



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
PROGRAMA DE MAESTRÍA Y DOCTORADO EN INGENIERÍA
INGENIERÍA ELÉCTRICA – INSTRUMENTACIÓN

SISTEMA INTELIGENTE EMBEBIDO APLICADO A UNA LANZADERA DE μ -UAV'S DE
ALA FIJA

TESIS
QUE PARA OPTAR POR EL GRADO DE:
MAESTRO EN INGENIERÍA

PRESENTA:
JULIO CÉSAR BALANZÁ RAMAGNOLI

TUTOR PRINCIPAL
GRACIELA VELASCO HERRERA, CCADET
COMITÉ TUTOR
JORGE PRADO MOLINA, INSTITUTO DE GEOGRAFÍA
NICOLÁS KEMPER VALVERDE, CCADET
ERNST KUSSUL, CCADET
GABRIEL EDUARDO SANDOVAL ROMERO, CCADET

CD. DE MÉXICO, JUNIO 2016



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

JURADO ASIGNADO:

Presidente: Dr. Jorge Prado Molina

Secretario: Dr. Nicolás C. Kémper Valverde

Vocal: Dra. Graciela Velasco Herrera

1^{er.} Suplente: Dr. Ernst Kussul

2^{d o.} Suplente: Dr. Gabriel Eduardo Sandoval Romero

Cd. Universitaria, Centro de Ciencias Aplicadas y Desarrollo Tecnológico,
Posgrado en Ingeniería Eléctrica, UNAM

TUTOR DE TESIS:

Dra. Graciela Velasco Herrera

FIRMA

*El presente trabajo de tesis fue desarrollado
en el Centro de Ciencias
Aplicadas y Desarrollo Tecnológico
de la Universidad Nacional Autónoma
de México, bajo la tutoría de la
Dra. Graciela Velasco Herrera
con beca CONACYT 625727*

DEDICATORIA

A Dios y a mi FAMILIA.

Así como a todas las personas que con cariño y respeto estuvieron conmigo y me ayudaron durante la realización de esta tesis, por decisión personal no pondré los nombres para evitar omitir alguno, infinitas gracias a todos ellos.

ÍNDICE

Resumen	9
CAPÍTULO 1 GENERALIDADES	10
1. Introducción.....	10
1.2. Justificación del Problema	10
1.3. Objetivo General	11
1.4. Objetivos Particulares	11
1.5. Metodología.....	12
1.6. Antecedentes	13
1.6.1. UAS.....	13
1.6.2. Categorías de los vehículos aéreos no tripulados	14
1.7. Sistemas de Despegue	16
1.7.1. Carrera.....	17
1.7.2. RATO	17
1.7.3. VTOL.....	18
1.7.4. Manual.	18
1.7.5. Lanzaderas.	19
1.8. Estado del Arte.....	22
1.8.1. Lanzaderas por tipo de diseño.....	22
1.8.2. Control difuso para motores a pasos	28
CAPÍTULO 2 CONCEPTOS Y FUNDAMENTOS TEÓRICOS	32
2.1. Sustentación de un UAV	32
2.2. Elevación.....	34

2.2.1. Tipos de perfil alar	37
2.3. Teoría de Lanzaderas	38
2.3.1. Lanzadera tipo elástica	38
2.3.1. Lanzadera tipo neumática	44
2.4. Sistemas de Orientación.....	47
2.4.1. Parametrización de la Orientación	47
2.4.2. Ecuaciones de movimiento.....	52
2.4. Lógica Difusa.....	59
2.5. Elementos de un conjunto difuso.....	59
2.5.1. Tipos de Funciones de Membresía (<i>FM</i>).....	59
2.6. Elementos de las funciones de membresía.	62
2.7. Sistema Difuso.	63
2.7.1. Reglas Lingüísticas.	64
2.7.2. Tipos de Sistemas Difusos.....	64
2.8. Algoritmo de inferencia difuso	69
2.9. Tipos de Sistemas Embebidos	73
2.9.1. Hardware en Sistemas Embebidos	74
CAPÍTULO 3. DISEÑO DEL SISTEMA DIFUSO PARA LA LANZADERA.....	76
3.1. Sistema difuso de ángulo de rotación y ángulo de ataque para la lanzadera	76
3.2. Sistema Embebido	76
3.2. Subsistemas Embebidos	78
3.2.1. Subsistema 1	78
3.2.1.1. Funcionamiento $e(t)$ y $de(t)$	79
3.2.2. Conjuntos difusos del subsistema 1	81

3.2.3. Reglas difusas.....	85
3.2.4. Defuzzificación del subsistema 1	87
3.3. Subsistema 2.....	87
3.3.1. Funciones de Membresía.....	88
3.3.1. Defuzzificación del Subsistema 2.....	90
3.3.1. Diseño de las reglas difusas.	91
CAPÍTULO 4. ELEMENTOS DEL SISTEMA.....	93
4.1. Tarjeta electrónica.....	93
4.2. Sensores	94
4.2.1. Sensor de dirección de viento	94
4.2.2. Sensor de peso	96
4.3. Filtro de Media Móvil.	96
4.4. Actuadores	98
4.5. Etapa de potencia y <i>Driver</i> Controlador.	98
CAPÍTULO 5. RESULTADOS	101
5.1. Comprobación de las Reglas Sistema 1.	101
5.2. Comprobación de las Reglas Sistema 2.	102
5.3. Simulación en <i>Simulink</i> ®.....	103
5.4. Resultados experimentales	108
5.5. Instrumentación Electrónica	114
CAPÍTULO 6. CONCLUSIONES	116
6.1. Conclusiones y discusión	116
6.2. Trabajo a futuro	117
6.3. Publicaciones y presentaciones en foros académicos	118

BIBLIOGRAFÍA.....	119
ANEXO A. CARACTERÍSTICAS DE LA TARJETA ELECTRÓNICA TIVA C LAUNCHPAD	126
ANEXO B. CARACTERÍSTICAS DEL SENSOR POTENCIÓMETRO.....	127
ANEXO C. CARACTERÍSTICAS DEL SENSOR DE PESO.	128
ANEXO D. TEORÍA MOTOR A PASOS	129
ANEXO E. CÓDIGO EN LENGUAJE C.....	136

Resumen

En éste trabajo se desarrolló un sistema inteligente embebido de orientación aplicado a una lanzadera de μ -UAV's. Se realizó en tres etapas: 1) simulación de dos subsistemas de rotación sobre la vertical y ángulo de ataque, 2) instrumentación de subsistemas embebidos y 3) comparación entre las simulaciones y pruebas experimentales en laboratorio. Se utilizó el método de inferencia de Sugeno, como datos de entrada fueron la dirección del viento y peso del UAV (de 0.5 a 2.0 kg) y el de salida: ángulo de rotación (de 0° a 360°) y ángulo de ataque (de 0° a 7.5°). La comparación de los resultados experimentales con la simulación mostró para el subsistema 1 un error máximo 0.5% y para el subsistema 2 un error máximo 4%.

CAPÍTULO 1 GENERALIDADES

1. Introducción

El uso de μ -UAV (micro-Vehículo Aéreo no Tripulado) hoy en día ha tomado gran relevancia en múltiples aplicaciones e investigaciones relacionados con el monitoreo aéreo, agricultura, y algunas otras más. Estos vehículos toman elementos de áreas como la aero-espacial, inteligencia artificial, sistemas de comunicación, sistemas de orientación satelital: GPS (Global Positioning System, por sus siglas en inglés), GLONASS (Global'naya Navigatsionnaya Sputnikovaya Sistema, por sus siglas en ruso), Galileo (Sistema de Posicionamiento Global Europe), sistemas embebidos y transporte; estos son considerados actualmente una tecnología de punta con múltiples aplicaciones que van desde las civiles hasta las militares. Una parte de su operación que es de gran importancia en éste tipo de vehículos son las técnicas de despegue, donde existen áreas de oportunidad para innovar, específicamente para los tipo μ (vehículo no mayor a 2 *kg*); éstos actualmente utilizan técnicas y métodos manuales o semi-automáticos que permiten la presencia de errores al momento del despegue ocasionando la posible pérdida parcial o total del vehículo; por lo que es necesario el desarrollo de un sistema de orientación, que permita un despegue en diferentes condiciones ambientales aleatorias y el éxito de la tarea a realizar.

1.2. Justificación del Problema

En éste trabajo se desarrolla un sistema inteligente embebido para la orientación de una lanzadera de μ -UAV de ala fija; utilizando inteligencia artificial, particularmente la lógica difusa y por primera vez el método de inferencia de Sugeno, como experto en la toma de decisiones para ejecutar la primera etapa de vuelo en condiciones reales.

1.3. Objetivo General

El objetivo principal de este trabajo es el desarrollo de un sistema de orientación difuso embebido basado en el método de inferencia de Sugeno, que se propone para ser integrado por primera vez como un sistema inteligente a una lanzadera de micro-UAV de ala fija, para accionar dos motores a pasos que se posicionen en la dirección opuesta al viento y en función del peso, respectivamente.

1.4. Objetivos Particulares

- Desarrollo de algoritmos utilizando el método de inferencia de Sugeno para el ángulo de rotación y ángulo de ataque de dos motores a pasos
- Implementación del sistema difuso embebido dentro de una tarjeta electrónica.
- Adquisición y procesamiento de la señal de un sensor que proporcione la dirección del viento
- Adquisición y procesamiento de la señal de un sensor que proporcione el peso de los UAV's.
- Desarrollo e implementación de la electrónica necesaria para la adquisición de las señales de los sensores.

1.5. Metodología

Éste trabajo de tesis se desarrolló un sistema difuso embebido utilizando el método de inferencia de Sugeno para realizar la orientación en un plano de coordenadas XYZ siendo Z la vertical. La metodología de éste trabajo se divide en:

- Acondicionamiento de las señales de entrada con un filtro de media móvil
- Desarrollo de algoritmos de lógica difusa utilizando el método de inferencia de Sugeno utilizando *Matlab*®
- Programación en *Dev C* de un algoritmo de lógica difusa utilizando el método de inferencia de Sugeno
- Implementación en C de un algoritmo de lógica difusa para ser embebido dentro de una tarjeta electrónica modelo *TIVA-C Launch Pad*
- Acondicionamiento de la señal de salida utilizando una etapa de potencia
- Integración de sensores de: dirección de viento, así como de peso para ser leído por la tarjeta electrónica y dos motores a pasos

1.6. Antecedentes

1.6.1. UAS

Por definición un UAS (Sistema Aéreo no Tripulado, por sus siglas en inglés) está conformado por siete subsistemas principales: 1) vehículo aéreo no tripulado (UAV), 2) carga útil, 3) estación de control, 4) lanzamiento y recuperación de la aeronave, 5) navegación, 6) comunicaciones y 7) transporte, (fig. 1) [1].

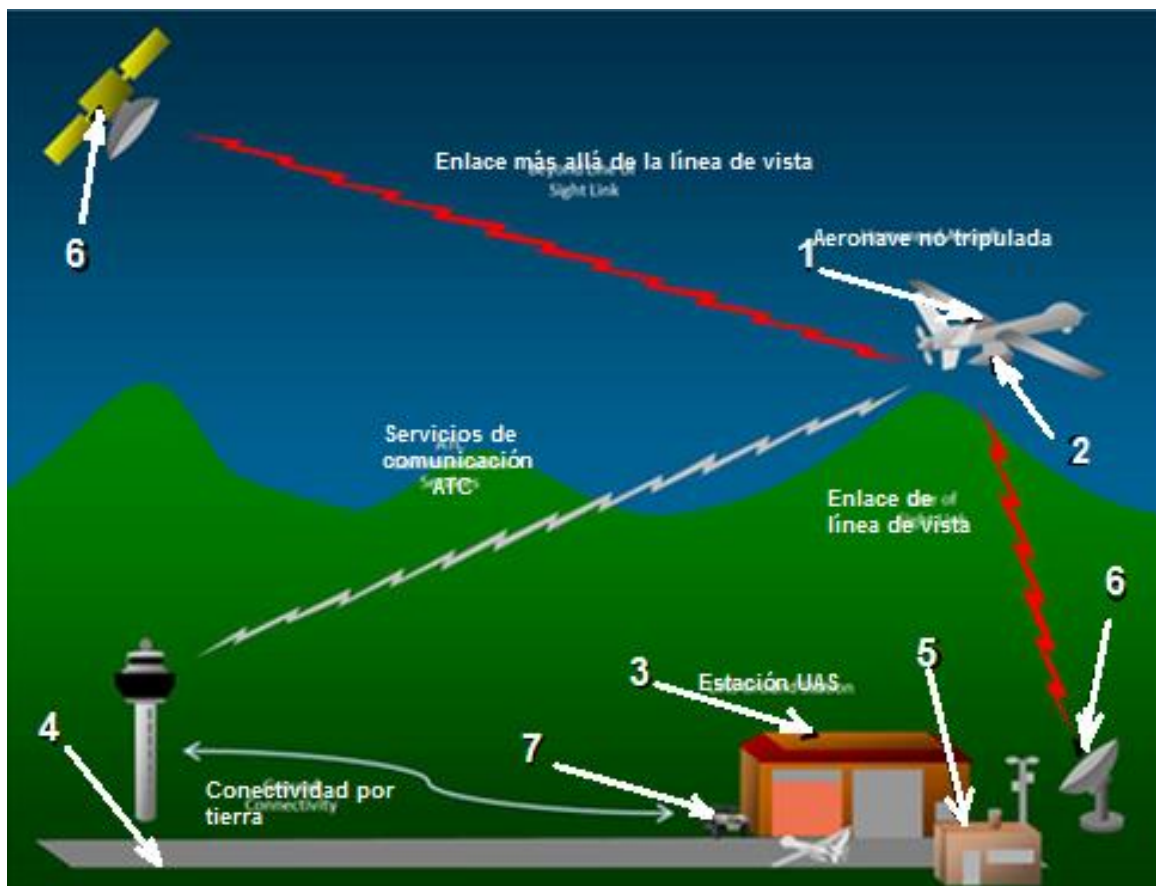


Figura 1. Sistema Aéreo no Tripulado (UAS) [2]

Algunas aplicaciones civiles de los UAS son: la fotografía aérea, agricultura, guardia costera, deforestación, entre otros. Dependiendo de la aplicación, existen categorías específicas para cada tipo de tarea a realizar.

1.6.2. Categorías de los vehículos aéreos no tripulados

El UAV de ala fija es un vehículo aéreo con alas rígidas con un perfil aerodinámico característico, (fig. 2).



Figura 2. UAV de ala fija tipo micro [3]

El UAV de ala fija se clasifica por su nivel de vuelo, capacidad en horas de operación, tamaño, peso, sistema de lanzamiento y recuperación, que son requeridas para seleccionar el dispositivo aéreo correcto para llevar a cabo y tener éxito en una misión en específico. Las categorías se muestran en la tabla 1 [1]:

Tabla 1. Categoría de UAV's [4]

Clase	Categoría	Altitud de operación	Ejemplos
III (> 600 kg)	Combate	20,000 m	Predator (U.S.)
	HALE	20,000 m	Global Hawk (U.S)
	MALE	14,000 m	Heron (U.S)
II (150 kg – 600 kg)	Táctico	3,000 m	Örnen (SWE)
I (<150 kg)	Pequeño (15 kg – 150 kg)	1,500 m	Scan eagle
	Mini (< 15 kg)	1,000 m	Falken (SWE)
	Micro	60 m	Black Widow

- **HALE** - *High Altitud Long Endurance*. Este UAV lleva misiones hasta los 15,000 m de altura y con una duración por más de 24 horas, por lo general lleva armamento y es operado por la fuerza aérea en una base terrena.
- **MALE** - *Medium Altitude Long Endurance*. Sus misiones van desde los 5,000 hasta los 15,000 m y por lo general por más de 24 horas de duración.
- **MRE** - *Multi Role Endurance*. El UAV más versátil, que conlleva características del MALE y HALE, capaz de realizar misiones medianas de 4 a 5 horas y/o largas de hasta más de 24 horas.
- **MR** - *Medium Range*. Mantiene las mismas características que un MALE pero con misiones de duración no mayor a 5 horas.
- **SR**- *Short Range*. Sus misiones van hasta alturas alrededor de los 5,000 metros, su característica principal es el tiempo corto de sus misiones, entre 4 y 5 horas máximo.
- **CR** - *Close Range*. Su operación no va más allá de los 100 m.
- **MINI** - Mini UAV. Su característica más importante es su peso, que es no mayor a los 20 kg
- **MICRO** - Micro UAV. El más pequeño de su categoría, su peso es de alrededor de 2 kg.

