de control y monitoreo de máquinas de producción (Delgado, 2019), este sistema permite al usuario monitorear de una manera más eficiente de acuerdo a las necesidades del usuario, permitiendo traducir las variables de procesos complejos en información relevante, intuitiva y procesable como niveles de depósito, tiempos, gestión de alarmas, estados de maquina y otros parámetros del proceso (Edimar, 2019)

## 2.6 Telemetría

La telemetría es un sistema que permite monitorear de forma remota cualquier tipo de variable ya sea física o eléctrica, mediante un sensor u otro dispositivo que emita señales analógicas o digitales. Los datos se enviarán cada cierto intervalo de tiempo, por lo cual la información guardada se procederá a enviar usando un protocolo y medio de comunicación que podrá ser accesible desde cualquier computador (Miño, 2009)

## 2.7 Motor brushless

La figura.2.7-1 se muestra el motor brushless , tomando en cuenta que no posee escobillas, lo que permite mantener el par en todo su abanico de velocidad, manteniendo su eficiencia máxima. Dentro del prototipo este motor en todo momento lo que permitirá elevarse al colocar la hélice. El motor posee las siguientes características.

- Revoluciones por minuto: 1000Kv
- Corriente máxima: 12 A
- Voltaje: 12V
- Potencia: 109 W

Figura.2.7-1 Motor Brushless



Autor: (Barragan, 2009)

## 2.8 Hélice

La hélice es una pieza que está conformado por aspas o palas como se muestra en la figura.2.8-1, la cual se encuentra montada de forma concéntrica en un eje, se encarga

de transformar la energía mecánica en empuje, lo que provocara cambios en la presión y velocidad del aire, las aspas tienen una forma curva, que provoca una diferencia de velocidad entre la cara superior e inferior, lo que conlleva a una diferencia de presión, provocando una fuerza propulsora la cual permitirá elevarse o efectuar un movimiento ya sea en el avión o barco. (villacastin, 2020)

Figura.2.8-1 Helices ft 1045



(Reyes, 2020)

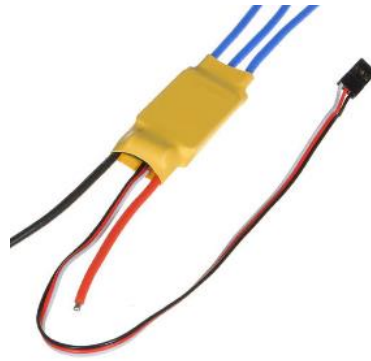
## **2.9 Electronic speed control ( ESC 30 A )**

El ESC también conocido como variador, es un dispositivo que recibe una señal del control de vuelo y es el encargado de alimentar adecuadamente al motor, hay que tener en cuenta que el ESC ver figura.2.9-1, es importante en este tipo de motores, porque determinar qué tan rápido y en qué dirección gira el motor, dispone de su propio firmware con su adecuada interface en la cual puede ser programa de esta manera, ayuda a un control en el motor para el sistema VTOL. A continuación, se detalla las siguientes características.

- Modelo: ESC 30 A
- Corriente Continua: 30 A
- Corriente Máxima
- Salida Regulador BEC: 2 A / 5 V



Figura.2.9-1 Driver ESC 30 A



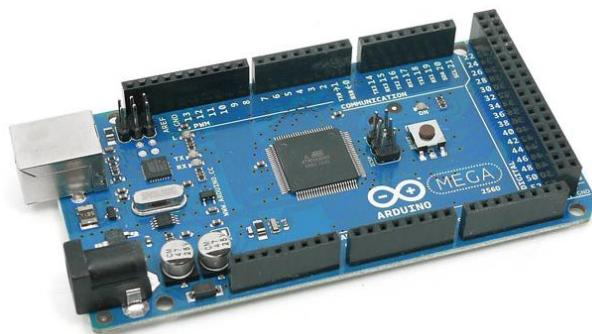
Autor: (SN, 2018)

## 2.10 Arduino mega 2560

Arduino mega 2560 es una placa desarrollada por ATmega2560 como se observa en la figura.2.10-1, con la configuración de un “sistema mínimo”, cabe destacar que posee una interfaz USB-Serie ya que permite reprogramar con el software Arduino IDE y un ordenador (Carmenate, 2021). Posee la capacidad de usar un power Jack como fuente de alimentación externa, dispone de 54 pines de entrada/salida digital lo que le hace la placa más grande, 15 de los pines pueden ser utilizados como salidas PWM. Sus características más importantes se detallan a continuación.

- 256 kB de Memoria flash.
- 8 kB de Memoria SRAM.
- 4 kB de Memoria EEPROM.
- 16 MHz Frecuencia de CPU Máxima.
- Voltaje de Operación Máximo 6.0v.

Figura.2.10-1 Arduino Mega 2560



Autor: (Velsos, 2018)

## 2.11 Tarjeta ESP8266 NodeMCU

La tarjeta ESP8266 NodeMCU como se muestra en la figura.2.11-1, es un módulo con sistema WIFI que sirve para realizar aplicaciones de IOT y es compatible con Arduino, el objetivo principal es dar acceso a una red. Sus características técnicas se detallan a continuación. (Hernández, 2018)

- Voltaje de Operación: 3V y 3.6V.
- Temperatura de Operación: -40°C y 125°C.
- Corriente de Operación: 80 mA.
- Soporta IPv4
- Protocolos TCP/UDP/HTTP/FTP.

Figura.2.11-1 Modulo ESP8266 NodeMCU



Autor: (Javier, 2019)

## 2.12 Sensor HC-SR04

El HC-SR04 es un sensor de distancia como se muestra en la figura.2.12-1, basado en un sistema de ultrasonido que permite medir trayectos de objetos en un rango de 2cm a 450 cm, el cual fue seleccionado por su mejor tiempo de respuesta y configuración para el desarrollo del proyecto. El sensor dispone las siguientes características.

- Voltaje operativo: 5 VDC.
- Frecuencia de trabajo: 40 KHz.
- Duración mínima del pulso de disparo: 10 uS.
- Tiempo mínimo de espera entre distancias: 20 mS.

Figura.2.12-1 Sensor HC-SR04



Fuente: (Arduino, 2014)

### 2.13 Sensor ACS712

El ACS712 es un sensor de corriente como se muestra en la figura.2.13-1, con efecto hall que detecta el campo magnético producido por un flujo de corriente, el cual permitirá medir la corriente que genera el sistema VTOL al momento de entrar en funcionamiento con el controlador (Garcia V. , Electronica Practica Aplicada, 2015).

Este módulo se caracteriza por.

- Rango Maximo de Corriente: +/- 30 A.
- Sensibilidad: 66mV/A.
- Resistencia interna: 1.2 mΩ.
- Voltaje de Trabajo: 5V.

Figura.2.13-1 Sensor ACS712



Fuente: (Garcia V., 2015).

### 2.14 Fuente de poder ac-dc 12v 30a 360w

Es una fuente conmutada de 12V a 30A, que transforma la corriente alterna en corriente continua en una o varias salidas como se muestra en la figura.2.14-1, está construida por una carcasa metálica y un ventilador para evitar sobrecalentamientos al momento de entrar en funcionamiento, acorde a los parámetros y características de los

elementos que conforman el sistema VTOL se adquirió esta fuente, tomando en cuenta las siguientes características que se detallan a continuación.

- Voltaje de Salida: 12V DC.
- Corriente de Salida: 30 A.
- Frecuencia: 50/60Hz.
- Potencia Máxima de salida: 360 W.
- Temperatura de Trabajo: -10 ~ +50° C.

Figura.2.14-1, Fuente de poder AC/DC



Fuente: (solectro, 2022)

### **2.15 Power Analyzer prova WM-02**

Es un analizador de potencia, que mide el flujo de potencia en un sistema eléctrico en el cual se refiere a la cantidad de transferencia eléctrica entre la fuente y su consumo, en el cual se ocupa energía por segundo (MideBien, 2019). En la figura.2.15-1 se observa cómo está conformado el analizador de potencia, con el cual se realiza pruebas del consumo de potencia que generan los controladores al cpu de nuestro sistema VTOL.

Figura.2.15-1 Power analyzer prova WM-02



Fuente: Adrian Mendoza, Jefferson Rolando

## CAPÍTULO 3

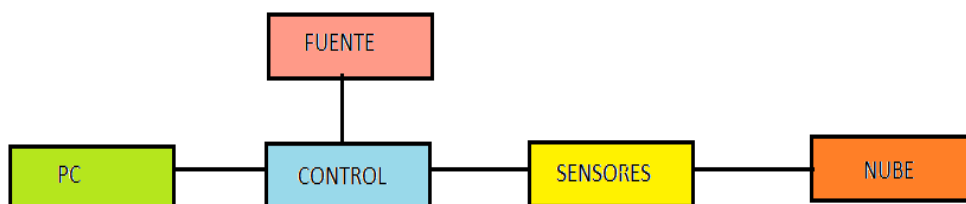
### DISEÑO E IMPLEMENTACIÓN

En este capítulo se presenta el diseño del hardware y software del sistema VTOL, como las configuraciones, diagramas de los controladores PID y FUZZY, para lo cual se utilizó el microcontrolador Arduino mega 2560 que realiza el control del sistema VTOL, del mismo modo se utilizó el módulo ESP8266 para la comunicación inalámbrica entre la planta y el servidor thingSpeak para el envío de datos y para la creación de sistema remoto.