En la tabla 2 se muestra las categorías de UAV's y forma de lanzamiento.

Tabla 2. Tabla comparativa de UAV's en función de su tipo de lanzamiento [1]

Vehículos Aéreos no Tripulados											
Tácticos									Estratégicos		
Micro	Mini	CR	SR	MR	MRE	LADP	MALE	HALE	HALE	STRAT	
											Ala rotativa
											Ala fija
											Ultraligero
											Opcionalmente piloteado
Manual, Lanzadera, RATO (<i>Rocket assisted Take off</i>), Carrera									Carrera	Lanzadera	Lanzamiento
Deslizamiento, Red, Paracaídas, Vuelo vertical (VTOL), Carrera									Carrera		Recuperación

1.7. Sistemas de Despegue

El despegue es la fase inicial y esencial del vuelo con el cual se consigue el efecto aerodinámico de sustentación que es provocado por el flujo a una determinada velocidad del aire sobre las alas.

Para conseguir el despegue se consideran tres factores: lado tierra, lado aparato y lado aire.

Dentro de la categoría de UAV's de ala fija se encuentran clasificados según su tiempo de despegue o lanzamiento, los tipos de despegue se clasifican de la siguiente manera:

- Carrera
- RATO
- VTOL
- Manual
- Lanzadera

1.7.1. Carrera

Éste tipo de despegue se basa en el principio de carrera o empuje horizontal para acelerar a máxima potencia la aeronave para alcanzar durante la carrera la velocidad requerida para obtener sustentación, (fig. 3). La sustentación es un efecto vertical que siente un cuerpo que se desplaza a través de un fluido. A través del efecto aerodinámico se alcanza la sustentación suficiente para levantarla aeronave del suelo, para lo cual se requiere que las condiciones atmosféricas previas sean convenientes para conseguir un despegue seguro.

Por último, se mantiene la potencia máxima en los motores del aparato, se mueven los elevadores o timones de profundidad y parte de los *flaps* y se logra la separación de las ruedas del tren de aterrizaje del suelo, el avión inicia el vuelo.

El despegue se considerará finalizado una vez que el tren de aterrizaje haya quedado guardado [1].



Figura 3. UAV de ala fija despegando [4]

1.7.2. RATO

Abreviado como RATO (*Rocket Assisted Take-Off*, “despegue asistido por cohetes”), éste sistema de lanzamiento tiene su mayor ventaja al no requerir una pista de despegue. El funcionamiento básico del sistema es que el UAV está montado sobre un estante con un ángulo fijo. Sobre el UAV se encuentra un cohete adherido junto al sistema del motor. Al momento del despegue el cohete se enciende para que acelere lo suficiente y con ayuda

de su propio motor permita el despegue. Después del despegue el cohete es liberado para no agregar peso extra al resto de la misión del vehículo, (fig. 4) [1].



Figura 4. UAV con técnica de despegue RATO [5].

1.7.3. VTOL

VTOL (del inglés de *Vertical Take-Off and Landing*, “despegue y aterrizaje verticales”), es la capacidad de ciertos aviones para despegar de forma vertical, (fig. 5).



Figura 5. Mc Douglass AV-88 realizando despegue vertical desde un portaviones [6].

1.7.4. Manual.

Sistema de despegue utilizado por algunos UAV's menores a 2 kg en donde se sustituye la lanzadera para ser puestos en vuelo, (fig. 6).



Figura 6. Lanzamiento manual de un UAV de ala fija [7].

1.7.5. Lanzaderas.

Los sistemas de lanzamiento para lanzaderas empleados hoy en día, se pueden agrupar en cinco categorías: neumáticos, hidráulicas, elásticos, RATO (Rocket Assisted Take-off) y electromagnéticos [10].

1.7.5.1. Lanzaderas Elásticas

Es un dispositivo de lanzamiento tipo catapulta que hace uso de la energía que se almacena en resortes o cuerdas elásticas para acelerar un vehículo aéreo. Éste tipo de sistemas incluyen un riel de metal que se coloca en un ángulo de lanzamiento fijo, una cuerda elástica o resortes y una palanca para estirar el cable o elástico. Para el lanzamiento, el cable se estira para obtener la tensión máxima y se asegura con un perno, este último se libera para lanzar el UAV. Debido a la simplicidad en el diseño no requiere de muchos componentes eléctricos lo que lo hace relativamente ligero, (fig. 7) [1].



Figura 7. Lanzadera elástica modelo ASM300 de la empresa *air visión*® [5].

1.7.5.2. Lanzaderas Hidráulicas

Éste tipo de sistemas suelen también ser diseñados como lanzaderas. Los fundamentos del sistema: utiliza un cilindro de dos compartimentos, que están separados por un pistón que tiene un gas compresible en un lado y aceite en el otro. Para almacenar la energía necesaria para el lanzamiento, el aceite se bombea de modo que empuja el pistón y comprime el gas en el otro lado. El lado con el aceite está conectado a través de una válvula de apertura rápida a un motor hidráulico, el cual es el encargado de dar la energía para empujar el UAV y generar la aceleración necesaria para tener un despegue.

El sistema de lanzamiento hidráulico es mucho más complejo que los sistemas con cuerda elástica o con cohete y requiere de energía eléctrica para funcionar. Las ventajas son que puede producir velocidades más altas en comparación al de cuerda elástica, lo que permite un diseño para UAV's de mayor peso, (fig. 8) [1].



Figura 8. Lanzadera hidráulica de la empresa *UAVRL*® [5].

1.7.5.3. Lanzaderas Neumáticas

Los sistemas de lanzamiento neumáticos son muy parecidos a los hidráulicos. El almacenamiento de energía se lleva a cabo con un líquido compresible que mueve un pistón con aceite o algún tipo de gas como aire comprimido. Por lo general, el gas que se utiliza es el aire ambiental que se presuriza en tanques acumuladores. El lanzamiento se lleva a cabo dejando que el aire comprimido pase a través de una válvula controladora y acelere una cuna donde se encuentra asegurado el UAV. La fuerza con la que el lanzamiento se realiza, se puede aumentar o reducir mediante el ajuste de la presión. Algunas de las desventajas de esta técnica es que requiere de un compresor para presurizar el aire necesita tiempo para alimentar el sistema lo cual puede tardar en rellenar los tanques acumuladores, (fig. 9) [5].



Figura 9. Lanzadera hidráulica de la empresa *RPDEFENSE*® [6].

1.7.5.4. Lanzaderas Electromagnéticas

Éste tipo de lanzaderas utiliza el efecto de repulsión de los campos magnéticos para acelerar un vehículo aéreo no tripulado. La ventaja de ésta lanzadera es que permite tener un control de la velocidad y aceleración al variar la corriente que pasa por un conjunto de bobinas que se polarizan en función de lo que se desee, además de poder detener el

lanzamiento al cambiar la polaridad de las últimas bobinas. Su desventaja es la necesidad de requerir energía eléctrica para su funcionamiento, (fig. 10).



Figura 10. Lanzadera electromagnética modelo *Penguin W*® [7].

1.8. Estado del Arte

1.8.1. Lanzaderas por tipo de diseño

En el trabajo de Novakovic Z., [8] se presenta el diseño de una lanzadera para pequeños UAV (de 20 hasta 50 kg) que utiliza cuerdas elásticas tipo *Bungee*. El autor divide el análisis en secciones: 1) sistema de lanzamiento, 2) movimiento del subsistema sobre la carrera de despegue en el riel, 3) selección del sistema elástico para el estudio de la influencia de elevación, arrastre y de las fuerzas aerodinámicas.

El artículo de Gherman L. en [9], presenta el diseño de una lanzadera para UAV tipo mini (hasta 4 kg.), que utiliza como principio de lanzamiento, la Fuerza de Lorentz y el principio de repulsión de campos electromagnéticos; el autor desarrolla un modelo matemático y realiza simulaciones de movimiento para un diseño de lanzamiento de dos rieles.

El artículo de Silkov V. [10], presenta un modelo matemático utilizando ecuaciones cinemáticas para modelar el lanzamiento de un micro UAV de un peso máximo 1 kg, a través

de una rampa. El análisis obtenido puede ser utilizado para UAV's de hasta 149 kg con un tiempo de despegue de 2 segundos a una velocidad máxima de 95 km/hr.

El trabajo de Miller B. et. al. [11], presenta el diseño, construcción y pruebas de una lanzadera de un pequeño UAV no mayor a 25 kg, con una técnica de propulsión utilizando un sistema de resortes intercambiables. Ésta consta de un eje de disparo con tope final base o montura para el UAV y un cilindro para el enrollamiento del sistema de propulsión.

El artículo de Bernhard R. [12], presenta una simulación para lanzaderas utilizando fuerzas electromagnéticas para UAV táctico (hasta 300 kg) las condiciones iniciales: velocidad a 50 m/s, distancia de carrera de 14 m, y un ángulo de 30° de despegue. El artículo presenta tres secciones: 1) selección de un motor síncrono lineal, 2) cálculo del empuje requerido desde los parámetros de estudio básicos y 3) el diseño de los circuitos magnéticos y eléctricos. Se describe el modelado para obtener la fuerza necesaria de despegue, se muestran las ecuaciones para calcular el campo magnético máximo necesario en función del diámetro del cable, la fuerza y los polos necesarios para el accionamiento, el tamaño de la armadura, el empuje requerido y la carga térmica debida al cableado. Los resultados mostraron un 78% de eficiencia con respecto a la carga eléctrica y el peso del avión.

El artículo de Reck B. et. al. [13], muestra el modelado de una lanzadera tipo catapulta electromagnética, utilizando un motor lineal permanente tipo *brushless* que se mueve alrededor de la estructura y reduce la fricción entre el eje del lanzamiento y la montura para el UAV, de tal manera que evita el atascamiento con la línea de lanzamiento; el sistema de frenado se basa en un cableado sólido con núcleo de acero y un mecanismo de arrastre, que permiten obtener una velocidad de 22 m/s con 16 kwatts y un tiempo de 1.5 seg a una distancia máxima de despegue de 3 m.

El trabajo de Bumb N. en [14], presenta una metodología para el proceso de diseño de una lanzadera para UAV's de un peso máximo de 10 kg. El autor parte desde el concepto hasta

el diseño final, este proceso incluye identificar los requerimientos del cliente, la investigación sobre el estado del arte y el análisis de costos.

El artículo de Mirosław K. et. al. [15] presenta un prototipo de una lanzadera electromagnética para micro UAV's (menores a 2 kg). El diseño consta de 10 bobinas enrolladas separadas linealmente alrededor del marco o eje de lanzamiento en forma de solenoide, el cual es la sección de carrera en donde se encuentra montado el UAV, éste está fijo a un tubo diamagnético en cuyo interior se coloca un núcleo ferromagnético para aumentar el campo magnético. Se realizó una simulación para encontrar el número óptimo de vueltas de las bobinas y como condiciones iniciales se tomó una tensión máxima de 100 V, un impulso de 3 segundos para acelerar de 10, 20 y 30 m/s² y un peso máximo de 2 kg. La lanzadera se construyó y los resultados de laboratorio mostraron una resolución de 0.01 % de error respecto a la simulación, así como una aceleración máxima de 10 m/s². En [16] presenta el diseño CAD (Diseño Asistido por Computadora) de una lanzadera, implementando un sistema de propulsión electromagnética para micro UAV's (menores a 2 kg). Los elementos principales del modelo son: bobinas, sensores, plataforma donde se monta el UAV y el sistema de despegue. Los resultados de simulación se compararon con pruebas experimentales en donde existe una diferencia debido al coeficiente de fricción no considerado. En [17], [18] presenta un sistema de propulsión basado en el proceso de selección de solenoides para lanzaderas de micro UAV's. En este tipo de dispositivos se utiliza la conversión de energía magnética a eléctrica y a su vez en cinética. Este estudio lo realizó para la caracterización del desempeño y especificaciones de sus parámetros: espesor, largo, fuerza e inductancia intrínseca de esta.

El artículo de Emmanuelli A. et. al. [19], realiza un análisis con la metodología QFD (Despliegue de la Función de Calidad) para el diseño de lanzaderas; su propuesta de lanzadera consta de 4 partes importantes: riel de lanzamiento, soporte o montura para el UAV, pistón y base, cada uno de estas piezas se analizaron con elemento finito. La

propuesta final es un diseño para soportar el peso máximo de 25 kg, lo que permite ser cargada por una persona y ensamblada por un máximo de dos.

El artículo de Kaluzhikov I. et. al. [20], presenta una lanzadera para mini UAVs. Su diseño está compuesto por 11 elementos, consta principalmente de dos resortes que convierten la energía cinética en potencial para el despegue, el diseño consiste en ajustar el UAV a los resortes, en el piso está una estructura fija que proporciona el ángulo de despegue y mediante la acumulación de la energía cinética se libera un seguro que acelera el aparato. Se muestra un algoritmo para el cálculo de la aceleración utilizando un método numérico para el proceso de lanzamiento.

El artículo de Skurdal B. et. al. en [21] presenta el diseño de una lanzadera multi-misiones electromagnética para cargas de hasta 25 kg. El sistema está compuesto por un controlador y un sistema de alimentación a bases de pulsos. El objetivo de este sistema es el diseño de una lanzadera para múltiples tipos de misiones, así como eliminar el propelente en los sistemas de lanzamiento de cohetes y aumentar el rango de velocidad. Los resultados obtenidos son para una carga de 25 kg, con un ángulo de 88 grados y una velocidad de salida de 34 m/s.

El artículo de Liu X. et. al. en [22] presenta el modelado y simulación de un sistema neumático de lanzamiento para UAV's con un peso sin especificar. Se presenta el modelo matemático de todo el sistema neumático, así como los resultados simulados para diferentes presiones en el tanque y el tiempo de despegue alcanzado. La velocidad máxima alcanzado fue de 50 m/s.

El trabajo de Francis J. en [23] presenta una lanzadera para UAV's utilizando como modelo el UAV militar "*Eagle Eye*", el cual es un UAV militar que utilizan los botes de patrullaje costero del ejército de Australia para aumentar su gama de capacidad de vigilancia de objetivos específicos así como la vigilancia de barcos. El diseño es específico para un peso

máximo de 30 kg. El proyecto se diseñó en base a una metodología de siete pasos: planeación del proyecto, desarrollo de la calidad, diseño conceptual, diseño preliminar y detallado, limitaciones y resultados. Su aportación es diseñar una lanzadera para el UAV Eagle Eye con miras a su aplicación para vehículos aéreos del mismo peso o menores. Los resultados calculados fueron: fuerzas vs. distancia de disparo, ángulo de disparo vs. fuerza, finalmente terminando con el diseño conceptual en CAD.

La patente del trabajo [24] se presenta una lanzadera balística para pequeños UAV's (menores 10 kg). El tipo de propulsión está conformado por: sistema de aire comprimido, un sistema de empuje interno y una cámara de expansión encargada de mover y acelerar el UAV para el despegue. Las características del diseño alcanzan velocidades de aproximadamente 40 m/s y una aceleración lineal de 2000 m/s².

La patente de Miller W., et. al. [25] presentan una lanzadera neumática para pequeños UAV's. Éste utiliza sistema de acción tubular para el lanzamiento, se conecta por la parte trasera una manguera con gas a presión que mueve un carrito que se encuentra enganchado al UAV y al actuar genera la energía cinética necesaria para acelerar y realizar el lanzamiento.

La patente de Young E. et. al. [26] presenta el diseño de una lanzadera para micro UAV's. La razón principal de dicho desarrollo es evitar los lanzamientos a mano para tener un mejor control y de esta manera permitirle al operador tener las dos manos disponibles para utilizar el control remoto del UAV. El sistema de despegue consta de un riel unido a un resorte del tipo *Bungee* y una canastilla central en donde descansa el UAV el cual es accionado por la energía cinética del resorte. El sistema además tiene un sistema de poleas que permite subir o bajar el ángulo de despegue del UAV. Su principal invención es agregarle un pedal a distancia que permite accionar el sistema vía pie y no manual, volviéndolo más práctico.

La patente de la empresa *BAE Systems* [27] presenta una lanzadera para pequeños UAV's (menores a 10 kg). Dicho trabajo se diseñó para ser aplicado en el campo de batalla debido a su compacto y pequeño tamaño de no más de 6 m de largo y un diseño ergonómico que le permite ser cargado a lo más por una persona. El sistema de despegue está compuesto por un lanzador de mortero y una tapa que comprende una superficie de contacto adecuada para acoplarse con la cabeza de un proyectil de dicho mortero, dicha tapa está conectada con una cuerda elástica como accionamiento para el UAV. La ventaja de este diseño es permitir conectarse a una base y tapa de equipamiento ya disponible por el ejército.

La patente de Gromov Vladimir et. al. [28] presenta una lanzadera del tipo neumática que incluye un compresor eléctrico conectado a una válvula controladora para la apertura y cierre y de un pistón neumático, el cual está conectado a una carretilla donde se encuentra montado un UAV, la carretilla se desplaza por un eslabón hasta llegar a un tope con un sensor de fin de carrera que detiene su avance al desactivar la válvula del compresor. Cuenta con un cilindro que le permite elevar su posición para alcanzar un ángulo deseado

En los trabajos de Santamaría L. et. al. [29], [30] se desarrolló un algoritmo difuso tipo Mandami para el despegue de micro-UAV (hasta 2 kg); las variables consideradas son: la magnitud de viento, peso y envergadura del UAV. En la tesis de Santamaría L.[31] presenta además un primer diseño y algoritmos de control difuso para el lanzamiento en un plano del ángulo de despegue.

El trabajo Liyi A. et. al. [32] presenta el diseño de un sistema de control Fuzzy-PI para una lanzadera de carrera larga; para el sistema de control las variables de fuzzificación fueron la velocidad del error y los cambios con respecto a velocidad del error a la salida o defuzzificación. Los resultados obtenidos muestran que al comparar el control tradicional PI con el Fuzzy-PI se mantuvo una aproximación del 0% de sobrepaso y una estabilización del 50%.

1.8.2. Control difuso para motores a pasos

El artículo de Sarac V. et. al. en [33] muestra un esquema en *Simulink*® de Matlab® de un motor a pasos híbrido con dos bobinas para la simulación de sistemas de control tipo clásico o inteligente. Su sistema de modelado es un motor que permite ingresar características variables como, número de pasos (grados), pasos por ángulo (pasos/°), reluctancia (H), resistencia de las bobinas (Ohms), etc. A la salida del sistema se puede obtener el ángulo alcanzado (grados) así como la velocidad angular (rad/s), torque (N-m) y corriente (A).

El artículo de Grimbleby J. B. [34] presenta un algoritmo para controlar la posición de un motor a pasos a través de un lazo cerrado de control. Los motores a pasos son normalmente operados sin retroalimentación, lo que ocasiona que se pueda perder la sincronía al momento de moverse a un punto determinado, esto se puede prevenir mediante el uso de realimentación de la posición. El algoritmo funciona para operar con eficacia en el modo de bucle abierto, siempre y cuando se mantenga sincronizado. El algoritmo de control puede implementarse como cuatro máquinas de estado interconectados, un contador / descontador y un tacómetro.

El artículo de Betin F. et al [35], [36] presenta el diseño de un controlador por lógica difusa implementado en un microcontrolador de 16 bits para el control de velocidad de un motor a pasos y un sistema de pruebas para realizar variaciones de velocidad. Las funciones de membresía utilizadas son, el error de posición y cambio de velocidad, medido a partir de un sensor tacómetro, se utilizan funciones de membresía tipo triangular. Los resultados se compararon con un sistema en lazo abierto mostrando bajas oscilaciones que bajo lazo abierto llegaron hasta 20% por arriba del valor de control.

En el artículo de Barman J. K et al [37] desarrolla un controlador difuso tipo PD para la posición exacta de un motor a pasos. El controlador funciona con dos variables de entrada, error y el cambio en el error, las funciones de membresía para cada variable son 5 funciones triangulares, en el error el universo de discurso va de -60 a +60 y de idéntica manera el de

cambio en el error, la salida son cinco funciones triangulares distribuidas de -5 a 5 que representa el número de pasos a avanzar. El sistema de control se probó en *Simulink*® de Matlab con un motor a pasos híbrido de 1.8° por paso. Los resultados mostraron 1 milisegundo de respuesta con sobrepaso despreciable.

El artículo de Dominic A. et al [38] muestra la simulación de un control de posición difuso tipo PD y singletons a la salida para un motor a pasos con una función de transferencia dada. Las funciones de membresía para error y cambio en el error son tres funciones triangulares y dos trapezoidales que cierran el universo distribuido de -1 a 1. Las funciones de defuzzificación son 9 singletons distribuidos en un universo de discurso de -1 a 1. El controlador difuso fue simulado en *Simulink*® de Matlab utilizando un modelo de motor híbrido a 1.8° por paso. Los resultados muestran que el controlador responde en un tiempo mínimo a los cambios del error con un sobresalto menor al 1%.

El artículo de Roshan H. et al [39] muestra un sistema de control difuso para la posición de un orientador o seguidor solar para buscar el máximo gradiente de luz utilizando un motor a pasos y un microcontrolador de 12 bits. Las funciones a la entrada del sistema, son posición y gradiente de luz, a la salida se tienen funciones tipo singletons. El sistema fue simulado en *Simulink*® de Matlab utilizando el modelo de un motor a pasos híbrido a 1.8° por paso. Los resultados comentados por el autor muestran que el sistema es capaz de orientarse en base al mayor gradiente de radiación solar.

El artículo de Mohamed A. et al [40] muestra la simulación del control de posición de un robot utilizando motores a pasos y lógica difusa. El sistema de control no se muestra en el artículo. Para la comprobación se simuló en *Simulink*® de Matlab utilizando un modelo de motor a pasos por paso. Los resultados muestran que el controlador alcanzó la estabilidad en un tiempo 10 s sin sobrepaso.

El artículo de Kong X. et al [41] muestra el diseño de un controlador difuso tipo PI para una máquina de pruebas no destructivas embebido dentro de un DSP TMS320F240. Las funciones de membresía de salida como la de entrada no son mostradas. El resultado con el controlador difuso es comparado con uno tipo PI clásico. El controlador difuso tuvo una mejoría de 90% con respecto al sobrepaso en el tipo PI. El tiempo de estabilización fue de 70 milisegundos con respecto al clásico.

El artículo de Melin P. et al [42]–[44] presenta el diseño de un controlador difuso utilizando el método de inferencia de Sugeno y ANFIS. Se utiliza una serie de datos del error de posición de un motor a pasos y mediante la herramienta de ANFIS del *Toolbox* de *Matlab* se generaron 9 reglas como funciones $f(x)$. El controlador fue implementado dentro de un microprocesador y un microcontrolador SX28 como contador para un *encoder* que mide el número de grados por paso del motor. Los resultados se compararon con un controlador tipo PID, así como uno utilizando Mamdani, ambas comparaciones mostraron como mejor resultado el diseñado con ANFIS.

El artículo de Chen C. [45] presenta un modelo T-S-K (Takagi-Sugeno-Kant) autoorganizado neuro-fuzzy-recurrente (T-SORNFN) como propuesta para el control de trayectoria de un motor a micropasos (LMSM). Los resultados mostraron que la técnica de T-SORNFN tiene buenos resultados y es posible ser aplicado para el seguimiento de trayectorias para motores a micropasos.

El reporte técnico de Dannenberg A. en [46] presenta el desarrollo de un sistema embebido en un microcontrolador MSP430 de la empresa *Texas Instrument*. El autor desarrolla un sistema difuso tipo PD utilizando como variables a la entrada, error y cambio del error. El error corresponde a tres funciones triangulares y dos trapezoidales que cierran el universo, misma manera para el cambio en el error. A la salida se tiene 5 singletons que activan un PWM para mover un motor de DC. Los resultados mostraron que el sistema tuvo un sobre de 20% sobre el valor deseado y un tiempo de estabilización de 1 segundo.

El artículo de Bawa D. et al [47] presenta el diseño de un controlador difuso tipo PD implementado dentro de una tarjeta arduino UNO para el control de posición de un seguidor solar a través de un motor a pasos de 1.8° por paso. Para la variable error las funciones de membresía fueron 7 funciones triangulares distribuidas en un universo de -0.8 a 0.8 y para el cambio en el error de idéntica manera. A la salida fueron 7 funciones triangulares distribuidas idénticamente que proporcionan un ciclo PWM en porcentaje para accionar el motor. Los resultados fueron comparados utilizando retroalimentación sin controlador, mostrando tener una mejoría del 20% con respecto al sistema sin controlador.

CAPÍTULO 2 CONCEPTOS Y FUNDAMENTOS TEÓRICOS

2.1. Sustentación de un UAV

La sustentación se define como una fuerza contraria al peso de un avión, y se genera por un viento que viaja en sentido contrario a ésta, que al chocar con el ala genera un sistema de alta presión sobre la parte inferior del ala, que se traduce como una fuerza de levantamiento (fig. 11)

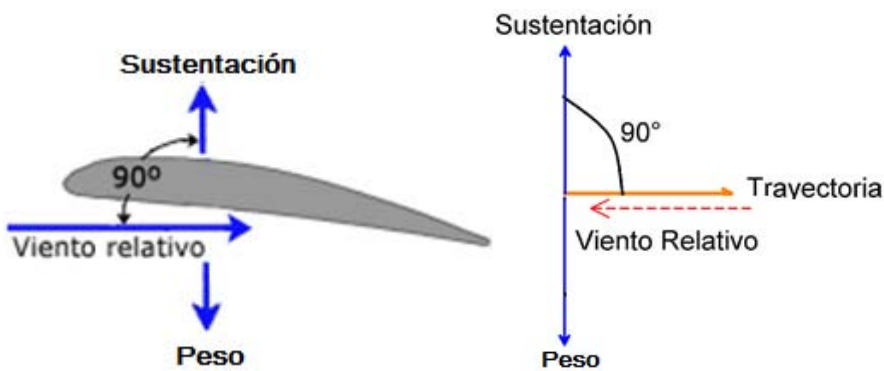


Figura 11. Sustentación en una aeronave [48]

El principio de la sustentación parte del principio de conservación de la energía o la ecuación de Bernoulli [48], aplicado este principio del punto A al B (fig. 12):



Figura 12. Efecto de sustentación en un ala [48]

Representado por la ec. (1)

$$P_A + \rho gh_A + \frac{1}{2}\rho v_A^2 = P_B + \rho gh_B + \frac{1}{2}\rho v_B^2 \quad (1)$$

Donde:

P_A - la presión en el punto A [N/m²]

ρ - densidad del fluido circundante [kg/m³]

g - constante de la gravedad [m/s²]

h_A - la altura en el punto A [m]

v_A^2 - la velocidad al cuadrado en el punto A [m/s]

P_B - presión en el punto B [N/m²]

h_B - la altura en el punto B [m]

v_B^2 - la velocidad al cuadrado del punto B [m/s]

El cual es conocido también como el principio de Bernoulli aplicado a fluidos [48].

$$\frac{1}{2}\rho(v_A^2 - v_B^2) = P_B - P_A + \rho g(h_B - h_A) \quad (2)$$

despejando P_B y eliminando la altura (despreciable en comparación con las dimensiones del propio avión), se tiene la presión en el punto B o parte inferior del ala, la presión en B es la presión de A más una constante, es decir, que la presión en B es mayor que la de A y para mantener la igualdad en el punto A, se tiene la ec. (3)

$$P_B = \frac{1}{2} \rho (v_A^2 - v_B^2) + P_A \quad (3)$$

2.2. Elevación

El ángulo α (fig. 13) se encuentra relacionado con el ángulo de ataque del perfil alar de la aeronave [48], que se genera a partir de una línea imaginaria llamada cuerda que divide el perfil alar en dos.



Figura 13. Ángulo de ataque referido a un tipo de perfil alar [48]

Existen diferentes tipos de perfiles alares, cada uno tiene ligado su propia gráfica de ángulo de ataque vs el coeficiente de sustentación. El coeficiente de sustentación es un parámetro adimensional necesario para obtener su valor. La sustentación se define por la ec. (4):

$$L = C_L \frac{1}{2} \rho v^2 S \quad (4)$$

Donde:

L - a la sustentación de la aeronave [N]

C_L - es el coeficiente de sustentación [u.a]

ρ - es la densidad del aire a la temperatura ambiental [kg/m³]

v - velocidad relativa de la aeronave [m/s]

S - superficie alar [m²]

Los valores C_L son obtenidos experimentalmente al someter los diferentes tipos de perfiles alares a pruebas en túnel de viento. A los diferentes tipos de perfiles se les denomina NACA (*National Advisory Committee for Aeronautics*), siendo ésta una agencia federal de Estados Unidos que se fundó en 1915 para fomentar las investigaciones aeronáuticas, ésta agencia se disolvió en 1958 para darle paso a lo que es actualmente la NASA (*National Aeronautics and Space Administration*).

En la figura 14 se tiene la gráfica de ángulo de ataque vs coeficiente de sustentación para el perfil NACA 0012, su ángulo de ataque va desde los 0° hasta los 16°, a partir de ángulos mayores a 16° disminuye la sustentación y se corre el riesgo de perder el control de la aeronave, depende del experto conocer estas variables para mantener la estabilidad y la sustentación de la aeronave para un correcto vuelo, (ec. (4)), de 20° en adelante es posible mantener la sustentación utilizando los *flaps* del ala, el cual es un mecanismo que modifica la geometría de la aeronave y permite tener una ganancia en la sustentación (fig. (15)).

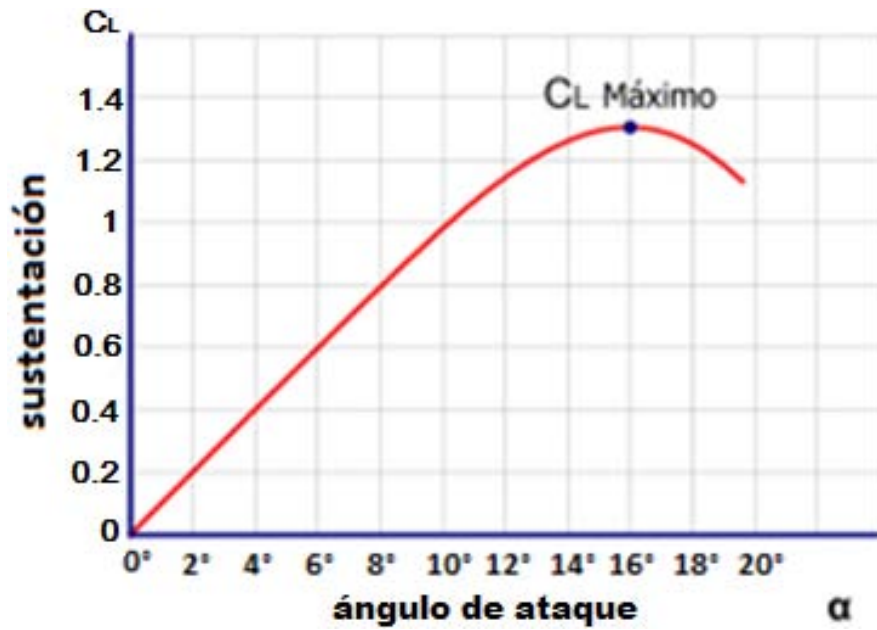


Figura 14. Relación ángulo de ataque y coeficiente de sustentación para el perfil alar

NACA 0012 [49]

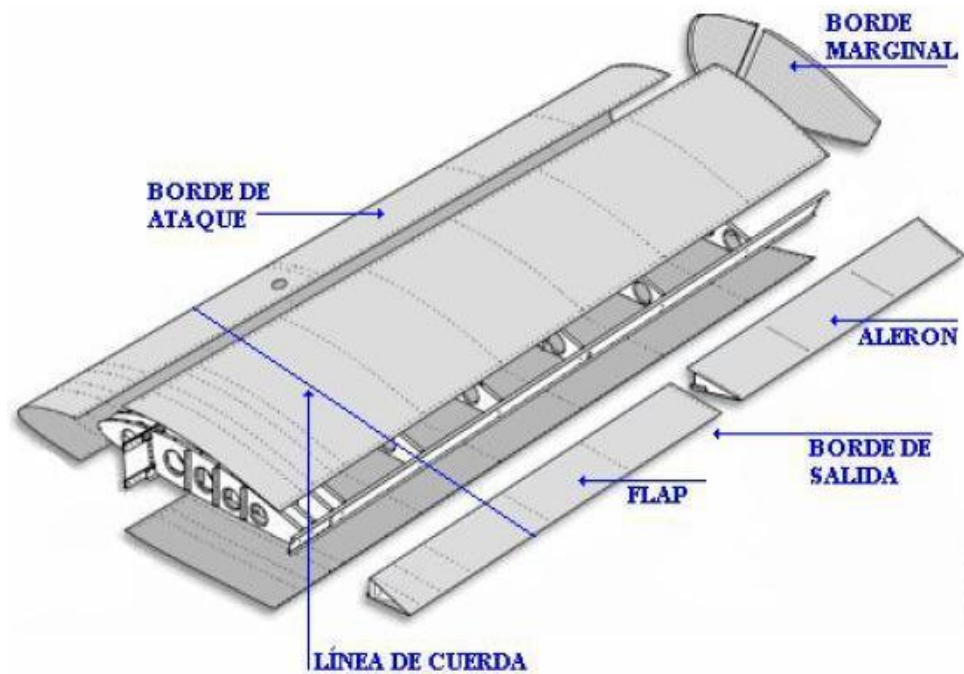


Figura 15. Partes principales del ala de un avión [49]

2.2.1. Tipos de perfil alar

Existen 3 tipos de familias NACA, la familia denominada cuatro y cinco dígitos y la serie 6.