#### 3.1 Descripción del hardware del sistema de aterrizaje y despegue vertical (VTOL)

Para el desarrollo del sistema de aterrizaje y despegue vertical (VTOL), se tomaron en cuenta diferentes puntos para su construcción, como la fuerza que ejerce el motor brushless al momento de entrar en funcionamiento, la estabilidad que debe tener para que no existan problemas en las guías, por lo cual se optó por construir una estructura robusta la parte exterior del módulo, después se procedió a colocar las guías en el interior teniendo en cuenta que son de material acero plata de 8mm, se escogió este material ya que tiene un acabado milimétrico lo que evita tener fuerza de rozamiento entre la guía y el rodamiento lineal, en la parte superior e inferior de las guías se colocaron resortes, para evitar que el motor de golpee y sufra daños. La estructura del módulo se puede apreciar en la figura.3.1-1

Figura.3.1-1 Diagrama de bloques sistema VTOL

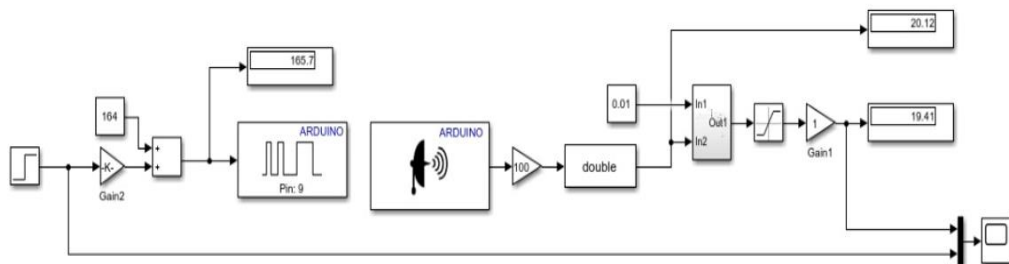


Autor: Adrian Mendoza, Jefferson Zapata

### 3.2 Diseño de la función de transferencia del motor brushless

Para la obtención de la función de transferencia, se hizo uso del motor brushless, mismo que fue sometido a varias pruebas en lazo abierto como se muestra en la figura.3.2.1, en la cual se obtuvo una matriz de datos de altura con respecto al tiempo de 5548x3.

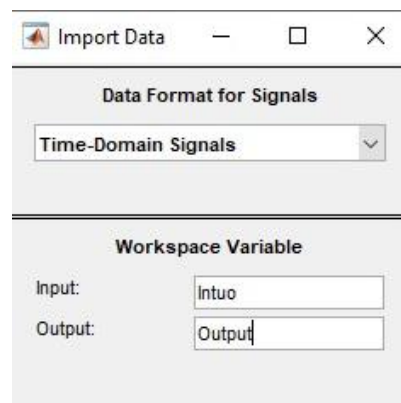
Figura.3.2-1 Sistema en lazo abierto



Autor: Jefferson Zapata, Adrian Mendoza

Con los datos recopilados en el proceso mencionado anteriormente, a la matriz obtenida se descompone en dos variables de entrada y salida, a la cual se procede a importar a la herramienta SystemIdentification de Matlab como se observa en la figura.3.2-2.

Figura.3.2-2 Importación de datos en entradas y salidas

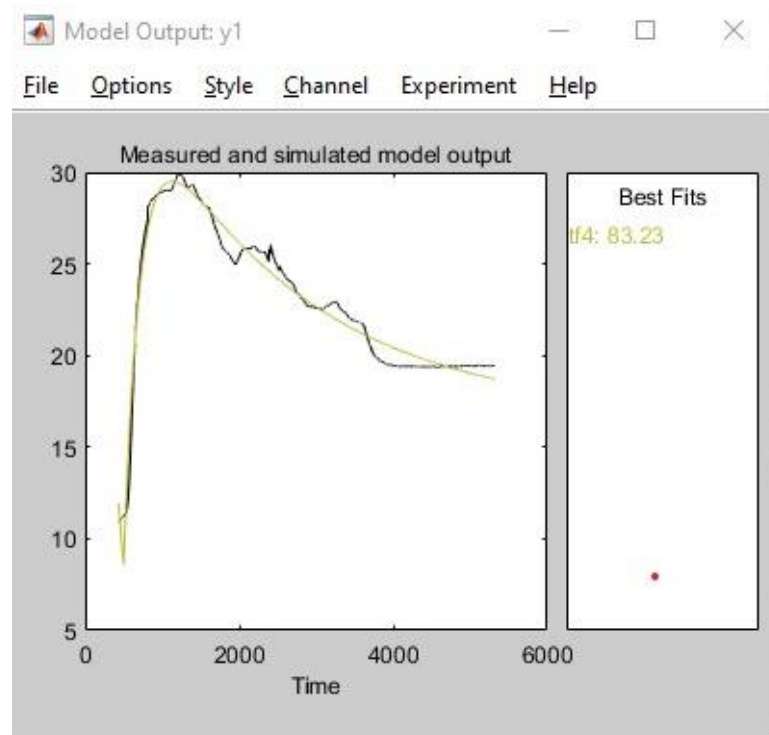


Autor: Jefferson Zapata, Adrian Mendoza

Una vez importados los datos, se realiza una estimación en la cual se obtiene un 84.23 % con respecto a la gráfica de datos, esto se puede observar en la figura.3.2-3, con

esta estimación entregada por el SystemIdentification se obtiene la función de transferencia adecuada para el sistema VTOL.

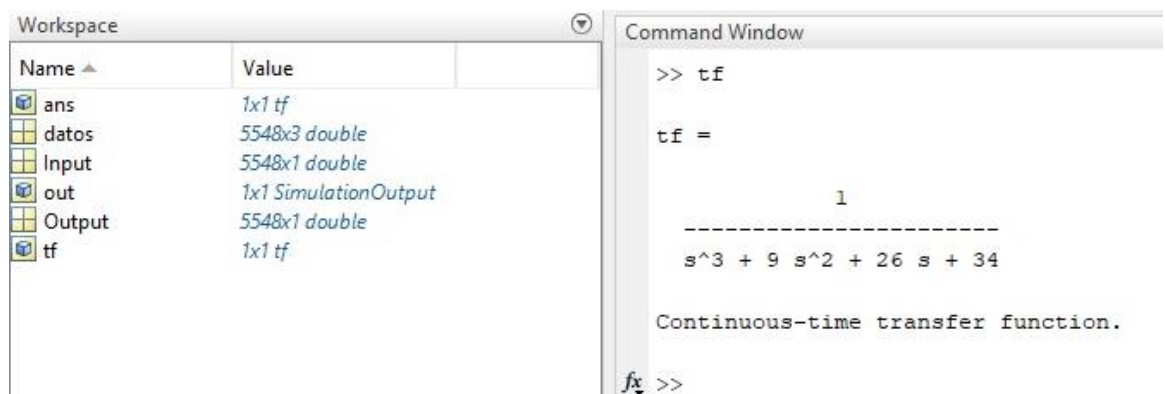
Figura.3.2-3 Estimación de datos del motor brushless



Autor: Jefferson Zapata, Adrian Mendoza

Posteriormente se procede a exportar los datos al workspace, en el cual se crea la variable tf, la cual en el Command Window se la ejecutara para visualizar la función de transferencia de la planta, esto se puede observar en la figura.3.2-4.

Figura.3.2-4 Ejecución de la variable en el command window



Autor: Jefferson Zapata, Adrian Mendoza

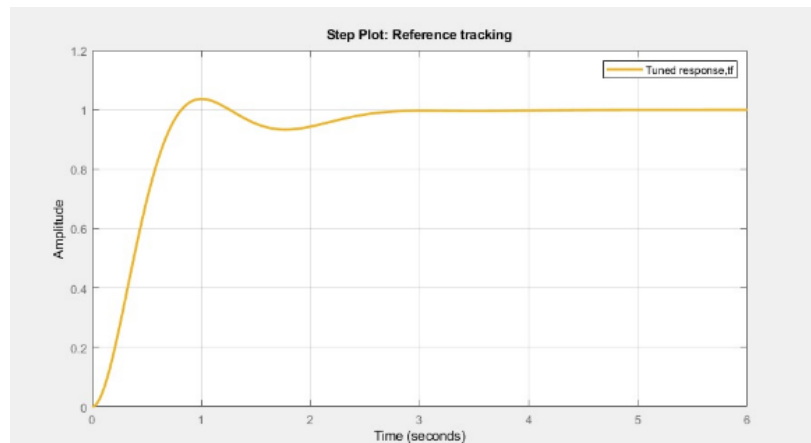


La ecuación (1) representa la función de transferencia de tercer orden obtenida para el sistema VTOL

$$\frac{1}{s^3 + 9s^2 + 26s + 34} \quad (1)$$

Con la función de transferencia obtenida, se aplica la herramienta pidTuner, la cual nos ayuda encontrar las ganancias  $K_p, K_i, K_d$ , ideales para el sistema, mediante la manipulación de la gráfica con respecto a la señal escalón, como se muestra en la figura 3.2-5, de la cual pudimos obtener los valores de las ganancias demostradas en la tabla 3.2-1

Figura.3.2-5 Grafica de la funcion de transferencia para la obtencion de las ganancias



Autor: Jefferson Zapata, Adrian Mendoza

Tabla.3.2-1 Valores de las ganancias del controlador PID

GANANCIAS	VALORES
<b>KP</b>	0.835
<b>KI</b>	0.48

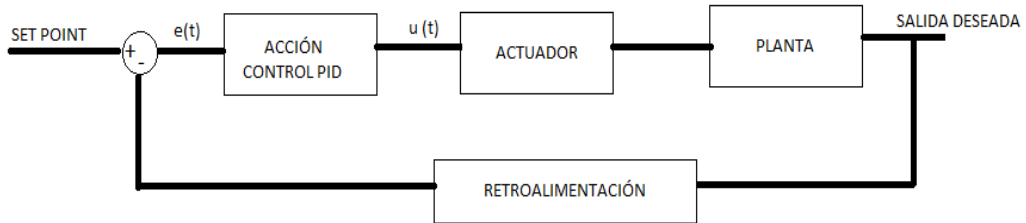
Autor: Jefferson Zapata, Adrian Mendoza

### 3.2.1 Diseño del controlador pid

El control PID es un sistema de control que a través de un lazo de retroalimentación permite regular la variable de un proceso. El controlador PID calcula el error entre la

variable real contra la variable deseada para obtener una respuesta más eficiente. (Molar, 2021) La estructura de un controlador PID se muestra en la figura.3.2.1-1.

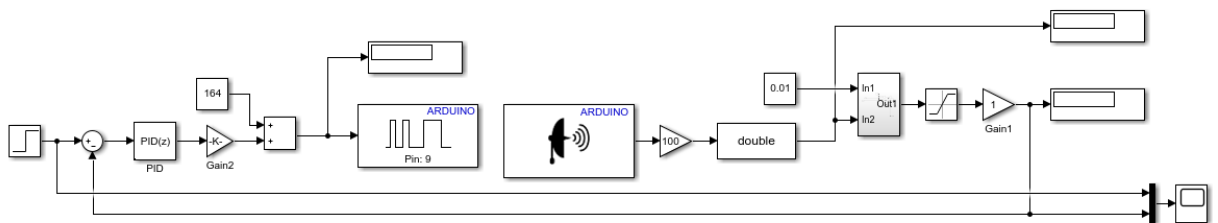
Figura.3.2.1-1 Estructura de un controlador PID



Autor: Jefferson Zapata, Adrian Mendoza

En el diseño del controlador PID para el módulo VTOL, partimos desde el sistema en lazo abierto, descrito en el literal 3.2, al cual se le implemento el bloque Discrete PID Controller en el cual se podrá modificar las ganancias,  $K_p$ ,  $K_i$ ,  $K_d$ , el mismo que realiza la acción de control a la planta del sistema VTOL, posteriormente se añadió la retroalimentación para realizar una comparación entre la variable real vs la variable deseada. En la figura.3.2.1-2 se observa la implementación del control PID en tiempo real.

Figura.3.2.1-2 Implementación del control PID



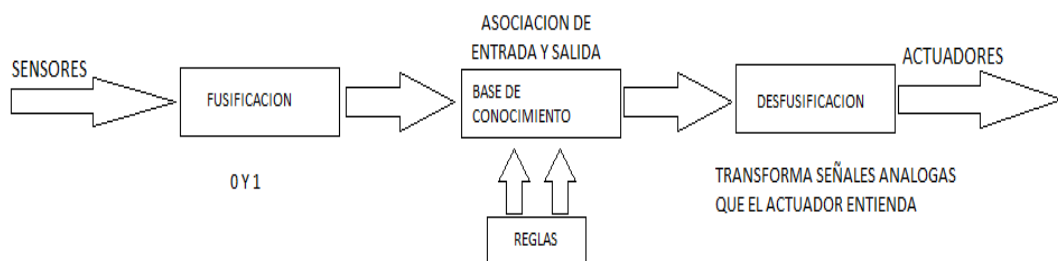
Autor: Jefferson Zapata, Adrián Mendoza

### 3.3 Diseño del controlador fuzzy

La lógica difusa permite a los sistemas verificar su funcionamiento, cuando tienen alto grado de imprecisión, los conjuntos difusos en su función de pertenencia adquieren valores en el rango de 0 a 1, el cual se les agrega un nombre a sus variables lingüísticas para diferenciarlos. (Luigi Constante Martinez, 2018) La estructura de un controlador

difuso se muestra en la figura.3.3-1, en primer lugar, tenemos los conjuntos que conforman la entrada que corresponde a la altura , posteriormente ingresa al bloque de fusificación donde se atribuyen grados de pertenencia a cada variable de la entrada , luego ingresa a la base de conocimiento donde se asocia los conjuntos de entrada con su conjunto de salida, dando paso a la creación las reglas lógicas las cuales son únicas e irrepetibles, en el siguiente bloque se tiene la defusificación que será la salida deseada para el controlador fuzzy.

Figura.3.3-1 Proceso de un controlador difuso



Autor: Jefferson Zapata, Adrián Mendoza

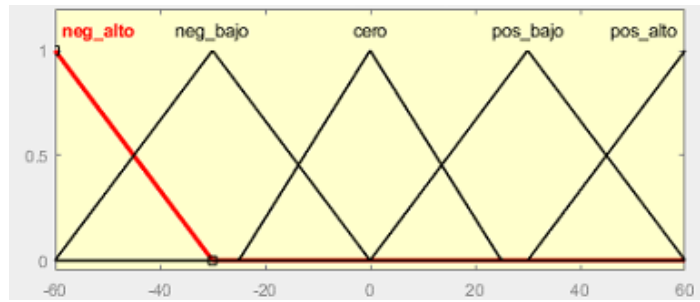
### 3.3.1 Diseño de los conjuntos difusos

En la elaboración de los conjuntos difusos del controlador, hay que tomar en cuenta diferentes aspectos como se detalla a continuación:

- A una entrada difusa siempre se dibuja empezando en 1 y se termina dibujando en 1.
- Los conjuntos difusos deben solaparse obligatoriamente.
- Se dibujan los conjuntos en números impares.

Las funciones de pertenencias utilizadas para el controlador difuso son de tipo triangular, donde se definió los nombres de las variables lingüísticas de los conjuntos de entrada que tienen un rango de -60 a 60 debido al máximo alcance que presta el sensor ultrasónico HC-SR04, se asignó los siguientes nombres: “neg\_alto”, “neg\_bajo”, “cero”, “pos\_bajo”, “pos\_alto”, como se muestra en la figura.3.3.1-1

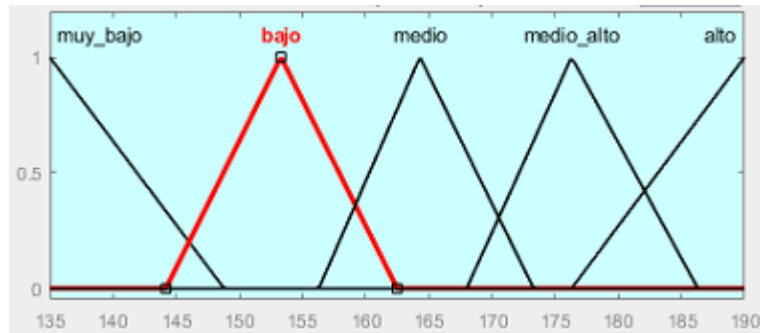
Figura.3.3.1-1 Conjuntos difusos de la entrada altura.



Autor: Jefferson Zapata, Adrián Mendoza

Asimismo, se asignó los nombres y variables lingüísticas para los conjuntos de la salida como se describe a continuación: “muy\_bajo”, “bajo”, “medio”, “medio\_alto”, “alto”, como se muestra en la figura.3.3.1-2.

Figura.3.3.1-2 Conjuntos difusos de la salida RPM.



Autor: Jefferson Zapata, Adrián Mendoza

Finalmente, se elabora los conjuntos del controlador difuso, creando las reglas o también conocida fans, donde se utiliza un algoritmo siso que es de forma unidimensional como se muestra en la tabla.3.3.1-1. Donde se observará que cada conjunto de entrada pertenece a un conjunto de salida.