En la familia de cuatro dígitos son tipos de perfiles con nomenclatura de 4 cifras, el primero de los dígitos da la curvatura en porcentaje de la cuerda; el segundo da la posición de la curvatura máxima en décimas de la cuerda y los dos últimos dígitos el espesor máximo en porcentaje de la cuerda. Por ejemplo, un perfil NACA 2415 tiene la curvatura máxima del 2% de la cuerda, situada en el punto del 40% de la cuerda (medido desde el borde frontal del perfil) y con un espesor máximo del 15% de la cuerda. El perfil NACA 0012 es un perfil simétrico (de curvatura 0) y con un espesor del 12% de la cuerda.

La familia de cinco dígitos surgió de experimentos y estudios empíricos con el objetivo de incrementar el coeficiente de sustentación máximo de los diferentes tipos de perfil alares ya existentes. Los resultados demostraron que si se acercaba la cuerda en dirección al borde de ataque se obtenía un incremento en el coeficiente de sustentación, (fig. 15). El primer dígito proporciona el coeficiente de sustentación de diseño en décimas multiplicado por $3/2$. El segundo y tercer dígito forman un número que multiplicado por $1/2$ otorga la ubicación máxima en porcentaje de la longitud de cuerda tomada desde el borde de ataque y los últimos dos dígitos dan el grosor máximo del perfil alar. Por ejemplo el NACA 23012, el primer dígito que corresponde al coeficiente de sustentación es de 0.3, el segundo dígito corresponde al 15% de la longitud de cuerda, los últimos dos dígitos dan el grosor máximo del perfil alar que en este caso es del 12% de la longitud de cuerda.

Los perfiles de serie 6 son perfiles nacidos en la era supersónica, para aviones que sobrepasan la velocidad del sonido, en estos perfiles, el primer dígito indica la serie y el segundo la posición de presión mínima en décimas de la cuerda, el tercer dígito representa el coeficiente de sustentación en décimas y los dos últimos dígitos el espesor en porcentaje de la cuerda. Por ejemplo, el NACA 64-212 es un perfil con el punto de presión mínima en

el 40% de la cuerda, un coeficiente de sustentación de diseño de 0.2 y espesor del 12% de la cuerda.

2.3. Teoría de Lanzaderas

2.3.1. Lanzadera tipo elástica

En ésta sección se realiza un estudio de las fuerzas en un UAV consideradas para su despegue. Basado en el trabajo del autor Novakovic Z. [8], se muestra una lanzadera que consta de 6 elementos: 1) riel de lanzamiento, 2) cuna o soporte donde se monta el UAV, 3) cuerdas elásticas, 4) polea trasera, 5) polea frontal y 6) el UAV. La lanzadera considera un rango de α de 5° a 15° , (fig. 16).

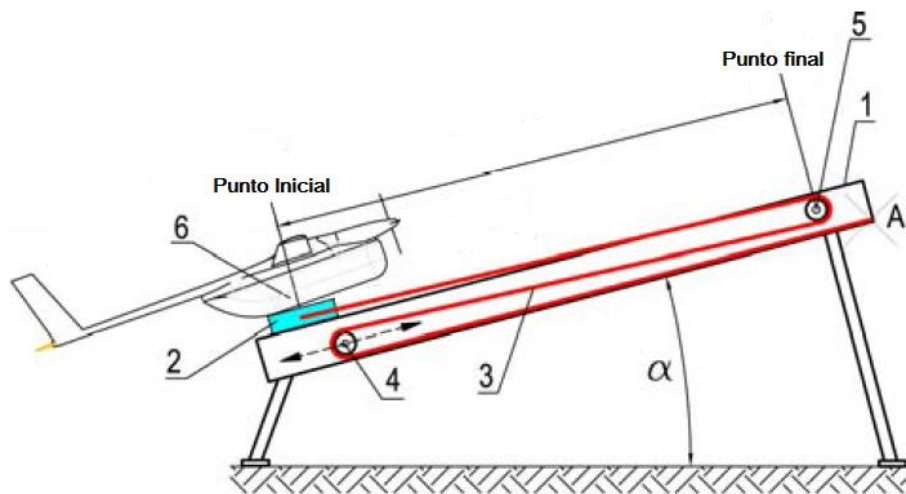


Figura 16. Esquema por partes de la lanzadera [8]

De la figura 16, la cuna o montura donde va soportado el UAV (2), tiene la posibilidad de moverse a lo largo del plano inclinado (riel de lanzamiento) (1) del punto inicial al punto final, el conjunto de poleas están conectadas a la cuna o montura por un resorte y éste está enrollada sobre dos poleas una frontal (5) y una trasera (4) que al final se asegura sobre el

riel de lanzamiento (punto A). La forma de tensión de la cuerda elástica es a través de la cuna ajustada sobre el punto inicial donde ésta se encuentra asegurada.

El autor presenta el análisis matemático tomando 12 suposiciones, 1) la lanzadera es un sistema considerando sólo el UAV como una partícula, 2) la masa del resorte utilizado es despreciable, 3) la masa de las poleas se omite, 4) la dirección de las fuerzas son coplanares al eje de referencia tomado como un plano inclinado, 5) la fricción de los resortes sobre las poleas es despreciable, 6) el grosor del resorte es constante en todo punto, 7) la fuerza de fricción al momento del despegue es sumamente pequeña por lo que se desprecia, 8) la dirección de las fuerzas es consistente a la dirección de elongación de la cuerda elástica, (fig. 17).

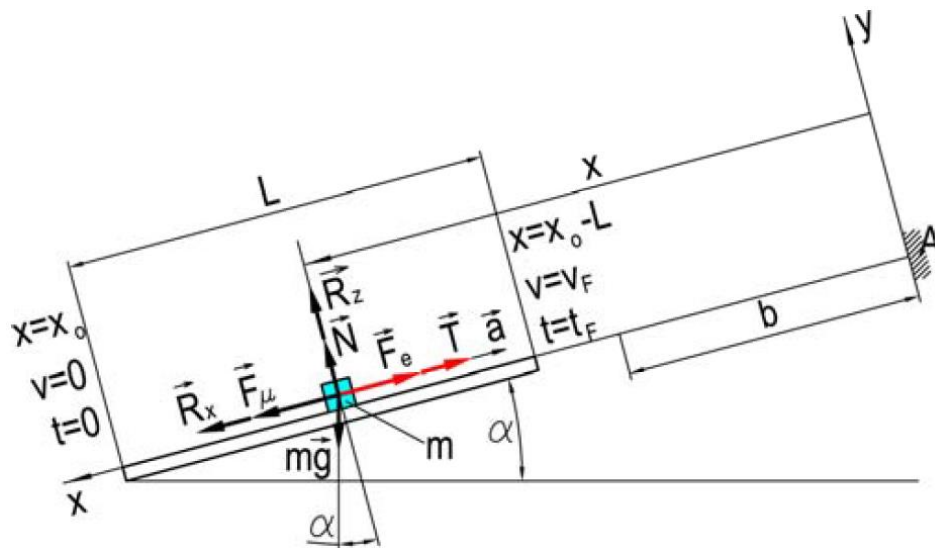


Figura 17. Diagrama de cuerpo libre de una lanzadera con un ángulo α [8]

La ecuación planteada es para el movimiento de una partícula:

$$m\vec{a} = \sum \vec{F} \quad (5)$$

Considerando en (5) las fuerzas que se ejercen en el sistema de lanzamiento del UAV, se obtiene la ec. (6):

$$m\vec{a} = \vec{F}_e + m\vec{g} + \vec{F}_\mu + \vec{R}_x + \vec{R}_z + T + N \quad (6)$$

La proyección en el eje x coincide con la dirección de la fuerza de la cuerda elástica:

$$F_e = q(x - b) \quad (7)$$

Donde:

F_e - fuerza elástica [N]

q - constante elástica del resorte [N/m]

x - distancia de elongación del resorte [m]

b - distancia total de carrera [m]

La fuerza de arrastre del UAV es:

$$R_x = C_x \frac{\rho v^2}{2} S \quad (8)$$

R_x - fuerza de arrastre del UAV [N]

C_x - coeficiente adimensional de arrastre [u.a]

ρ - es la densidad del aire a la temperatura ambiental [kg/m³]

v - velocidad relativa de la aeronave [m/s]

S - superficie alar [m²]

La fuerza de sustentación del UAV es:

$$R_z = C_z \frac{\rho v^2}{2} S \quad (9)$$

La fuerza de empuje de las hélices del UAV es [8]:

$$T = T_0 \left(1 - \frac{\dot{x}}{v_s}\right) \cong T(\dot{x}) \quad (10)$$

La fuerza de fricción sobre el riel es:

$$F_\mu = \mu N \quad (11)$$

Donde:

F_μ - fuerza de fricción [N]

μ - coeficiente de fricción cinemático [u.a]

N - fuerza normal [N]

El autor realiza cuatro suposiciones más, 1) la fuerza de arrastres R_x se desprecia, 2) la fuerza de sustentación R_z es desprecia, 3) la fuerza del empuje de las hélices del UAV se considera constante.

Aplicando las suposiciones hechas a (6) se obtiene la ec. (11):

$$m\vec{a} = \vec{F}_e + m\vec{g} + \vec{F}_\mu + \vec{T} + \vec{N} \quad (11)$$

y proyectando en el eje x (ver. fig. 17) se obtiene:

$$m\ddot{x} = F_\mu + mg\text{sen}\alpha - F_e - T \quad (12)$$

del análisis de las fuerzas que interactúan en el eje Y se obtiene:

$$N = mg\text{cosen}\alpha \quad (13)$$

Sumando las ec. (12) y (13) se obtiene:

$$\ddot{x} + \frac{q}{m}x = g(\mu\text{cosen}\alpha + \text{sen}\alpha) + \frac{q}{m}b - \frac{T}{m} \quad (14)$$

Siendo la ec. (14) de la forma:

$$\ddot{x} + kx = A\text{cosen}\alpha + B\text{sen}\alpha + C \quad (15)$$

La ec. (14) es una ecuación diferencial lineal no homogénea de orden dos y su solución es:

$$x(t) = C_1 \cos \sqrt{\frac{q}{m}} t + C_2 \sin \sqrt{\frac{q}{m}} t + \frac{mg}{q} (\mu \cos \alpha + \sin \alpha) + b - \frac{T}{q} \quad (16)$$

Tomando las definiciones anteriores de condiciones iniciales: $t = 0$, $x = x_0$, la ecuación (16) toma la forma:

$$x(t) = \left[x_0 + \frac{T}{q} - \frac{mg}{q} (\mu \cos \alpha + \sin \alpha) - b \right] \cos \sqrt{\frac{q}{m}} t + \frac{mg}{q} (\mu \cos \alpha + \sin \alpha) + b - \frac{T}{q} \quad (17)$$

La ec. (14) representa la ecuación de movimiento de la lanzadera. Derivando (17) se obtiene la ecuación de velocidad:

$$\dot{x}(t) = - \left[x_0 + \frac{T}{q} - \frac{mg}{q} (\mu \cos \alpha + \sin \alpha) - b \right] \sqrt{\frac{q}{m}} \sin \sqrt{\frac{q}{m}} t \quad (18)$$

Aplicando condiciones iniciales de valores finales, $t = t_F$, $x(t_F) = x_0 - L$ en la ec. (16), la expresión del momento en el que el UAV llega al punto final es:

$$t_F = \sqrt{\frac{m}{q}} \arccos \frac{x(t_F) - \frac{mg}{q} (\mu \cos \alpha + \sin \alpha) - b + \frac{T}{q}}{x_0 + \frac{T}{q} - \frac{mg}{q} (\mu \cos \alpha + \sin \alpha) - b} \quad (19)$$

Si se obtiene la derivada de (18), la expresión de la aceleración es:

$$\ddot{x}(t) = - \left[x_0 + \frac{T}{q} - \frac{mg}{q} (\mu \cos \alpha + \sin \alpha) - b \right] \sqrt{\frac{q}{m}} \cos \sqrt{\frac{q}{m}} t \quad (20)$$

2.3.1. Lanzadera tipo neumática

Basado en el trabajo [19] para una lanzadera neumática, se utiliza un sistema de aire a presión para acelerar el UAV a su velocidad máxima de despegue (fig. 18).

El cálculo de la fuerza necesaria en función de la presión del pistón se expresa como:

$$F = PA \quad (21)$$

Donde:

F - Fuerza del pistón [N]

P - Presión del pistón [Pa]

A - Área [m²]

El pistón es de área transversal circular, por tal:

$$A = \frac{\pi D^2}{4} \quad (22)$$

Donde:

A - área de la sección transversal [m²]

π - constante pi [u.a]

D - diámetro [m]

Sustituyendo (22) en (21) se tiene:

$$F = P \frac{\pi D^2}{4} \quad (23)$$

donde:

F - fuerza del pistón [N]

π - constante pi [u.a]

D - diámetro [m]

P - presión del pistón [N]

La ec. (23) representa la expresión para encontrar la fuerza necesaria que debe soportar el pistón.

$$D = \sqrt{\frac{4F}{P\pi}} \quad (24)$$

La ec. (24) denota el diámetro necesario que debe tener el pistón para la fuerza requerida para acelerar el UAV.

El largo del brazo del pistón está en función de la carrera necesario para el despegue del UAV.

Ésta lanzadera es del tipo neumática que incluye un compresor eléctrico conectado a una válvula controladora para le apertura y cierre y de un pistón neumático, el cual está conectado a una carretilla donde se encuentra montado un UAV, la carretilla se desplaza por un eslabón hasta llegar a un tope con un sensor de fin de carrera que detiene su avance al desactivar la válvula del compresor. Cuenta con un cilindro que le permite elevar su posición para alcanzar un ángulo deseado, fig. 18.

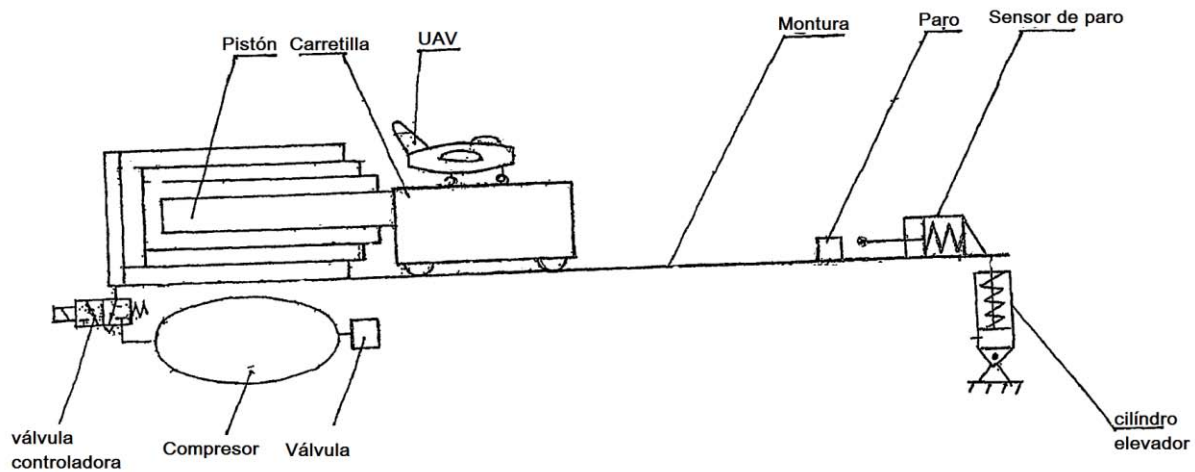


Figura 18. Lanzadera neumática por partes [28]

2.4. Sistemas de Orientación

La orientación está dada por tres ejes ortogonales denominados: rotación, cabeceo y guiñada. La denominación de estos ejes es análogo a la utilizada en aviación y se describen de la siguiente manera: el eje de rotación se encuentra localizado de manera axial al vector velocidad, el eje de guiñada se define como el vector que une el centro de masa con el nadir y, el eje de cabeceo es perpendicular a los dos anteriores para formar un sistema de ejes ortogonal [50].

2.4.1. Parametrización de la Orientación

Cualquier vector \vec{A} puede ser expresado como:

$$\vec{A} = A(\cos\alpha \vec{i} + \cos\beta \vec{j} + \cos\gamma \vec{k}) \quad (25)$$

donde α , β , γ son ángulos que forma \vec{A} con los ejes \vec{i} , \vec{j} , \vec{k} respectivamente y $\cos\alpha$, $\cos\beta$ y $\cos\gamma$ son conocidos como los cosenos directores de \vec{A} [50].

Para poder determinar la orientación en tres ejes, es necesario llevar a cabo una transformación para pasar de un sistema de coordenadas fijo al cuerpo a un sistema de coordenadas inercial, de tal manera que:

$$\vec{u} \times \vec{v} = \vec{w} \quad (26)$$

El problema básico es especificar la orientación de esta terna, con respecto a algún marco de referencia fijo, (fig. 19).

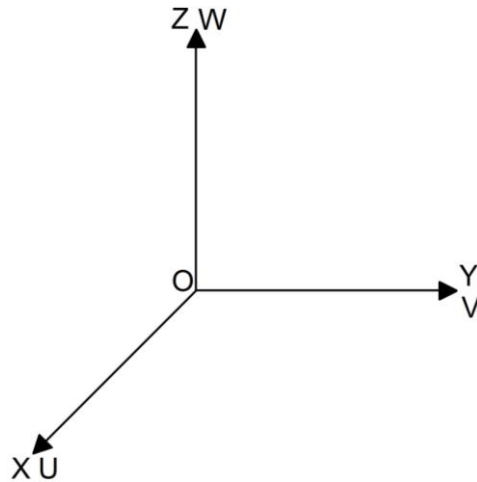


Figura 19. El problema fundamental de la parametrización de la orientación en tres ejes, consiste en especificar la orientación de los ejes u, v, w , en el marco de referencia x, y, z .

Si se puede relacionar los componentes $\vec{u}, \vec{v}, \vec{w}$ a lo largo de tres ejes fijos llamados: 1, 2, 3, se tendrá entonces la orientación completa definida. Para esto se necesita nueve parámetros que pueden ser vistos como los elementos de una matriz A de 3×3 llamada matriz de orientación:

$$A \equiv \begin{bmatrix} u_1 & u_2 & u_3 \\ v_1 & v_2 & v_3 \\ w_1 & w_2 & w_3 \end{bmatrix} \quad (27)$$

Cada uno de estos elementos es el coseno del ángulo entre un vector unitario fijo y uno de los ejes de referencia; u_1 por ejemplo, es el coseno del ángulo formado entre \vec{u} y el eje 1. Es por esta razón que a la matriz A se le llama matriz de cosenos directores [50].

Sus elementos no son todos independientes, por ejemplo, el hecho de que \vec{u} sea un vector unitario implica que:

$$u_1^2 + u_2^2 + u_3^2 = 1 \quad (28)$$

y la ortogonalidad entre \vec{u} y \vec{v} , significa que:

$$u_1 v_1 + u_2 v_2 + u_3 v_3 = 0 \quad (29)$$

Todas estas relaciones pueden resumirse con la siguiente igualdad: el producto de A por su transpuesta es la matriz identidad [50].

$$AA^T = 1 \quad (30)$$

Esto quiere decir que A es una matriz ortogonal real. La definición del determinante de A es:

$$\det A = \vec{u} \cdot (\vec{v} \times \vec{w}) \quad (31)$$

El hecho de que los vector \vec{u} , \vec{v} , \vec{w} formen una tercia ortogonal (en el sentido positivo) quiere decir que $\det A = 1$. Entonces, A es una matriz ortogonal real propia [50].

La matriz de cosenos directores es una transformación de coordenadas que mapea vectores desde el marco de referencia inercial hacia el marco fijo al cuerpo. Esto quiere decir que si a es un vector con componentes a_1 , a_2 , a_3 localizado a lo largo de los ejes de referencia, entonces:

$$A\vec{a} = \begin{bmatrix} u_1 & u_2 & u_3 \\ v_1 & v_2 & v_3 \\ w_1 & w_2 & w_3 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} \vec{u} \cdot \vec{a} \\ \vec{v} \cdot \vec{a} \\ \vec{w} \cdot \vec{a} \end{bmatrix} \equiv \begin{bmatrix} a_u \\ a_v \\ a_w \end{bmatrix} \quad (32)$$

Los componentes de $A\vec{a}$ son los mismo del vector \vec{a} a lo largo de la tercia \vec{u} , \vec{v} , \vec{w} . Una transformación de una matriz ortogonal real propia conserva las longitudes de los vectores y los ángulos entre ellos. Esto significa que puede representar una rotación. El producto de

dos matrices reales propia $A'' = A'A$ representa los resultados de rotaciones sucesivas, primero por A y luego por A' , en ese orden. Debido a que la transpuesta y la inversa de una matriz ortogonal son idénticas, A^T mapea los vectores fijos al cuerpo, hacia el marco de referencia inercial [50].

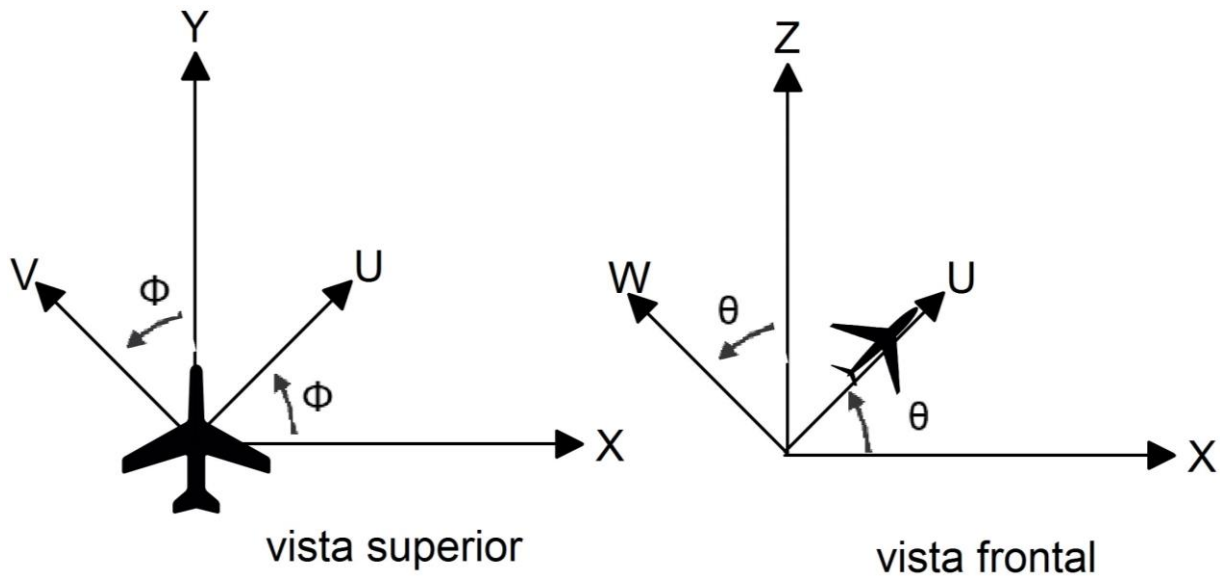
La matriz de cosenos directores es vista como una relación fundamental que nos especifica la orientación de un cuerpo rígido. Sin embargo, algunas otras parametrizaciones que se resumen en la tabla 1 pueden ser más convenientes dependiendo de la aplicación [50]. En todos los casos, siempre se relacionará los parámetros con la matriz de cosenos directores.

Tabla 1. Representaciones alternativas de la orientación en tres ejes [50]

Parametrización	Notación	Ventajas	Desventajas	Aplicaciones más comunes
Matriz de cosenos directores	$A = [A_{ij}]$	No existe singularidades ni funciones trigonométricas. La regla de productos es adecuada para representar rotaciones sucesivas.	Seis parámetros redundantes	En el análisis para transformar vectores de un marco de referencia a otro
Ejes de Euler y ángulos de Euler	e, Φ	Clara representación física	Un parámetro redundante. Un eje indefinido cuando $\Phi = 0$.	Comando de maniobras de orientación.
Parámetros simétricos de Euler (cuaterniones)	q_1, q_2, q_3, q_4	No tiene singularidades ni funciones trigonométricas. La regla de productos es adecuada para representar rotaciones sucesivas	Un parámetro redundante. No hay una interpretación física obvia	Navegación inercial a bordo
Vector de Gibbs	g	No existen singularidades ni funciones trigonométricas. La regla de productos es adecuada para representar rotaciones sucesivas	Valor infinito para una rotación de 180°	Estudios analíticos
Ángulos de Euler	Φ, θ, ψ	No tiene parámetros redundantes. La interpretación física es clara en algunos casos	Con algunos valores de θ presenta singularidades. La regla de productos no es adecuada para representar rotaciones sucesivas	Estudios analíticos. Entrada/Salida. Control de orientación a bordo en naves estabilizadas en tres ejes

2.4.2. Ecuaciones de movimiento

Para modelar la dinámica se propone la secuencia ZY, cuyas matrices de rotación básicas son: R_1^2 y R_2^b , (fig. siguiente).



El subíndice denota el marco de referencia inicial y el superíndice denota el marco de referencia final. Estas matrices están definidas de la siguiente manera:

$$R_1^2 = \begin{bmatrix} \cos\theta & 0 & -\text{sen}\theta \\ 0 & 1 & 0 \\ \text{sen}\theta & 0 & \cos\theta \end{bmatrix} \quad R_2^b = \begin{bmatrix} \cos\Phi & -\text{sen}\Phi & 0 \\ \text{sen}\Phi & \cos\Phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (33)$$

El orden de la multiplicación para obtener R_1^b debe ser: R_2^b, R_1^2 , de tal manera que la matriz de rotación queda:

$$R_1^b = \begin{bmatrix} \cos\Phi\cos\theta & -\text{sen}\Phi & -\text{sen}\theta\cos\Phi \\ \text{sen}\Phi\cos\theta & \cos\Phi & -\text{sen}\Phi\text{sen}\theta \\ \text{sen}\theta & 0 & \cos\theta \end{bmatrix} \quad (34)$$

y las ecuaciones de movimiento se pueden escribir como:

$$\omega^b = \dot{\alpha} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} + \dot{\theta} R_2^b \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \quad (35)$$

$$\begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \dot{\alpha} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} + \dot{\theta} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & \sin\alpha \\ 0 & -\sin\alpha & \cos\alpha \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \quad (36)$$

$$\begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \dot{\alpha} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} + \dot{\theta} \begin{bmatrix} 0 \\ \cos\alpha \\ -\sin\alpha \end{bmatrix} \quad (37)$$

$$\begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & \cos\alpha & 0 \\ 0 & -\sin\alpha & 0 \end{bmatrix} \begin{bmatrix} 0 \\ \dot{\theta} \\ \dot{\alpha} \end{bmatrix} \quad (38)$$

\vec{H}_0 , el vector de momento angular alrededor del centro de masa puede escribirse como:

$$\vec{H}_0 = M(\vec{r} \times \vec{V}) \quad (39)$$

Donde M es la masa total del sistema, r es el vector que conecta el centro de rotación con el centro de masa y V es la velocidad correspondiente al centro de masa. Por otra parte, se puede demostrar que la tasa de variación del momento angular con respecto del tiempo, es igual a las fuerzas externas aplicadas a la plataforma, es decir:

$$\frac{dH_0}{dt} = M_0 \quad (40)$$

donde M_0 son todos los pares aplicados a la plataforma (internos, gravitacionales y aerodinámicos) [50].

Los momentos externos aplicados M_0 , pueden ser descompuestos en:

$$M_0 = M_{internos} + M_g + M_{aereo} \quad (41)$$

donde $M_{internos}$ son los pares internos, M_g es el par gravitacional aplicado al centro de masa y M_{aereo} es el par de arrastre aerodinámico cuando la plataforma se encuentra girando [50].

La lanzadera tiene una longitud L , masa M y densidad de masa λ lineal que gira sobre el eje y para el ángulo de rotación θ y z para el ángulo de ataque. El momento de inercia para el eje y se muestra en la figura 20.

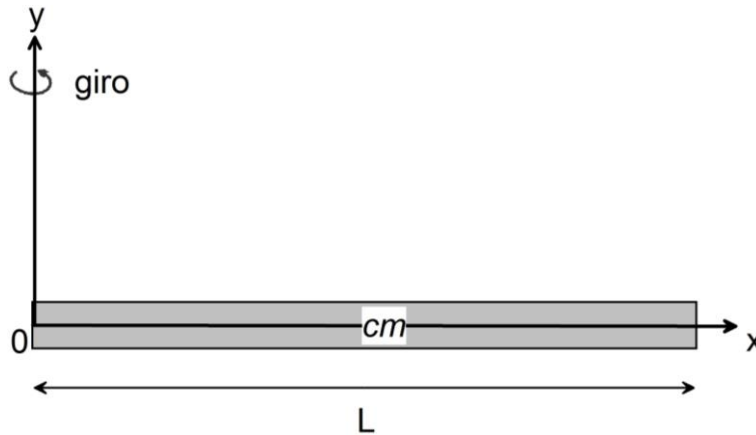


Figura 20. Barra de longitud L , con masa M y densidad de masa lineal.

Para un cuerpo de masa continuo lineal su momento de Inercial se calcula a partir de la ec. (42):

$$I_{eje} = \int_m r^2 dm \quad (42)$$

Donde I_{eje} es momento de inercial sobre el eje indicado, m es toda la masa del sólido integrada sobre un diferencia de masa dm , r es la distancia al eje de rotación.

El cálculo del momento de inercia consiste en expresar cualquier elemento de masa dm a lo largo del eje. Para realizar la integral, es necesario expresar todo lo que haya en la integral en términos de una variable de longitud r . Dado que el total de la longitud L tiene masa M , entonces M/L es la proporción de masa sobre longitud y entonces el elemento de masa dm se puede expresar como se muestra en la ec. (43)

$$I_0 = \int_0^L r^2 \frac{M}{L} dr \quad (43)$$

desarrollando la integral de la ec. (43) se tiene

$$I_0 = \frac{M}{L} r^2 dr \Big|_0^L \quad (44)$$

$$I_0 = \frac{M}{L} \frac{r^3}{3} \Big|_0^L \quad (45)$$

$$I_0 = \frac{1}{3} ML^2 \quad (46)$$

la ec. (46) corresponde al momento de inercia en 0 sobre el eje y . El resultado se comprueba a partir del teorema de Steiner. El teorema de Steiner dice que el momento de inercia sobre cualquier eje es igual al momento de inercia del centro de masa más la multiplicación de la masa por la distancia al cuadrado, ec. (47), donde cm indica el centro de masa.

$$I = I_{cm} + Md^2 \quad (47)$$

donde I_{cm} corresponde a la ec. (48)

$$I_{cm} = \int_{-\frac{L}{2}}^{\frac{L}{2}} r^2 \frac{M}{L} dr \quad (48)$$

resolviendo la ec. (48)

$$I_{cm} = \frac{1}{12} ML^2 \quad (49)$$

aplicando el teorema de Steiner en la ec. (47) a la ec. (49) se obtiene

$$I = \frac{1}{12}ML^2 + M\frac{l^2}{4} \quad (50)$$

$$I = \frac{1}{3}ML^2 \quad (51)$$

lo que la ec. (51) comprueba el resultado de la ec. (46).

El momento de inercia para el ángulo de ataque sería aplicar la ec. (42) sobre el eje z, donde el diferencial de masa dm sería M/L , idéntico al aplicado para el eje y, por tal, el momento de inercia sobre el eje z es igual al obtenido en el eje y.