Tabla.3.3.1-1 Reglas del controlador difuso

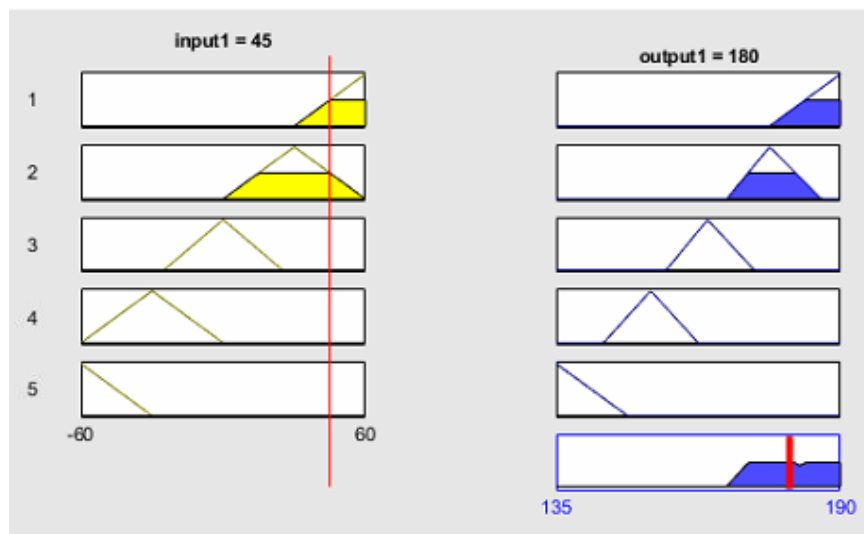
ENTRADA 1	SALIDA
Pos_alto	Alto
Pos_bajo	Medio_alto
cero	Medio
Neg_bajo	Bajo
Neg_alto	Muy_bajo

Autor: Jefferson Zapata, Adrian Mendoza

### 3.3.2 Fusificación en la entrada

Una vez definido los conjuntos de entrada, salida y reglas del controlador difuso, se realiza la fusificación con cualquier valor aleatorio en el rango de (-60 a 60), donde se calcula la función de membresía en cada uno de los conjuntos de entrada para verificar a que conjuntos pertenece y así obtener una salida única. En la figura.3.3.2-1 se observar la fusificación, donde la línea roja representa la variable y la sección de color amarillo representan los conjuntos que se intersecan en la entrada, para obtener la salida que esta de color azul con línea roja.

Figura.3.3.2-1. Fusificación de la entrada del controlador fuzzy

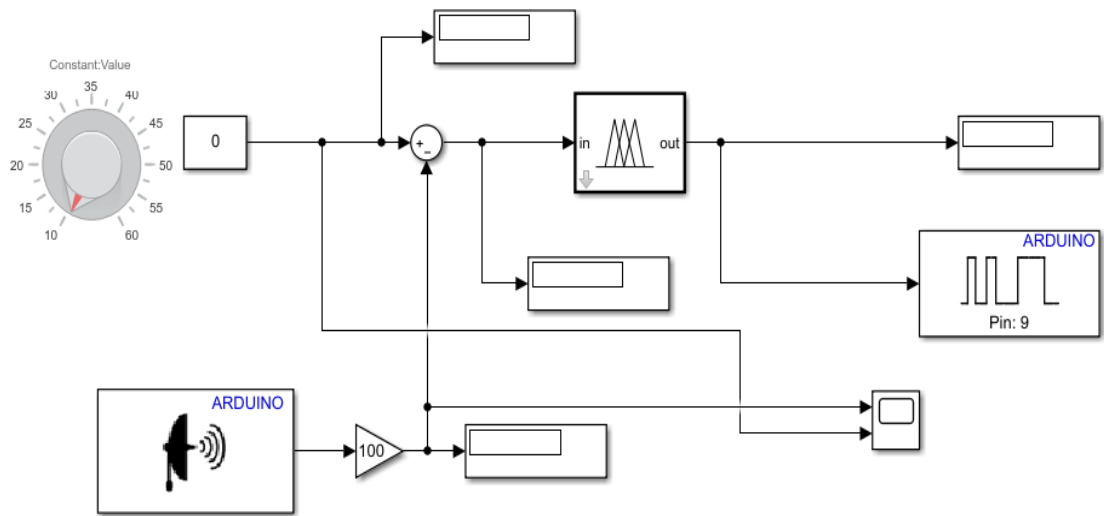


Autor: Jefferson Zapata, Adrián Mendoza

### 3.3.3 Controlador difuso en tiempo real

Luego, se incorpora el bloque de Fuzzy Logic Controller, en el resto de los bloques que conforman el controlador como son: Arduino, sensor ultrasónico, set point y displays como se muestra en la Figura.3.3.3-1, como primer paso se ingresa el set point, el mismo que realiza una comparación con el valor en tiempo real de la retroalimentación del sensor HC-SR04, a continuación, este valor ingresara al bloque del controlador fuzzy donde se realiza la fusificación, que luego pasa al PWM para obtener la salida deseada en el sistema VTOL.

Figura.3.3.3-1 Controlador difuso en tiempo real

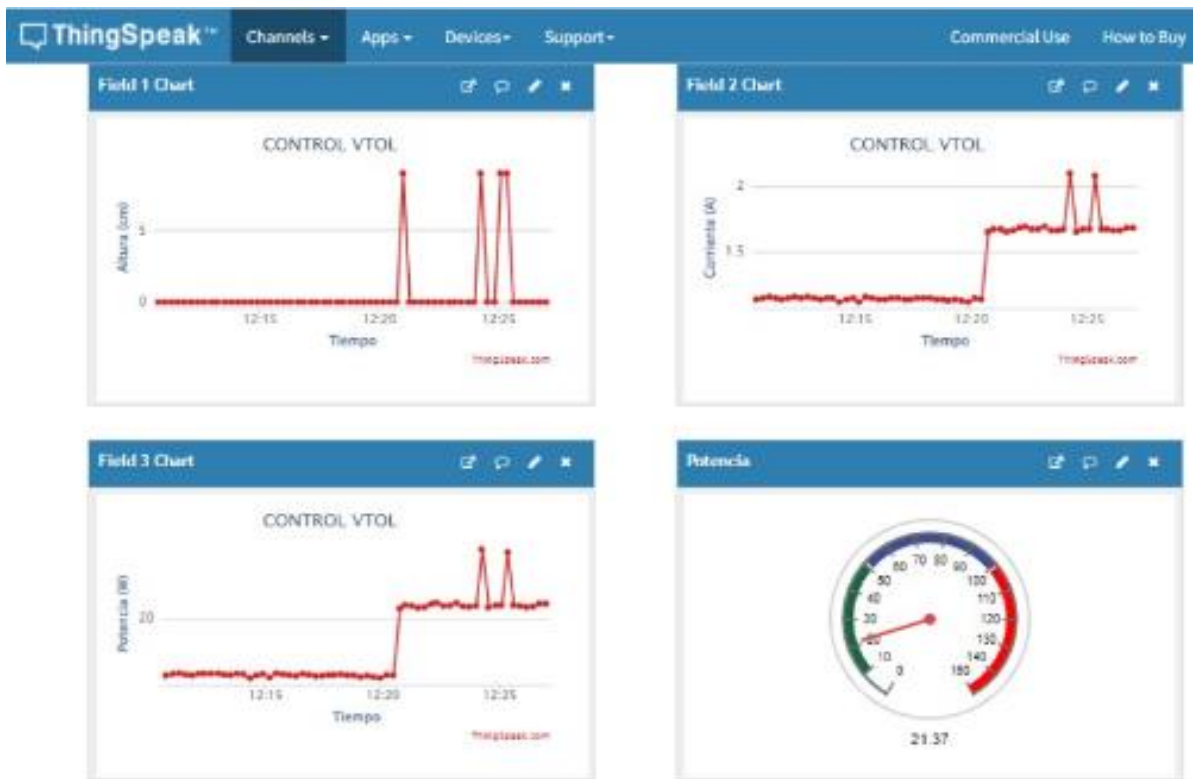


Autor: Jefferson Zapata, Adrian Mendoza

### 3.3.4 Visualización remota

Es importante mencionar que, para presentar los datos analíticos en el dashboard, se requiere establecer una conexión inalámbrica wifi de 2.4 GHz, basado en el protocolo IPV4 y compatible con Arduino, para lo cual se adaptó el módulo ESP8266MOD que maneja una frecuencia de operación de 80 MHz al prototipo del sistema VTOL. Con la finalidad de interactuar de forma remota con el funcionamiento de la planta. En la figura.3.3.4-1 que se presenta a continuación se indica los datos más relevantes del sistema VTOL al servidor de la nube (ThingSpeak), teniendo en cuenta que se actualizara cada 5 segundos, donde se observa los parámetros como: altura en el primer cuadrante, corriente en el segundo cuadrante y potencia en el tercer cuadrante, mismos que se obtiene en tiempo real de la planta.

Figura.3.3.4-1 Dashboard en ThingSpeak



Autor: Jefferson Zapata, Adrián Mendoza

### 3.3.5 Programación del módulo ESP8266MOD con el servidor ThingSpeak

Para la comunicación del módulo ESP8266MOD y el servidor, se creó una arquitectura cliente – servidor para la visualización de los datos en tiempo real, previamente se debe crear una cuenta en ThingSpeak para crear los canales que se utilizaran en el servidor. Donde se genera el código write API Key, que entrega la misma plataforma de ThingSpeak. Con estos datos se realiza la programación en IDE ARDUINO, en el anexo 1 se visualiza la programación para que exista comunicación entre el ESP8266MOD y el servidor ThingSpeak.

### 3.3.6 Interfaz Hombre-Maquina (HMI)

La necesidad de interactuar con el sistema VTOL y conocer su comportamiento en tiempo real, llevo a la creación de la interfaz HMI, el cual se encarga de supervisar, controlar e indicar históricos del proceso que lleva a cabo el módulo VTOL, en la

figura.3.3.6-1 se puede observar de cómo está conformado la interfaz, en la cual se observa el módulo VTOL en tiempo real y simulación y las fichas técnicas de los elementos que conformar la planta.

Figura.3.3.6-1 Interfaz gráfica del sistema VTOL



Autor: Jefferson Zapata, Adrian Mendoza

En el botón VTOL MÓDULO DE TIEMPO REAL al momento de accionar, se indica la identificación del driver del ESC y el motor Brushless, para luego proceder a utilizar el controlador PID y FUZZY en tiempo real, esto lo podemos observar en la figura.3.3.6-2.

Figura.3.3.6-2 Interfaz gráfica en tiempo real del sistema VTOL



Autor: Jefferson Zapata, Adrian Mendoza

Al accionar el botón VTOL MÓDULO DE SIMULACIÓN, se podrá realizar la simulación de los controladores PID y FUZZY, esto lo podemos apreciar en la



Figura.3.3.6-3 Interfaz gráfica en simulación del sistema VTOL



Autor: Jefferson Zapata, Adrian Mendoza

En la figura.3.3.6-4 se muestra que al momento de accionar el botón FICHAS TÉCNICAS, se observa una hoja técnica de cada elemento que conforma el módulo VTOL

Figura.3.3.6-4 Interfaz gráfica de la ficha técnica



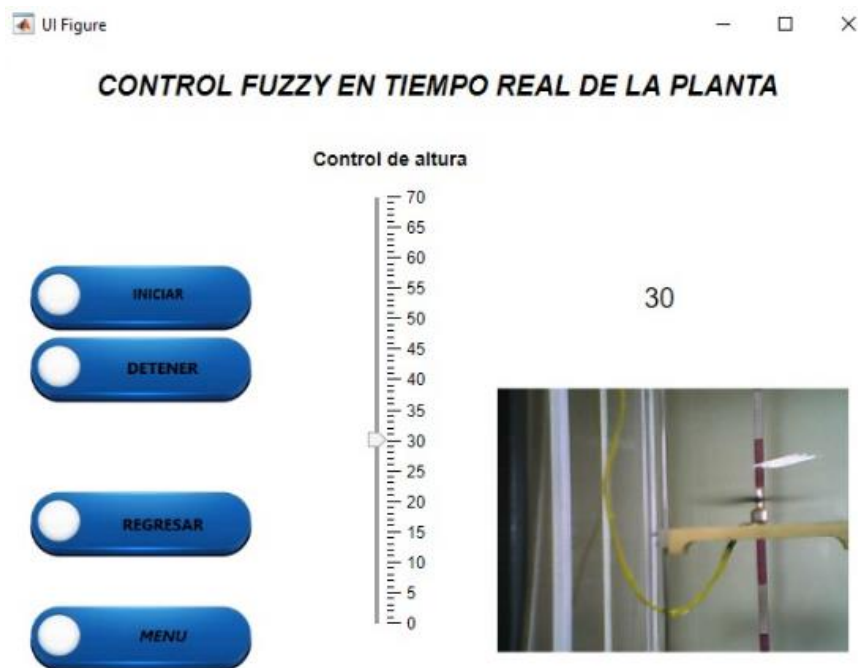
Autor: Jefferson Zapata, Adrian Mendoza

En la figura.3.3.6-5 se presenta la interfaz gráfica realizada de Matlab, la cual está conformada de una slider que permitirá establecer la altura del controlador FUZZY,

en la parte izquierda de la imagen se tiene los botones de: inicio, detener los cuales permitirán poner en funcionamiento la planta, en la parte inferior se tiene los botones de regresar y menú

Para ejemplificar el funcionamiento, se seleccionó el control de altura para 30[cm] para el control FUZZY. El rango de altura varia hasta 70[cm]. En la fotografía de la derecha se observa en tiempo que sistema ha subido la altura desea. Se ha incorporado una cámara web al módulo observar el comportamiento el sistema en tiempo real para el caso de acceder de forma remota.

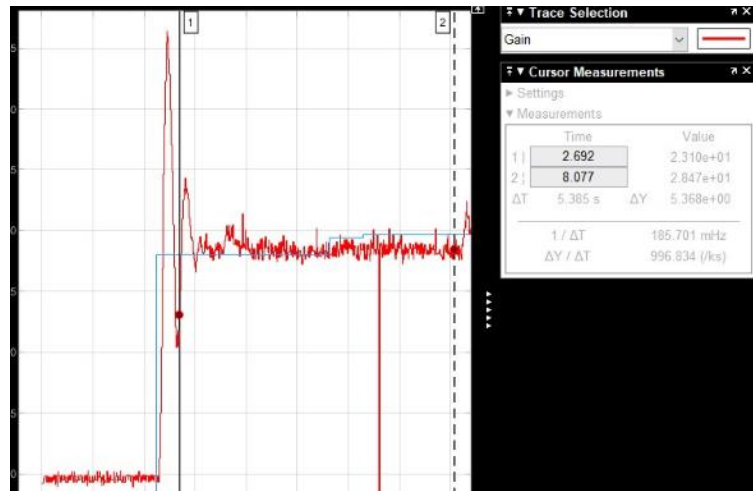
Figura.3.3.6-5 interfaz gráfica del controlador FUZZY tiempo real



Autor: Jefferson Zapata, Adrian Mendoza

En la figura.3.3.6-6 se aprecia mediante el osciloscopio la estabilidad que alcanza el controlador FUZZY, para alcanzar la altura de 30[cm], la línea de color azul es nuestro set point que está establecido a 30[cm], mientras tanto que la línea de color rojo es la señal que emite el sensor HC-SR04, en ella se aprecia que tiene un tiempo de subida de 0.16 segundos y un tiempo de establecimiento de 3.7 segundos,

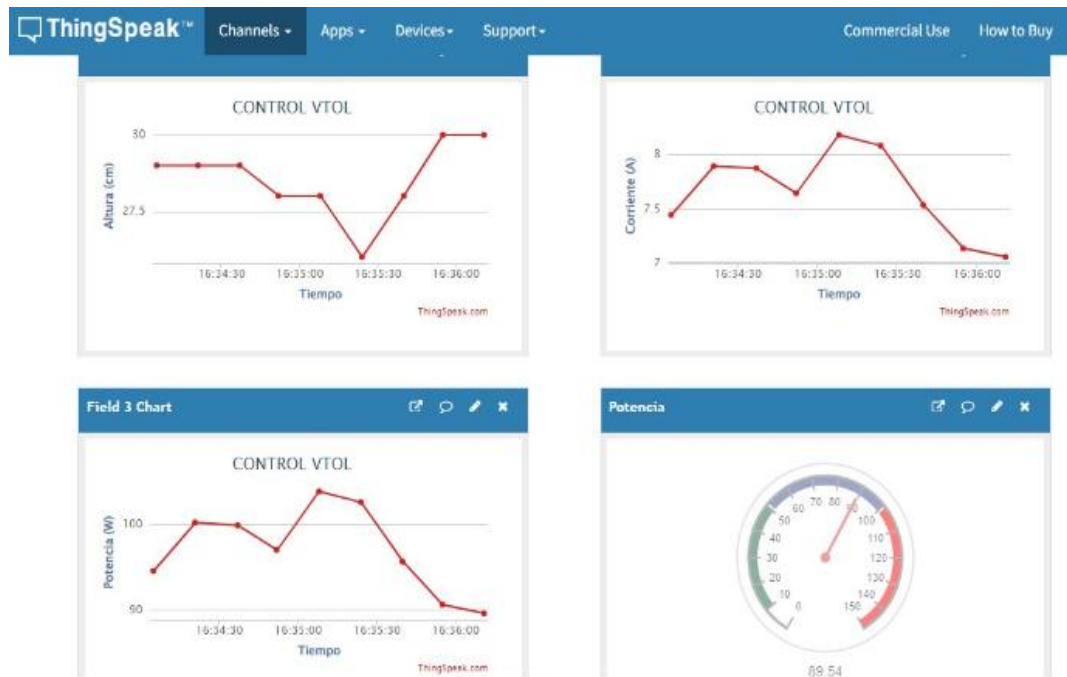
Figura.3.3.6-6 comportamiento del controlador FUZZY en un osciloscopio



Autor: Jefferson Zapata, Adrian Mendoza

En la figura.3.3.6-7 se observa los datos que son recopilados por el sensor HC-SR04 que se encarga de sensar la altura en tiempo real del controlador FUZZY, así mismo se tiene el sensor ACS712 que es el encargado de medir los datos de corriente, los cuales serán receptados por el módulo ESP8266MOD y se encargara de enviar los datos al servidor ThingSpeak en donde se visualiza los datos de: corriente, altura y potencia.