El tensor de inercia se define como la ec. (52) [51]:

$$I = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{bmatrix} \quad (52)$$

donde la componente I_{yy} e I_{zz} corresponde a los momentos de inercia calculado en la ec. (51). Las componentes restantes se calculan a partir de las ec. (53) a la ec. (56):

$$I_{xx} = \int_m x^2 dm \quad (53)$$

$$I_{xy} = \int_m -xy dm \quad (54)$$

$$I_{yz} = \int_m -yz dm \quad (55)$$

$$I_{zx} = \int_m -zx dm \quad (56)$$

El momento de inercia sobre su mismo eje es cero, por tal la ec. (53) se cancela, los momentos de inercia de la ec. (54) a la ec. (56) son cero, ya que no existe direcciones rotadas en la fig. (20).

Por lo que el tensor de inercia de la lanzadera se escribe como la ec. (57):

$$I = \frac{1}{3}ML^2 \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (57)$$

El momento (torque) aplicado a la lanzadera se define como la ec. (58):

$$\tau = I\dot{\omega} \quad (58)$$

Donde:

- I es el tensor de inercia de la ec. (57)

- $\dot{\omega}$ es la aceleración angular de la lanzadera

El momento aplicado a la lanzadera se escribe como la ec. (59)

$$\tau = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \frac{1}{3}ML^2 & 0 \\ 0 & 0 & \frac{1}{3}ML^2 \end{bmatrix} \begin{bmatrix} \dot{\omega}_x \\ \dot{\omega}_y \\ \dot{\omega}_y \end{bmatrix} \quad (59)$$

La energía cinética de la lanzadera se escribe como la ec. (60):

$$E_c = \frac{1}{2}I\omega^2 \quad (60)$$

donde:

- I es el tensor de inercia de la ec. (57)

- ω^2 es la velocidad angular de la lanzadera al cuadrado

El momento aplicado a la lanzadera se escribe como la ec. (61)

$$\tau = \frac{1}{2} \begin{bmatrix} 0 & 0 & 0 \\ 0 & \frac{1}{3}ML^2 & 0 \\ 0 & 0 & \frac{1}{3}ML^2 \end{bmatrix} \begin{bmatrix} \omega_x^2 \\ \omega_y^2 \\ \omega_z^2 \end{bmatrix} \quad (61)$$

El momento angular o la conservación del momento se define como la ec. (62):

$$L = I\omega \quad (62)$$

donde:

- I es el tensor de inercia de la ec. (57)

- ω es la velocidad angular de la lanzadera

El momento angular aplicado a la lanzadera se escribe como la ec. (63)

$$L = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \frac{1}{3}ML^2 & 0 \\ 0 & 0 & \frac{1}{3}ML^2 \end{bmatrix} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \quad (63)$$

2.4. Lógica Difusa

Una de las disciplinas matemáticas relativamente nuevas es la lógica difusa, ésta es aplicada a conceptos que pueden tomar una cantidad cualquiera de veracidad dentro de un conjunto de valores que oscilan entre dos extremos, la verdad absoluta y la falsedad total [52].

Definición: La lógica difusa es una lógica alternativa a la booleana que pretende introducir un grado de exactitud a la ambigüedad e imprecisión en los hechos que evalúa [52].

Ésta lógica fue introducida por L. Zadeh en 1965 [53]. Fue diseñada en un principio para explicar la vaguedad del lenguaje humano, llamado principio de incompatibilidad.

2.5. Elementos de un conjunto difuso

Los conjuntos difusos son elementos definidos a través de funciones, llamadas funciones de membresía (*FM*), donde cada valor existente en el dominio de él, se le asocia un valor de pertenencia en su rango.

2.5.1. Tipos de Funciones de Membresía (*FM*)

Las funciones de membresía agrupan un conjunto de datos y se les asocia un valor de membresía entre 0 y 1. Son también llamadas conjuntos difusos y al valor de membresía se le conoce también como valor de pertenencia.

Existen *FM* que son más comúnmente utilizadas como la triangular, trapezoidal y gaussiana, debido a su ventaja de conocerse los parámetros que la definen, como sus límites superior o inferior, altura, su valor central, así como los intervalos que la componen.

-Trapezoidal:

Esta función está definida por sus límites inferior a y superior d , los límites inferior y superior, b y c respectivamente y se define a través de tres funciones $f(x)$, tres rectas, con pendiente positiva, negativa y cero, ec. (64), (fig. 21).

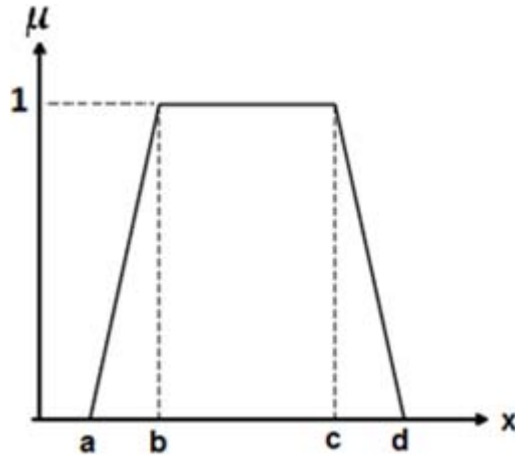


Figura 21. Función de membresía trapezoidal

$$\mu_{trapezoidal}(x; a, b, c, d) = \begin{cases} 0, & x < a \\ \frac{x-a}{b-a}, & a \leq x < b \\ 1, & b \leq x < c \\ \frac{d-x}{d-c}, & c \leq x < d \\ 0, & x \geq d \end{cases} \quad (64)$$

-Triangular:

Ésta función está definida por sus límites inferior a y superior c , y el valor central o medio b , tal que $a < b < c$. Se define a través de dos funciones $f(x)$, como una recta con pendiente positivo y negativa, definidas en ec. (65), (fig. 22).

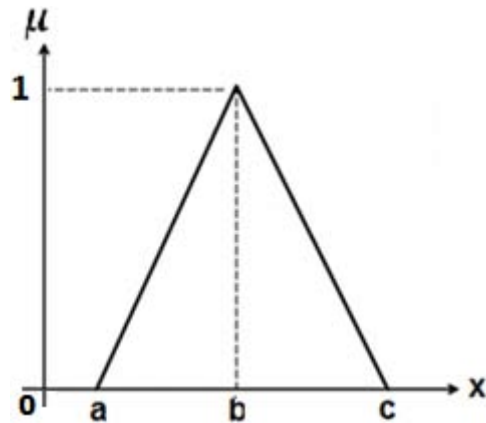


Figura 22. Función de membresía triangular

$$\mu_{\text{triangular}}(x; a, b, c) = \begin{cases} 0, & x < a \\ \frac{x-a}{b-a}, & a \leq x < b \\ \frac{c-x}{c-b}, & b \leq x < c \\ 0, & x \geq c \end{cases} \quad (65)$$

-Gaussiana:

Está definida por sus límites inferior a y superior b , su valor central c , y el valor $\sigma > 0$, cuanto mayor es σ , más estrecha es la función, definida en ec. (66), (fig. 23).

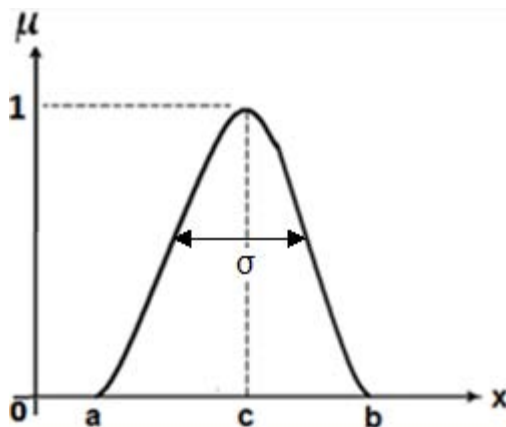


Figura 23. Función de membresía gaussiana

Está definida de la siguiente manera:

$$\mu_{Gaussiana}(x; c, \sigma) = \begin{cases} 0, & x < a \\ e^{-\frac{(x-c)^2}{2\sigma^2}}, & a \leq x \leq b \\ 0, & x > b \end{cases} \quad (66)$$

Donde:

c- valor central o media

σ - varianza

2.6. Elementos de las funciones de membresía.

-Etiqueta Lingüística.

La etiqueta lingüística es un término asociado a un sólo conjunto difuso.

-Variables Lingüísticas

Son aquellas cuyos valores se representan mediante términos lingüísticos, están asociadas al dominio de la función de pertenencia. Se les conoce también como variables difusas. A los valores en el dominio de la variable lingüísticas se les conocen como valores *crisp*.

-Universo de Discurso

El universo de discurso son todos los valores definidos dentro del dominio de los conjuntos difusos. Existe un universo de discurso para cada variable lingüística.

-Valor de pertenencia

Son aquellos que están asociados al rango de las funciones de pertenencia. A estos valores se les conoce como números difusos o valor de membresía.

La integración de los elementos anteriores se muestra en la figura 24.

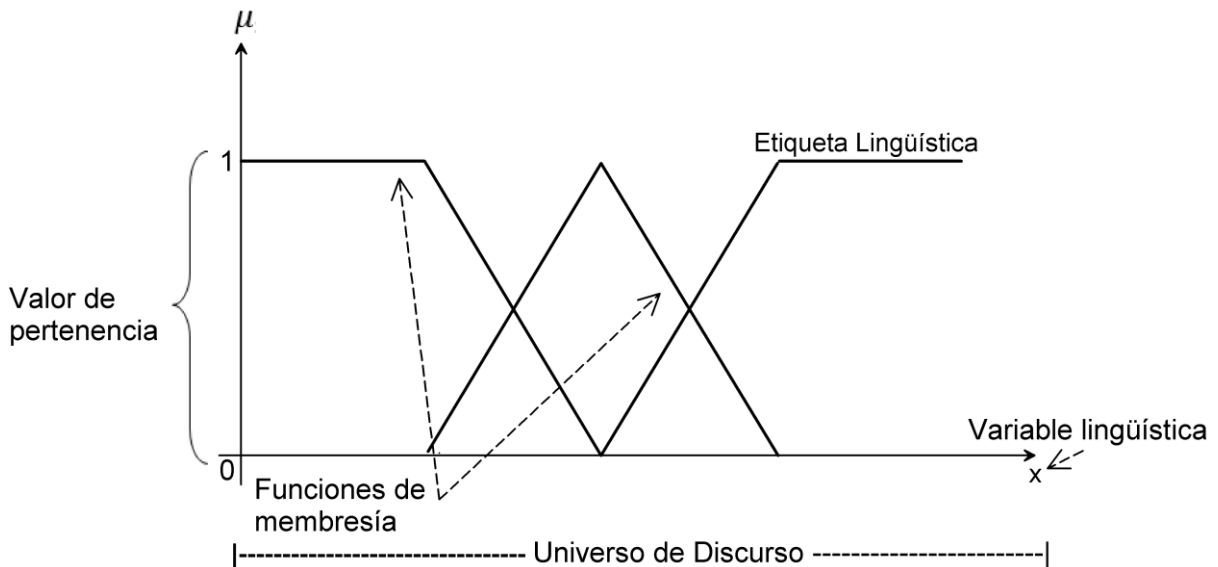


Figura 24. Partes que conforman los conjuntos difusos

2.7. Sistema Difuso.

Un sistema difuso es el medio que relaciona las variables de entrada con las de salida a través de reglas, es una estructura basada en el conocimiento a través de reglas lingüísticas del tipo *IF - THEN*, que contienen una cuantificación lingüística, llamadas funciones de membresía, para obtener una salida en el dominio del conjunto difuso.

2.7.1. Reglas Lingüísticas.

Son un conjunto de reglas que asocian una causa *IF* con un efecto *THEN*, siendo ésta asociación la manera en la cual se genera el conocimiento de un sistema difuso. Las reglas lingüísticas son conocidas también como reglas difusas y son de la forma:

IF premisa (antecedente), *THEN* conclusión (consecuente)

El conjunto de reglas difusas se le conoce como Base del Conocimiento.

2.7.2. Tipos de Sistemas Difusos.

En la actualidad los sistemas difusos más utilizados son los propuestos por Mamdani [54] y Sugeno [55].

2.7.2.1. Sistema Difuso Tipo Mamdani.

El sistema difuso tipo Mamdani [54] es un método que utiliza la teoría de conjuntos difusos y fue propuesto por E. Mamdani en 1962 [54].

El esquema general de un sistema difuso tipo Mamdani está conformado por diferentes procesos, base del conocimiento, fuzzificación y defuzzificación, así como un mecanismo de inferencia que se encarga de determinar las reglas *IF – THEN* definidas en la Base del Conocimiento, como se muestra en la figura 25.

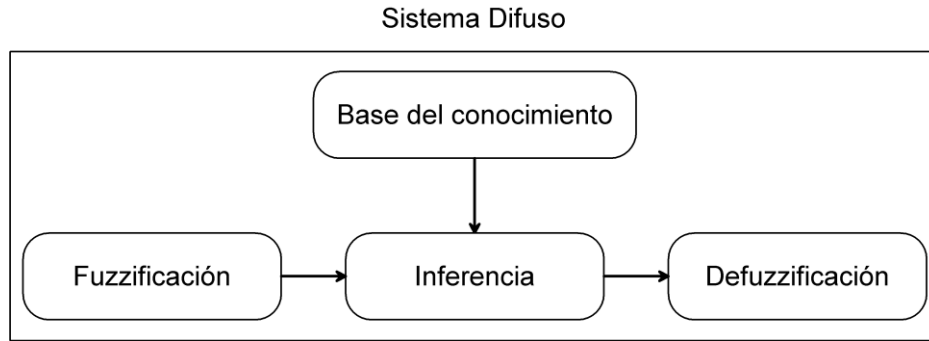


Figura 25. Esquema general de un sistema difuso

Los elementos que conforman al sistema difuso son:

-Fuzzificación.

La fuzzificación es un proceso con dos etapas, a) convertir un dato o un valor real en un valor lingüístico en el dominio y b) asignarle un valor de pertenencia en el rango del conjunto difuso correspondiente.

-Mecanismo de Inferencia.

El mecanismo de inferencia es la parte que infiere las decisiones, se debe entender a la palabra inferencia como la acción o el efecto de deducir. Está constituido por el conjunto de reglas lingüísticas que forman la Base del Conocimiento:

$$\text{IF } x \text{ es } A, \text{ THEN } y \text{ es } B,$$

en donde A y B son valores lingüísticos definidos en un conjunto difuso de un universo de discurso respectivo.

Con el conjunto de reglas lingüísticas, el mecanismo de inferencia realiza la acción o el efecto de deducir una acción para un conjunto de valores de entrada a partir de las reglas. A la inferencia se le conoce como implicación y corresponde a la premisa (cláusula IF). A éste proceso se le conoce como la parte inteligente de la lógica difusa.

Existen diferentes métodos de inferencia difusa que pueden ser formulados a través de los operadores t-normas.

Definición. Una norma triangular o t-normas, es un operador lógico Y o AND que representan una intersección.

Existen diferentes operadores lógicos propuestos por diferentes autores para inferir una solución. Los usuales se muestran en la tabla 2 [56].

Tabla 2. Operadores T-normas principales

Nombre del operador	Operación
Mínimo	$\min(x,y,\dots)$
Producto	$x \cdot y$
Producto drástico	$\begin{cases} \min(x,y) & \text{si } \max(x,y) = 1 \\ 0 & \text{en otro caso} \end{cases}$
Producto de Einstein	$\frac{xy}{1 + (1 - x) + (1 - y)}$

Comúnmente se utiliza el operador *min*, que representa la intersección entre dos funciones.

El método de inferencia utilizado por Mamdani es conocido como “min-max”, ya que utiliza como inferencia el mínimo y el máximo para el agregado.

-Defuzzificación

La defuzzificación es definida como un mapeo o asignación de un dato o conjunto de datos a un elemento o conjunto real de salida.

2.7.2.2. Método de Kulkarni

El método propuesto por Kulkarni [57]. integra una nueva etapa llamado agregado, que se encontraba implícito en el mecanismo de inferencia difuso propuesto por Mamdani. Se le conoce principalmente como el adaptado de Kulkarni, (fig. 26)

El agregado es el proceso de obtener un consecuente general o total (conclusión final) de los consecuentes individuales de cada regla activada en la inferencia, es conocido como el agregado de las reglas o la activación de las premisas de la cláusula *THEN*.