Figura.3.3.6-7 Datos del controlador FUZZY en thingspeak en tiempo real



Autor: Jefferson Zapata, Adrian Mendoza

### 3.3.7 Arduino

Para la selección del Arduino se consideró características detalladas en la tabla.3.3.7-1, obteniendo un mayor puntaje respecto a tarjetas similares, para lo cual el Arduino Mega es ideal para este proyecto.

Tabla.3.3.7-1 Parámetros de Arduino

	Arduino Mega	Arduino Uno	Arduino Nano
Microcontrolador	5	4	3
Memoria Flash	5	3	3
EEPROM	4	2	2
Salidas PWM	5	2	2
Puntuación	19	11	10

Autor: Jefferson Zapata, Adrian Mendoza

### 3.3.8 Modulo WIFI

En la selección del módulo WIFI se consideró características detalladas en la tabla.3.3.8-1, obteniendo un mayor puntaje respecto a módulos similares que se encuentran en el mercado, en el cual el módulo ESP8266 NodeMCU, es adecuado para enviar datos en tiempo real y exista comunicación con el servidor ThingSpeak.

Tabla.3.3.8-1 Parámetros del módulo WIFI

	ESP8266	ESP 8266 NODEMCU
Chip conversor USB-serial	3	4
CPU: Tensilica Xtensa LX3 (32 bit)	5	5
Frecuencia de Reloj: 80MHz/160MHz	3	5
Puerto serial UART	3	4
SDIO 2.0, SPI	5	5
Puntuación	19	23

Autor: Jefferson Zapata, Adrian Mendoza

### 3.3.9 Sensor de Corriente

En la selección del sensor de corriente se consideró características detalladas en la tabla.3.3.9-1, en la cual se obtendrá un mayor puntaje respecto a sensores similares que se encuentran en el mercado, por lo cual el sensor ACS712 es adecuado para la medición de corriente en este proyecto.

Tabla.3.3.9-1 Parámetros del módulo WIFI

	SCT-013-020	ACS712ELCTR-30A-T
Rango	2	5
Sensibilidad	3	4
Voltaje de salida	1	4
Puntuación	6	13

Autor: Jefferson Zapata, Adrian Mendoza

### 3.3.10 Sensor de Distancia

Para la selección del sensor de distancia se consideró características detalladas en la tabla.3.3.10-1, obteniendo un mayor puntaje respecto a sensores similares, para lo cual el sensor HC-SR04 es ideal para este proyecto.

Tabla.3.3.10-1 Parámetros del sensor de distancia

	SENSOR DE DISTANCIA INFRARROJO SHARP GP2Y0A21	SENSOR ULTRASONIDO HC-SR04
Rango	4	5
Precisión	2	5
Consumo de corriente	3	5
Puntuación	9	15

Autor: Jefferson Zapata, Adrian Mendoza

## CAPÍTULO 4

### ANÁLISIS DE RESULTADOS

En este capítulo se realiza el análisis del desempeño de cada uno de los controladores PID y FUZZY, posterior se compara el rendimiento con masa de (20gr y 40 gr) y sin masa y la ejecución de dichos controladores considerando variables como: tiempo de subida, corriente y consumo energético.

#### 4.1 Controlador PID y FUZZY sin masa

Con el propósito de identificar que controlador es más eficiente al momento de implementarlo se realiza una prueba al sistema VTOL sin masa en el cual se toma en cuenta los siguientes aspectos como: tiempo de subida, altura, corriente y potencia. En la tabla.4.1-1. Indica el valor ingresado de altura [cm], el tiempo de subida [S], el consumo de corriente [A] y potencia [W] de la planta, para el controlador PID y FUZZY, en el cual se realiza un análisis minucioso de los datos obtenidos a diferentes alturas.

Tabla.4.1-1 Tiempo de subida, establecimiento consumo de corriente y potencia de los controladores PID y FUZZY sin masa

ALTURA [cm]	Tiempo de subida [s]		Tiempo de establecimiento [s]		Corriente [A]		Potencia [W]	
	PID	FUZZY	PID	FUZZY	PID	FUZZY	PID	FUZZY
20	6,812	0,136	12,581	2,85	7,86	7,35	94,91	93,29
30	11,15	0,132	29,137	3,66	8,21	7,91	104,33	97,57
40	13,59	0,132	52,25	2,53	8,23	7,86	104,5	99,84

Autor: Jefferson Zapata, Adrian Mendoza

En la Tabla.4.1-1 se observa los datos de comportamiento del controlador PID para alcanzar una altura predeterminada versus el tiempo de subida. Para una altura de 20[cm] el tiempo de subida es de 6,812 segundos, en tanto para 40[cm] el tiempo de subida aumenta al doble alrededor de 13,590 segundos. El comportamiento del

controlador FUZZY versus el tiempo de subida para alcanzar una altura de 20 cm, 30 cm y 40cm se demora aproximadamente 0,13 segundos donde que el controlador FUZZY presenta una respuesta más rápida a la variable de entrada del sistema (Setpoint), se analiza el tiempo de establecimiento en el cual el controlador PID se establece en 12.581 segundos para alcanzar a una altura de 20[cm], mientras que para llegar a una altura de 40[cm] alcanza un tiempo de establecimiento de 52.25 segundos, en el cual se tiene un valor promedio de 31.322 segundos en el tiempo de establecimiento del controlador PID, se analiza el tiempo de establecimiento del controlador FUZZY, en el cual para alcanzar las alturas de 20cm, 30cm y 40cm su valor promedio bordea los 3 segundos para llegar a una estabilización adecuada en el controlador FUZZY.

Para determinar la eficiencia energética de los controladores, se realizó la prueba de consumo de corriente que generada cada controlador para alcanzar una altura de 20cm, 30cm y 40cm.

En la tabla.4.1-1, se muestra el consumo de corriente eléctrica que genera cada controlador para obtener una altura deseada y se mantenga estable. Se observa que el controlador PID y FUZZY tienen un valor promedio de 7.605[A] para alcanzar una altura de 20[cm], en tanto para alcanzar la altura de 30[cm] los controladores obtienen un valor promedio de 8.06[A],

En la tabla.4.1-1 se detalla el consumo de potencia que generan los controladores en el sistema VTOL, en la cual se observar que el controlador PID para alcanzar 20 [cm] genera una potencia de 94.91 [W], mientras que el controlador FUZZY para alcanzar las misma altura genera una potencia de 93.29 [W], en el cual no existe una diferencia significativa al inicio del proceso, sin embargo al aumentar la altura aumente la potencia para una altura de 40[cm] el controlador PID genera una potencia de 104.50[W] y el controlador FUZZY genera 99.84 [W], en el cual el valor promedio de potencia entre los controladores PID y FUZZY bordean los 99.07[W]

#### 4.2 Controlador PID Y FUZZY con masa = 20 gr

En la siguiente prueba que se realiza se considera una masa de 20[gr], en la tabla.4.2-1 se observa los datos del consumo de corriente de los controladores PID y FUZZY, también se encuentran los valores de altura que se ingresaron al sistema VTOL.

Tabla.4.2-1. Consumo de corriente con masa = 20gr

Altura [cm]	PID [A]	FUZZY [A]
20	8,850	7,420
30	8,420	7,540
40	8,700	7,340

Autor: Jefferson Zapata, Adrian Mendoza

En la tabla.4.2-1 se detalla el consumo de corriente que genera cada controlador, al momento de tomar una altura deseada, en la cual se observa que el controlador FUZZY, para alcanzar una altura de 20cm, 30cm, y 40 cm se obtiene un valor promedio de 7.43[A], de igual importancia se realiza para el controlador PID, el cual para alcanzar las mismas alturas obtiene un valor promedio de 8.65[A], por otra parte, se llega a determinar que el controlador FUZZY es el más eficiente al momento de implementarlo con una masa = 20[gr], ya que tiene un consumo de corriente menor al del controlador PID.

#### 4.3 Controlador PID Y FUZZY con masa = 40 gr

Para esta prueba se realiza con una masa = 40[gr] que se le acopla al módulo VTOL, a continuación, en la tabla.4.3-1, se detallan los valores obtenidos de: tiempo de subida, tiempo de establecimiento, corriente y potencia de los controladores PID y FUZZY, el cual permite hacer un análisis de su comportamiento con el peso aplicado.



Tabla.4.3-1. Consumo de potencia de los controladores PID y FUZZY con masa = 40gr

ALTURA (cm)	Tiempo de subida [s]		Tiempo de establecimiento [s]		Corriente [A]		Potencia [W]	
	PID	FUZZY	PID	FUZZY	PID	FUZZY	PID	FUZZY
20	15,52	0,201	46,78	2,64	8,67	7,53	110,1	95,62
30	12,23	0,167	28,69	1,95	9,12	8,38	110,4	106,41
40	10,25	0,161	38,15	3,7	9,21	8,4	110,5	108,54

Autor: Jefferson Zapata, Adrian Mendoza

La prueba que se realiza con una masa de 40[gr], el dato analizar es el consumo de corriente de los controladores que se muestra en la tabla.4.3-1, en el cual se aprecia que al momento de alcanzar la altura de 20[cm] en controlador PID genera un corriente de 8.670 [A], mientras que al realizar a 40[cm] su consumo de corriente es de 9.210[A], por el cual se obtiene un valor promedio de 8.98[A], mientras que el controlador FUZZY para alcanzar las mismas distancias de 20cm, 30cm, y 40cm, genera un valor promedio de 8.10[A], de esta manera se llega a determinar que el comportamiento del controlador FUZZY, es más optimo ya que tiene un consumo de corriente menor que el controlador PID, al momento que se implementa un peso al sistema VTOL

En la tabla.4.3-1 se detalla la cantidad de potencia que genera cada uno de los controladores, en ella se puede observar que la potencia consumida por el controlador PID para alcanzar las alturas de 20cm, 30cm, 40cm, tiene un valor promedio de 110[W] por lo cual en el controlador FUZZY la diferencia es notable, ya que empieza con un valor pequeño de 95,620 [W] a una altura de 20cm y para alcanzar una altura de 40[cm] genera un consumo de 108.540 [W], permitiendo obtener un valor promedio de 103.52[W], pero cabe destacar que el controlador FUZZY consume menos potencia en la planta.

En la tabla.4.3-1 se observa el tiempo de establecimiento que generan los controladores con una masa de 40[gr], en el cual el controlador PID para alcanzar una altura de 20[cm], genera un tiempo de establecimiento de 26.78 segundos, en tanto para alcanzar la altura de 40[cm], se obtiene un tiempo de establecimiento de 38.15 segundos, el cual se obtiene un valor promedio de 31.2 segundos, mientras que el controlador FUZZY para llegar a las alturas de 20cm, 30cm y 40 cm, su tiempo de establecimiento se obtiene un valor promedio de 3 segundos, llegando a concluir que el controlador FUZZY es más eficiente en su tiempo de establecimiento cuando se le incorpora una masa al sistema VTOL.

#### 4.5 Consumo de energía del procesador

Se implemento los controladores en el software Matlab que se instaló en una máquina Intel(R) core (TM) i5-3470 CPU de 3.20GHz. Mediante el POWER ANALYZER PROVA WM-02 se midió los consumos de energía para los controladores PID y FUZZY con la finalidad de determinar cuál controlador requiere más recursos de procesamiento para brindar una respuesta más rápida. En la tabla 4.5-1 se indica la toma de una muestra para una altura de 30 cm con una masa de 40 [gr], donde se obtuvo un consumo de energía del procesador de 17.596 [W] para el controlador PID y de 11.808 [W] para el controlador FUZZY. Esto es debido a que el controlador FUZZY presenta un mejor desempeño dando una respuesta más rápida para la estabilización del sistema por lo tanto requiere más recursos del procesador.

Tabla.4.5-1 Datos de potencia del CPU

ALTURA [cm]	PID [W]	FUZZY [W]
30	11.808	17.569

Autor: Jefferson Zapata, Adrian Mendoza

## CONCLUSIONES

En el presente proyecto se comparó los controles PID y FUZZY desarrollados en Matlab dentro de un sistema VTOL, con la finalidad de estudiar la estabilidad de vuelo mediante parámetros de control, para lo cual se obtuvo la función de transferencia y conjuntos difusos del motor brushless de forma experimental, el manejo del sistema VTOL se lo hace por medio de un HMI, donde se encuentran incorporados los controladores PID y FUZZY, el cual nos permite manipular: la altura deseada, las ganancias de controlador ( $K_p$ ,  $K_i$ ,  $K_d$ ) y los conjuntos difusos, la creación de un sistema remoto nos permite visualizar en tiempo real los índices de desempeño de: corriente, altura y potencia del módulo VTOL a través del servidor Thingspeak y de una cámara incorporada.

En la parte experimental se analizó el funcionamiento del sistema a diferentes alturas en el cual se observó que el controlador FUZZY presenta un tiempo de establecimiento promedio de 3.01[s], su tiempo de subida promedio de 0.1[s] en comparación al controlador PID que presenta un tiempo de establecimiento promedio de 31.31[s] y un tiempo de subida promedio 10.51[s] obteniendo como resultado experimental que el controlador FUZZY tiene una mejor respuesta.

Al realizar la construcción mecánica del sistema VTOL, se generó una fuerza de rozamiento entre la guía lineal y el rodamiento, la cual, para romper esta fricción, el motor Brushless genera un alto consumo de corriente de 12 A, que afecta al momento de sacar la función de transferencia de la planta.

Para el acceso remoto mediante el servidor Thingspeak se configuro y adapto al sistema VTOL el módulo ESP8266, el cual en su programación el servidor debe ser configurado la dirección IP de manera pública, ya que permitirá visualizar datos de: corriente, altura y potencia de forma remota, del mismo modo se crea el apiKey que genera el servidor para lograr una comunicación con el Arduino, lo que permite tener una conectividad entre la planta y el servidor.

## **RECOMENDACIONES**

Se recomienda que trabajos a futuro se pueda incluir una entrada más al controlador FUZZY y tomando en cuenta otros aspectos como: error de altura, revoluciones, los cuales permitirán dar mejor eficiencia a la planta.

Incorporar una cámara de humo, para visualizar el cono de aire que generan las hélices y así poder realizar un estudio de aerodinámica.

Seguir la guía de manejo del sistema VTOL para que no existan percances en el momento que se encuentre en funcionamiento.

## REFERENCIAS BIBLIOGRÁFICAS

- Arduino, T. (16 de 05 de 2014). GeeK Factory. Obtenido de GeeK Factory:  
<https://www.geekfactory.mx/tutoriales-arduino/hc-sr04-con-arduino-sensor-de-distancia-ultrasonico/>
- Barragan, M. G. (sn de sn de 2009). hetpro-store. Obtenido de hetpro-store:  
<https://hetpro-store.com/motor-brushless-a2212-13t-1000-kv/>
- Carmenate, J. G. (11 de 01 de 2021). Programarfacil.com. Obtenido de Programarfacil.com: <https://programarfacil.com/blog/arduino-blog/arduino-mega-2560/>
- Castaño Rivillas Andres Camilo, R. P. (2018). Modelamiento matematico y control de un sistema de despegue y aterrizaje certical a escala. Jovenes en la ciencia , 2480 - 2485.
- Cusidó, A. R. (2006). Control de procesos . Mexico: UPC,S.L.
- Delgado, R. (2019). Qué es un sistema HMI? Aplicaciones del mismo. Revistadigital INESEM, 3.
- Edimar, E. (15 de 03 de 2019). Electronica Edimar . Obtenido de Electronica Edimar : <https://edimar.com/hmi-tipos-y-funciones/>
- Fatima, K., Idrisi, A. H., & Mourad, A.-H. I. (2020). LabView based Conrol System Design for VTOL Aircraft System. Dubai, United Arab Emirates: IEEE.
- Garcia, P. (09 de 05 de 2013). Franklin Electric. Obtenido de Franklin Electric:  
<https://franklinlinkmx.wordpress.com/2013/09/05/que-es-el-control-pid/>
- Garcia, V. (28 de 09 de 2015). Electronica Practica Aplicada. Obtenido de Electronica Practica Aplicada: <https://www.diarioelectronicohoy.com/blog/sensor-de->

corriente-ac712

García, V. (28 de 11 de 2015). EPA. Obtenido de EPA:  
<https://www.diarioelectronico hoy.com/blog/sensor-de-corriente-ac712>

Han, Q., Nesi, P., Pantaleo, G., & Paoli, I. (2020). Smart City Dashboards: Design, Development, and Evaluation. IEEE.

Hernández, L. d. (11 de 02 de 2018). Programarfacil.com. Obtenido de Programarfacil.com: <https://programarfacil.com/podcast/esp8266-wifi-coste-arduino/>

Javier, L. (21 de 10 de 2019). hardzone. Obtenido de hardzone:  
<https://hardzone.es/reportajes/tema/esp8266-2n2222-arduino/>

Luigi Constante Martínez, J. C. (2018). Diseño e implementación de controladores PID y FUZZY para la planta de almacenamiento automático del sistema festo MPS500 en la UNIVERSIDAD POLITECNICA SALESIANA SEDE GUAYAQUIL . Guayaquil: UNIVERSIDAD POLITECNICA SALESIANA.

Marco Carpio-Alemán, W. O.-T. (2016). Design and Simulation of a Fuzzy Controller for Vertical take off and landing (VTOL) system. Cuenca-Ecuador: IEEE.