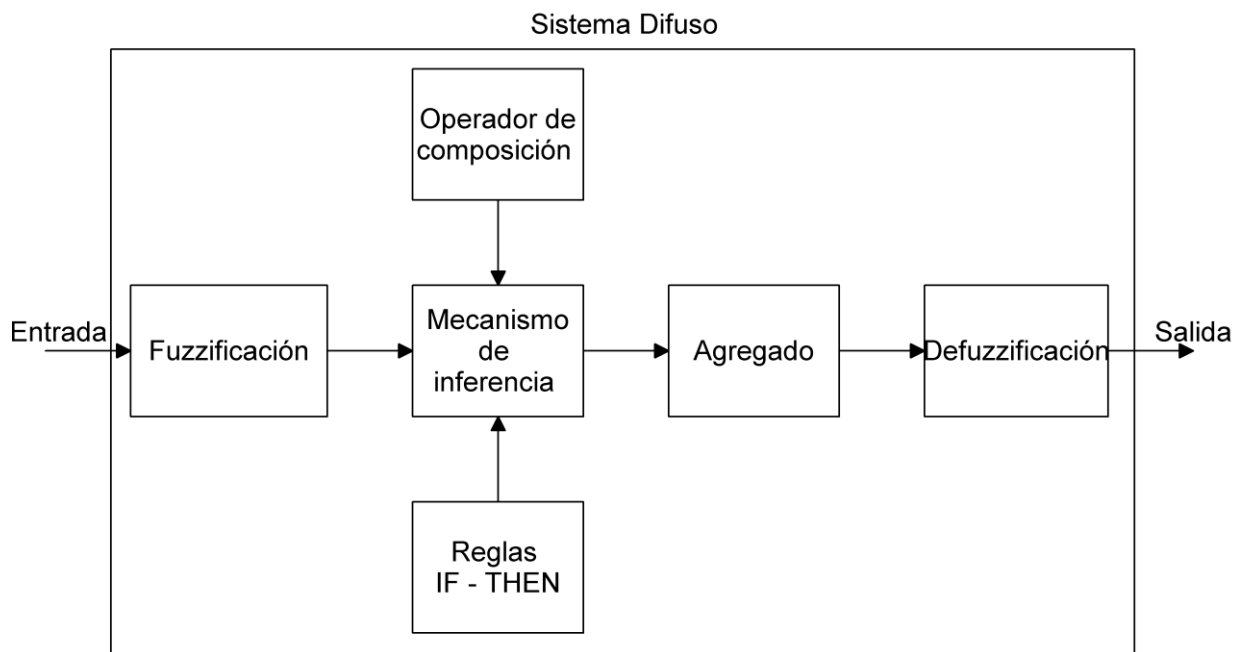


Figura 26. Sistema de inferencia difuso [57]

Como sucede con la inferencia, el agregado también puede definirse con operadores llamados s-normas. Algunos autores suelen llamarles a estos operadores t-conormas [56].

Definición. Una conorma triangular, también llamada s-norma o t-conorma, es un operador lógico O u OR que representan un modelo para la unión lógica.

La tabla 3 muestra los diferentes operadores existentes [56].

Tabla 3. S-normas principales

Nombre del operador	Operación
Máximo	$\max(x,y,\dots)$
Suma-Producto	$x + y - x \cdot y$
Suma drástica	$\begin{cases} \max(x, y) & \text{si } \min(x, y) = 0 \\ 1 & \text{en otro caso} \end{cases}$

2.7.2.3. Sistema Difuso Tipo Sugeno

Este método fue introducido por M. Sugeno [58]. Es una propuesta para utilizar funciones $Z_i = f(x,y)$ en vez de conjuntos difusos en el consecuente (fig. 27).

Una regla lingüística en el modelo de Sugeno es:

$$\text{IF } x \text{ es } A \text{ y } y \text{ es } B, \text{ THEN } Z = f(x,y)$$

donde A y B son los conjuntos difusos en el antecedente, mientras que $Z_i = f(x,y)$ puede ser un conjunto de funciones o una función en el consecuente. La función $f(x,y)$ es un polinomio que puede ser de cualquier grado y se encuentra en función de las entradas x y y .

El algoritmo de Sugeno contempla los siguientes pasos, (fig. 27):

1. Fuzzificación
2. Inferencia
3. Agregado
4. Promedio ponderado

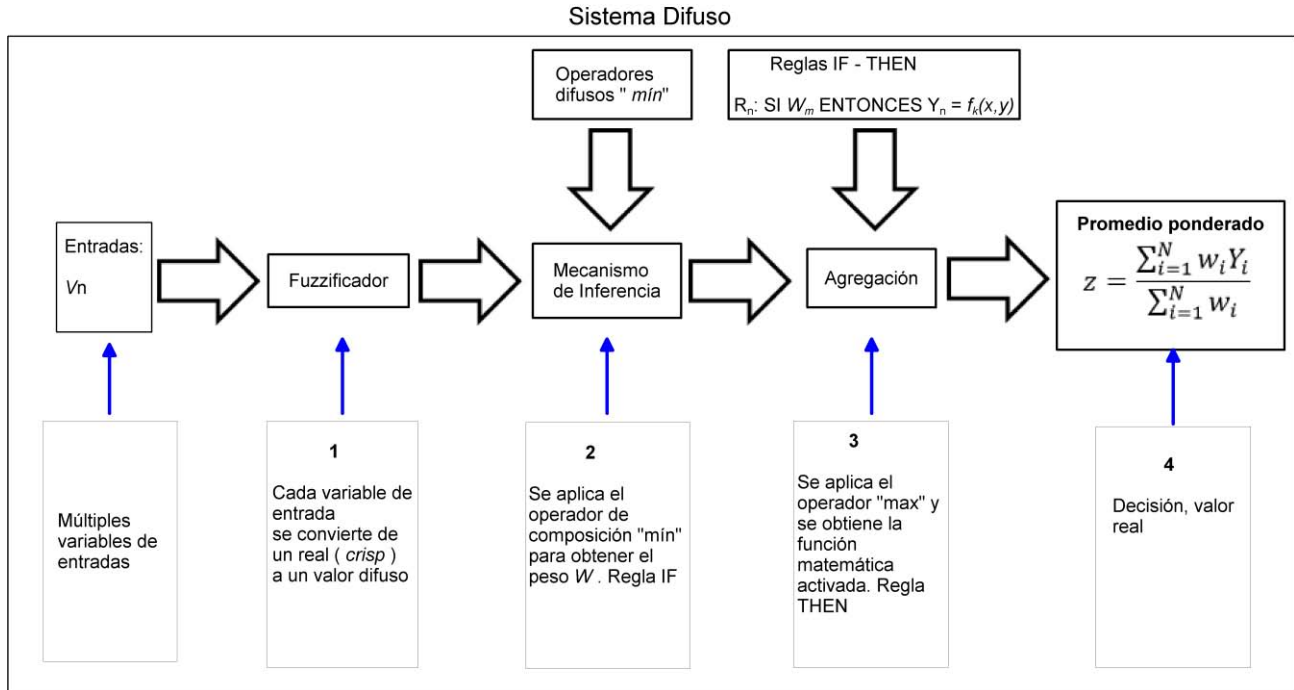


Figura 27. Esquema general del proceso de inferencia de Sugeno

2.8. Algoritmo de inferencia difuso

El algoritmo de inferencia difuso es aplicado para sistemas difusos como el de Mamdani o Sugeno y contempla 4 pasos principales:

Paso 1. Fuzzificación

El proceso de fuzzificación toma tres posibles casos:

-Una variable v_1 entra al sistema difuso y se evalúa en las funciones de membresía. Ésta variable toca a la función A_1 en el punto a y la función A_2 en el punto b , siendo a y b los valores de pertenencia, generando dos valores de pertenencia u_1 y u_2 , donde u_1 tiene un valor entre 0 y 1, y u_2 tiene un valor igual a 0.

-Una variable v_2 entra al sistema difuso y se evalúa en las funciones de membresía. Ésta variable toca a la función A_1 en el punto c y la función A_2 en el punto d , generando dos valores de pertenencia u_1 y u_2 , donde u_1 tiene un valor $u_1 \neq u_2$ que están en el intervalo cerrado $[0,1]$.

-Una variable v_3 entra al sistema difuso y se evalúa en las funciones de membresía. Ésta variable toca a la función de membresía A_2 en el punto e y la función de membresía A_1 en el punto f , generando dos valores de pertenencia u_1 y u_2 , donde u_1 tiene un valor entre 0 y 1, y u_2 tiene un valor igual a 0.

La explicación gráfica se muestra en la figura 28.

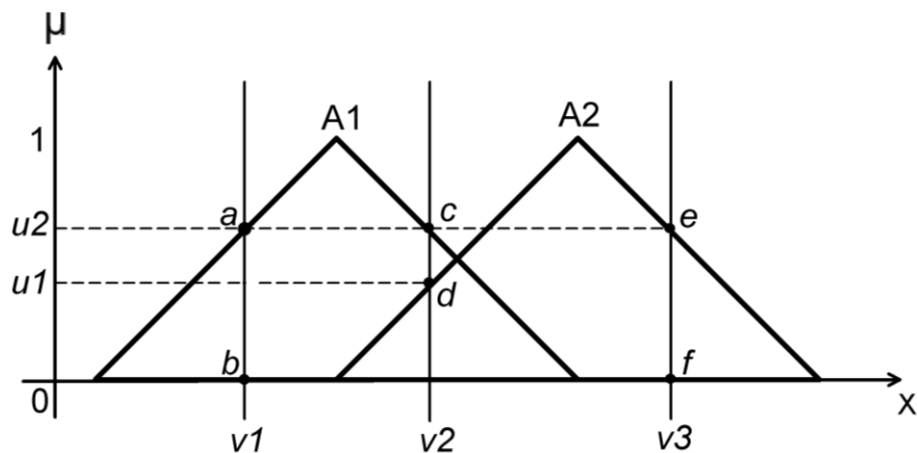


Figura 28. Proceso de fuzzificación para una variable

Paso 2. Inferencia

Después de obtener $u1$ y $u2$, ambos valores se combinan y se evalúan utilizando la operación difusa intersección o mínimo.

La operación mínimo se denota en la ec. (67):

$$\mu_{A \cap B}(x) = \min(u1(x), u2(x)), \quad (67)$$

el valor obtenido después de aplicar la operación difusa mínimo se le conoce como *peso* o *fuerza de activación*, denotada como w ; siendo éste el condicionante o sentencia *IF* necesaria para la aplicación de reglas *IF - THEN*. La realización de todos los pasos previos hasta éste, se les conoce como *antecedente* (fig. 29).

Paso 3. Agregado

Algunos sistemas difusos involucran más de una regla difusa, el proceso de obtener la regla general entre las todas las reglas es llamado agregado. El agregado es la sentencia *THEN*, necesario para completar el condicionante o reglas *IF - THEN*. A los pasos subsecuentes a partir de éste se le conoce como *consecuentes*, (fig. 29).

El procedimiento del agregado para consiste en obtener el máximo entre las intersecciones para cada regla, lo que correspondería a obtener la sentencia *THEN*,

$$\mu_{A \cup B}(x) = \max(\mu_{1_{A \cap B}}, \mu_{2_{A \cap B}}) \quad (68)$$

En el caso de Mamdani (fig. 25) consiste en obtener el máximo de las funciones de membresía de salida que fueron activadas por las reglas lingüísticas, para el caso de

Sugeno se obtiene el máximo de todas las funciones $f(x)$ que se activaron por algunas de las reglas, (fig. 29)

Paso 4. Defuzzificación o Promedio Ponderado.

Para el caso de Mamdani consiste en aplicar algunas de las técnicas de defuzzificación existentes, como el centro de gravedad, centro del área, etc.

Para Sugeno consiste en la aplicación de la ecuación del promedio ponderado, por ponderar se debe entender al proceso de dar un peso o relevancia a algo [59]. Dicho promedio ponderado consiste en la suma de la sustitución de los pesos multiplicados por su correspondiente función $f(x,y)$, dividido entre la suma de los pesos obtenidos, dicha ecuación se denota como [58]:

$$z = \frac{\sum_{i=1}^N w_i f(x, y)_i}{\sum_{i=1}^N w_i} \quad (69)$$

Donde:

w - fuerza de activación o peso, obtenido en el antecedente.

$f(x,y)$ - función matemática activada por el antecedente, obtenida en el consecuente.

La integración de los cuatro pasos anteriores se muestra de forma gráfica en la figura 29.

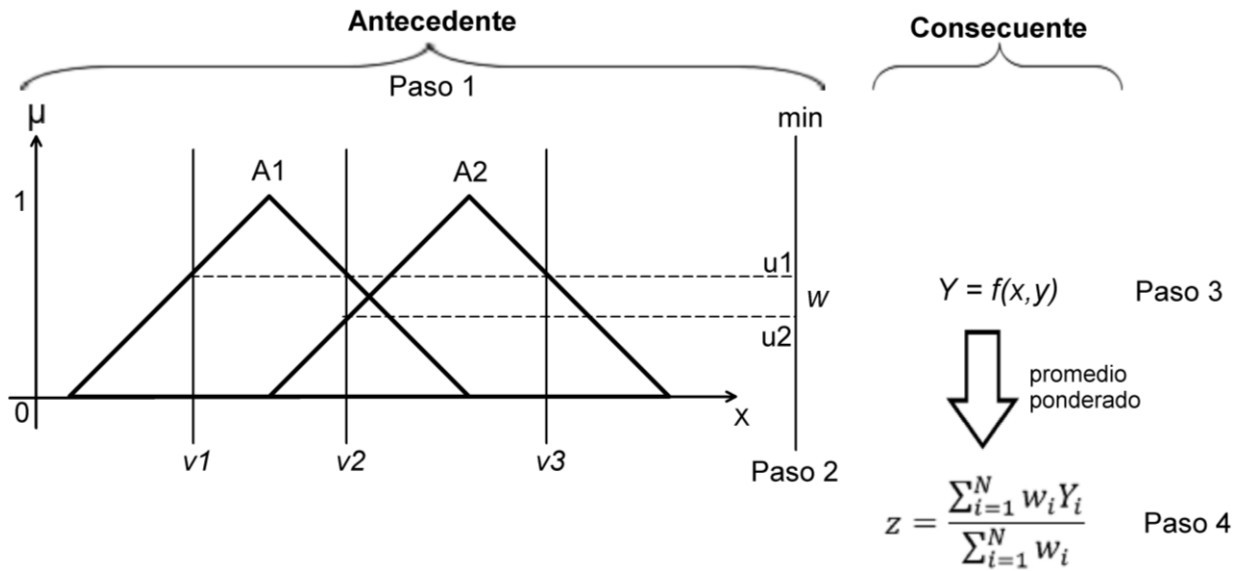


Figura 29. Algoritmo de razonamiento de Sugeno [60]

2.9. Tipos de Sistemas Embebidos

Definición. Un sistema embebido es un módulo electrónico alojado dentro de un sistema de mayor entidad (host o anfitrión) al que ayuda en la realización de tareas tales como el procesamiento de información generada por sensores o el control de determinados actuadores, etc. Dicha información es procesada por un módulo central o CPU (*Central Processing Unit*) que es la encargada de realizar los cálculos necesarios para la salida deseada [61].

Este tipo de sistemas utilizan algún tipo de microprocesador, microcontrolador, FPGA (*Field Programmable Gate Array*) o DSP (*Procesador Digital de Señales*) debido a su flexibilidad que tienen para adaptarse a diferentes tareas, además de ser programados en lenguaje de alto nivel, permitiendo una mayor facilidad para su uso en general.

2.9.1. Hardware en Sistemas Embebidos

Existen diferentes tipos pero en específico pueden ser cualquiera de los siguientes sistemas electrónicos [61]:

-Microprocesador: Es un chip que incluye básicamente la CPU (*Central Processing Unit*) y circuitería relacionadas con los buses de datos y memoria. Para poder realizar su tarea se necesitan de otros chips externos adicionales para incorporar lo que se le conoce como sistema mínimo, tales como memoria, circuitos de entrada salida E/S (I/O) y reloj.

-Microcontrolador de 4, 8, 16 o 32 bits: Es un dispositivo que alberga un sistema mínimo dentro de un único chip, esto es, incluye una CPU, así como buses, reloj, memoria ROM, memoria RAM, puertos de Entrada/Salida (E/S) y otros periféricos tales como convertidores analógicos/digitales. Se les puede encontrar con arquitectura que va desde los 4 hasta 32 bits.