MideBien. (10 de 01 de 2019). MideBien. Obtenido de MideBien:  
<https://midebien.com/que-es-un-analizador-de-potencia/>

Miño, A. R. (2009). Implementación del protocolo modbus sobre una tarjeta de desarrollo para su uso sobre una red gsm con enfoque en telemetría . Santiago de Chile : Universidad de Chile .

Molar, G. I. (31 de 03 de 2021). Aleph. Obtenido de Aleph: <https://aleph.org.mx/que-es-un-control-pid-y-como-funciona>

Morcillo, C. G. (2011). Lógica Difusa una introducción práctica . Castilla: Universidad

de Castilla.

Ortiz, D. (20 de Octubre de 2021). Cyberclick. Obtenido de <https://www.cyberclick.es/numerical-blog/que-es-un-dashboard>

Piccirillo, A. C. (2019). German VTOL Fighter Program: A quest for Survivability in a theater nuclear environment. Virginia: American Institute of Aeronautics and Astronautics.

Reyes, J. P. (27 de 01 de 2020). ElectroPro. Obtenido de ElectroPro: [https://electropro.pe/index.php?route=product/product&product\\_id=414](https://electropro.pe/index.php?route=product/product&product_id=414)

Rosanna. (2005). Conceptos fundamentales de lógica Difusa. SN: SN.

SN. (08 de 08 de 2018). AV Electronics. Obtenido de AV Electronics: <https://avelectronics.cc/producto/brushless-esc-30a/>

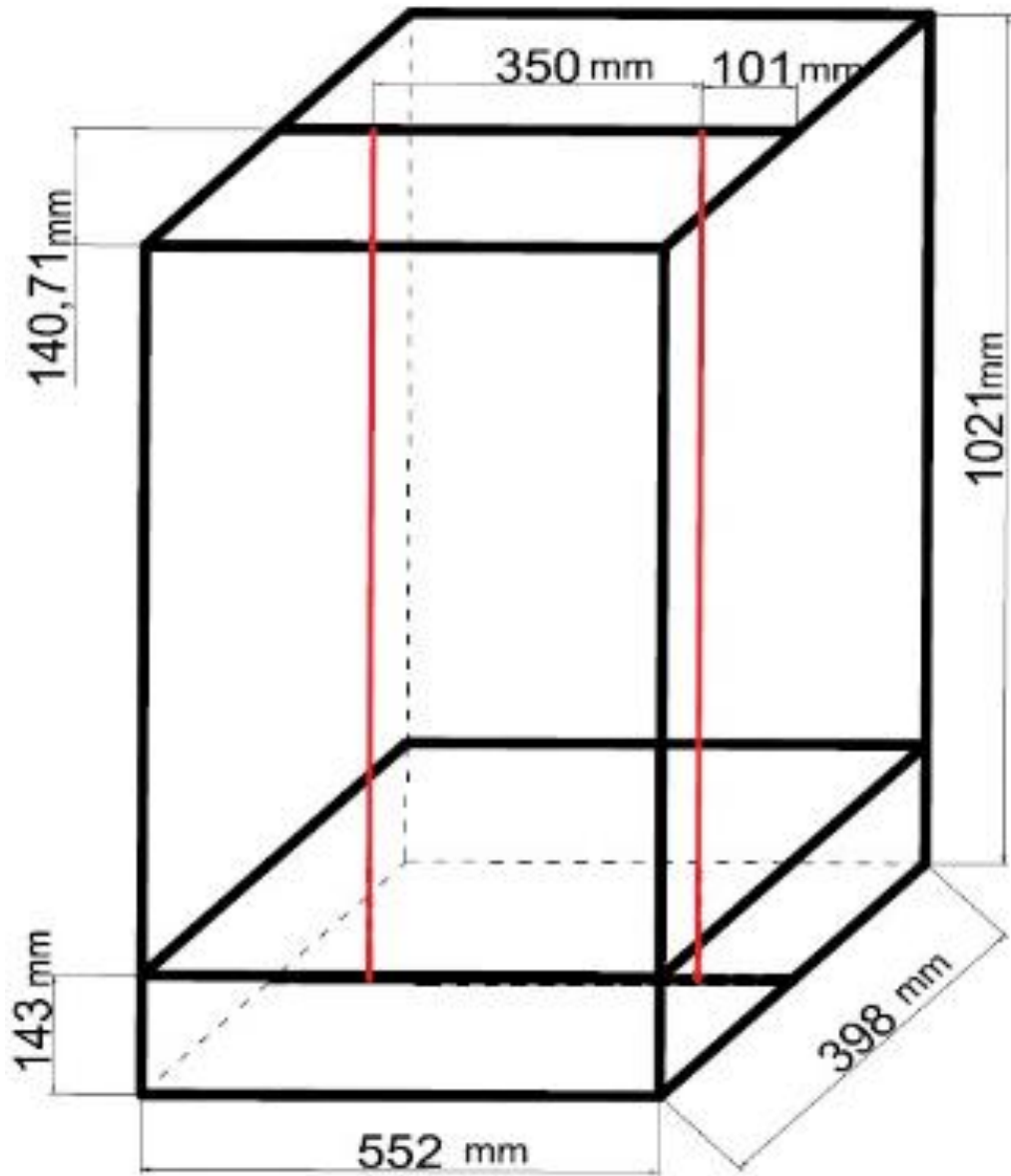
Vaquero, B. A. (12 de Abril de 2021). Que es Dashboard. Que es Dashboard . España.

Velsos, C. (19 de 06 de 2018). ETOOLS. Obtenido de ETOOLS: <https://www.electrontools.com/Home/WP/arduino-mega-2560-caracteristicas/>

villacastin, A. d. (20 de 04 de 2020). Escuela de aviadores. Obtenido de Escuela de aviadores: <https://escueladeaviadores.es/como-funciona-una-helice/>

## ANEXOS

**Anexo.1.** Plano de la estructura del sistema VTOL realizado en AutoCAD.





**Anexo.2.** Líneas de programación entre el servidor ThingSpeak y el módulo ESP8266MOD para envío y recepción de datos

```
#include <ESP8266WiFi.h>

String apiKey = "5GNMA5AXL90N6XZD"; // Enter your Write API key from
ThingSpeak

const char *ssid = "PROYECTO"; // replace with your wifi ssid and wpa2 key
const char *pass = "123456ZM";
const char* server = "api.thingspeak.com";
const int Trigger = D2;
const int Echo = D3;
float Sensibilidad=0.1;//sensibilidad para sensor de 20A
float Pot=0;
WiFiClient client;

void setup(){
  pinMode(D1,INPUT);
  Serial.begin(115200);
  pinMode(Trigger, OUTPUT); //pin como salida
  pinMode(Echo, INPUT); //pin como entrada
  digitalWrite(Trigger, LOW);//Inicializamos el pin con 0
  Serial.begin(115200);
  delay(10);
  Serial.println("Connecting to ");
  Serial.println(ssid);
  WiFi.begin(ssid, pass);
  while (WiFi.status() != WL_CONNECTED)

  {
    delay(500);
```

```

        Serial.print(".");
    }
    Serial.println("");
    Serial.println("WiFi connected");

}

void loop(){
    long t; //tiempo que demora en llegar el eco
    long d; //distancia en centimetros
    digitalWrite(Triple, HIGH);
    delayMicroseconds(10); //Enviamos un pulso de 10us
    digitalWrite(Triple, LOW);
    t = pulseIn(Echo, HIGH); //obtenemos el ancho del pulso
    d = t/59;
    float I=(get_corriente(200));
    Pot=12.7*I;

    if (client.connect(server,80)){// "184.106.153.149" or api.thingspeak.com

        String postStr = apiKey;
        postStr += "&field1=";
        postStr += String(d);
        postStr += "&field2=";
        postStr += String(I);
        postStr += "&field3=";
        postStr += String(Pot);
        postStr += "\r\n\r\n";

        client.print("POST /update HTTP/1.1\n");
        client.print("Host: api.thingspeak.com\n");
        client.print("Connection: close\n");
        client.print("X-THINGSPEAKAPIKEY: "+apiKey+"\n");
        client.print("Content-Type: application/x-www-form-urlencoded\n");
    }
}

```

```

client.print("Content-Length: ");
client.print(postStr.length());
client.print("\n\n");
client.print(postStr);

Serial.print("Distancia: ");
Serial.print(d);
Serial.print(" cm, Corriente: ");
Serial.print(I);
Serial.print("A. Potencia: ");
Serial.print(Pot);
Serial.println("W. Send to Thingspeak.");

}
client.stop();

Serial.println("Waiting...");

// thingspeak needs minimum 15 sec delay between updates
delay(1000);
}

float get_corriente(int n_muestras){
float voltajeSensor;
float corriente=0;
for(int i=0;i<n_muestras;i++){
    voltajeSensor = analogRead(A0) * (5.0 / 1024.0);///lectura del sensor
    corriente=corriente+(voltajeSensor -3.8)/Sensibilidad; //Ecuación para obtener la
corriente
}
corriente=corriente/n_muestras;
return(corriente);

```

**Anexo.3.** Sistema VTOL en funcionamiento en una altura de 30 cm