-DSP: Son microcontroladores o microprocesadores diseñados específicamente, tanto en arquitectura hardware como conjunto de instrucciones, para realizar tareas típicas de procesamiento digital de señales en tiempo real.

-Dispositivos FPGA: Es un dispositivo que contiene bloques de arreglos de compuertas cuya interconexión genera tareas y salidas específicas, como la programación de los operadores matemáticos tradicionales suma o multiplicación o la salida de datos en algún tipo de comunicación en específico

Se puede concluir que un sistema embebido está formado por una Unidad de Procesamiento Central (CPU) y software que se ejecuta dentro de él, que contiene las características mínimas para trabajar como memoria RAM, ROM y salidas analógica/digitales para comunicarse con el mundo exterior y recibir señal de diferentes

sensores, diseñados exclusivamente para realizar tareas específicas con el menor tamaño, recursos y tiempo posible.

CAPÍTULO 3. DISEÑO DEL SISTEMA DIFUSO PARA LA LANZADERA.

3.1. Sistema difuso de ángulo de rotación y ángulo de ataque para la lanzadera

Para el diseño de un sistema difuso que permita la rotación y ángulo de ataque de una lanzadera, se definieron dos variables de entrada: 1) dirección de viento, 2) peso del UAV, así como dos de salida: 1) ángulo de rotación, 2) ángulo de ataque. Previo se tienen dos etapas, 1) preprocesado, 2) posprocesado. El preprocesado es el acondicionamiento de la señal de entrada proveniente de los sensores y el posprocesado es el encargado del ajuste de la señal de salida a los actuadores, (fig. 30).

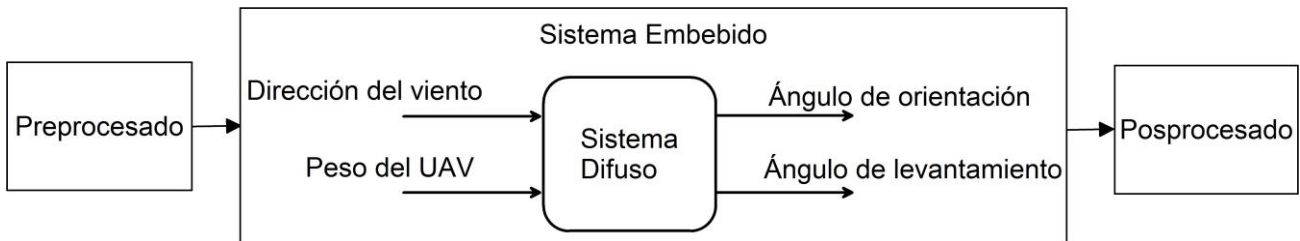


Figura 30. Sistema difuso

3.2. Sistema Embebido

El diseño del sistema embebido se muestra en el diagrama a bloques de la figura 31, el bloque exterior corresponde a la lanzadera, ésta tendrá incluida dentro de sí una tarjeta electrónica modelo *TIVA C LaunchPad*® en donde se encuentra embebido el sistema difuso. El funcionamiento del sistema toma valores del *ADC* donde se encuentran conectados los sensores de dirección del viento y peso, el sistema difuso los procesa y obtiene el ángulo correcto, así como la acción para la activación del motor y sentido de giro.

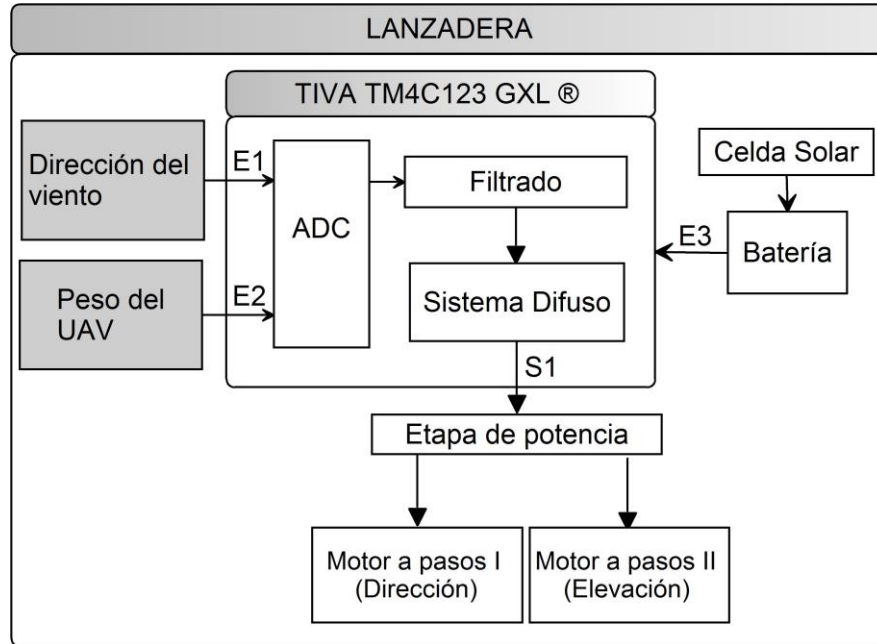


Figura 31. Diagrama a bloques del sistema embebido

El sistema embebido está constituido por dos sensores que proporcionan la dirección del viento y el peso del UAV, una tarjeta electrónica modelo *TIVA C LAUNCHPAD*, en donde se encuentra embebido el sistema difuso, dos *drivers* controladores para motores y dos motores a pasos que posicionan la lanzadera en los respectivos ángulos (fig. 32).

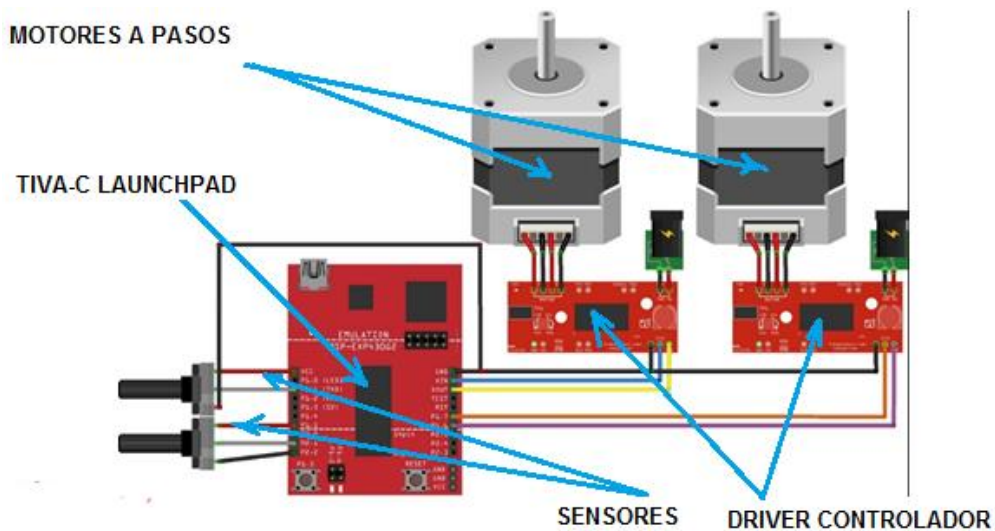


Figura 32. Elementos que conforman el sistema

3.2. Subsistemas Embebidos

El sistema difuso embebido está conformado por tres parte: 1) Entrada, donde se encuentran la señales recibidas por los sensores, 2) Sistema, donde se encuentran embebido los sistemas difusos, 3) Salida, donde se encuentra la señal que será enviada a los dos motores a pasos, así como por dos subsistemas difusos: 1) corresponde al motor de rotación que tiene como entrada la dirección del viento y sus variables se definen como el e - error y de - cambio en el error, 2) corresponde al motor de ángulo de ataque, dado por un motor a pasos que tiene como entrada el peso del UAV, (fig. 33).

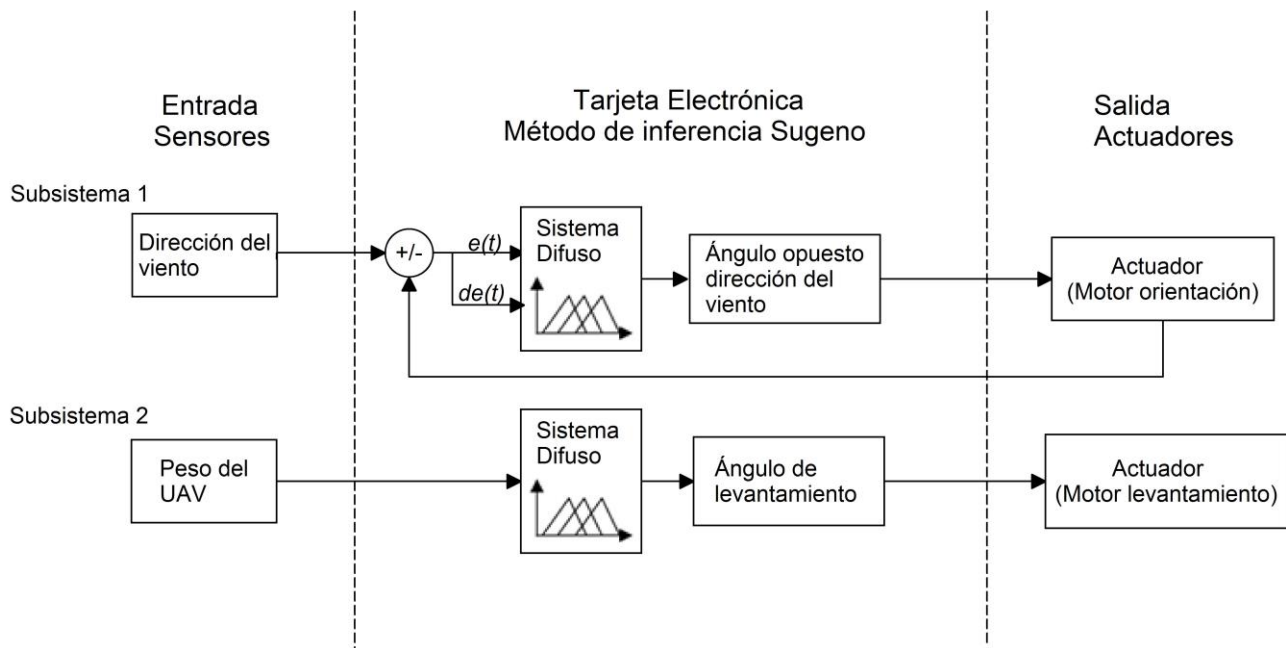


Figura 33. Diagrama a bloques de los sistemas difusos

3.2.1. Subsistema 1

En el diagrama a bloques de la figura 34 se muestra el sistema embebido utilizado para el subsistema 1 que corresponde a la entrada de la dirección del viento.

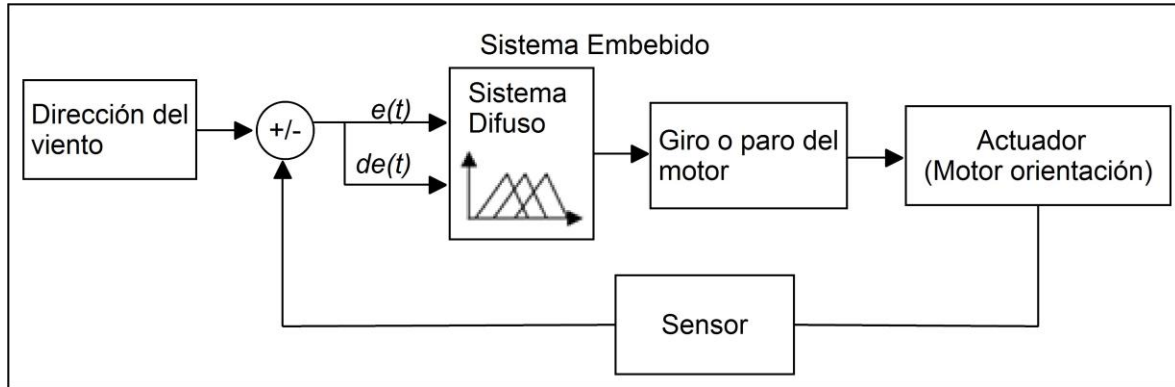


Figura 34. Sistema difuso para el motor de la base

donde:

$$e(t) = \text{entrada del sensor} - \text{ref. del motor} \quad (\text{error}) \quad (70)$$

$$de(t) = e(t-1) - e(t) \quad (\text{cambio en el error}) \quad (71)$$

$e(t)$ es la parte proporcional o el error instantáneo de posición entre el sensor y la referencia deseada del motor.

$de(t)$ es el cambio en el error o también conocido como parte diferencial, el cual mide las variaciones instantáneas del error o una predicción de los errores futuros. Su función es complementar las acciones a futuro de la parte proporcional, que por la naturaleza instantánea de ésta variable no es capaz de realizar.

3.2.1.1. Funcionamiento $e(t)$ y $de(t)$

El objetivo del sistema es orientar manteniendo la magnitud del error en un valor cercano a cero. La señal de entrada se resta con la de retroalimentación del motor para generar el error. Esta señal se introduce al sistema difuso, así como el cambio en el error. La salida

del sistema difuso son dos valores binarios para la activación y sentido de giro del motor, los cuales entran a un *driver* controlador de motores a pasos.

Para cerrar el lazo de control se toma como señal de retroalimentación una variable interna programada dentro de la tarjeta electrónica que guarda el número de pasos convertido a ángulo que ha girado el motor.

Si se considera solamente el error, se puede diseñar lo que se conoce como un sistema tipo *P*, por otro lado, un sistema tipo *PD* tiene como segunda variable la diferencia o el cambio del error con respecto del tiempo.

La señal del error (*e*) es necesario describirla con valores positivos o negativos, los valores del error negativos significan que en el proceso de control la salida tiene un valor menor que la entrada, es decir, $e(t) = \text{entrada} - \text{ref} < 0$. Por el otro lado cuando tenemos que un valor de error es positivo, significa que la señal de error del proceso actual se encuentra por arriba de la referencia. Hablando del cambio en el error (*de*), un valor negativo significa que en el proceso actual la salida se ha incrementado en comparación a su valor anterior, es decir, $de = e(t) - e(t-1) < 0$. Un valor positivo del cambio en el error significa que la salida ha decrementado su valor comparado con su valor anterior, (fig. 35).

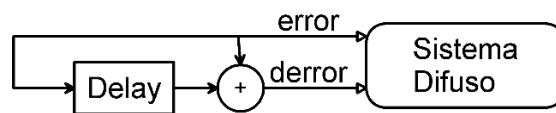


Figura 35. Representación en diagrama a bloques de un sistema difuso tipo *PD*

La primer variable de entrada requerida por el controlador es el error, el cual corresponde solamente a la parte proporcionar *P*, que se encarga de girar el motor para alcanzar la referencia entrante o el cero, la siguiente variable de entrada requerida por el controlador es el cambio en el error, que corresponde a la parte derivativo *D*, ésta variable mide los cambios bruscos del error o la predicción a futuro, tiene la principal finalidad de corregir la

inercia del motor debido a la carga de la lanzadera y como un apoyo para disminuir el sobrepaso máximo que pudiera haber en el proceso de sintonización y estabilización de la señal de retroalimentación del motor con la parte proporcional.

3.2.2. Conjuntos difusos del subsistema 1

3.2.2.1. Conjunto difuso $e(t)$

En la figura 36 muestra las funciones de membresía o conjuntos difusos para la entrada *error*, se muestra el intervalo $[-10, 10]$. El universo de discurso va desde $[-360, +360]$, que representa un giro completo de la base de lanzadera. Los valores lingüísticos utilizados son, “error-negativo”, “error-cero” y “error positivo”, donde error-cero es una función del tipo triangular sintonizada en $[-3, +3]$, estos valores de sintonización se deben a varias razones, una de las razones es para evitar ruido, si se sintoniza en un intervalo muy cercado al 0 el valor de paro puede que oscile a su entrada y eso ocasionará una salida ruidosa que pudiera en algún pico activar la salida, otra razón es por si existe algún cambio brusco con respecto de 0 hasta ± 3 , es decir, entre 0 y ± 3 se permite cambios bruscos u oscilaciones pero al salirse de ese rango se activa la salida. Las funciones de activación se utilizaron las variables lingüísticas, “error-negativo” y “error positivo”, ambas son funciones trapezoidales con máximo en +4 y -4 y cierre en +360 y -360 respectivamente.

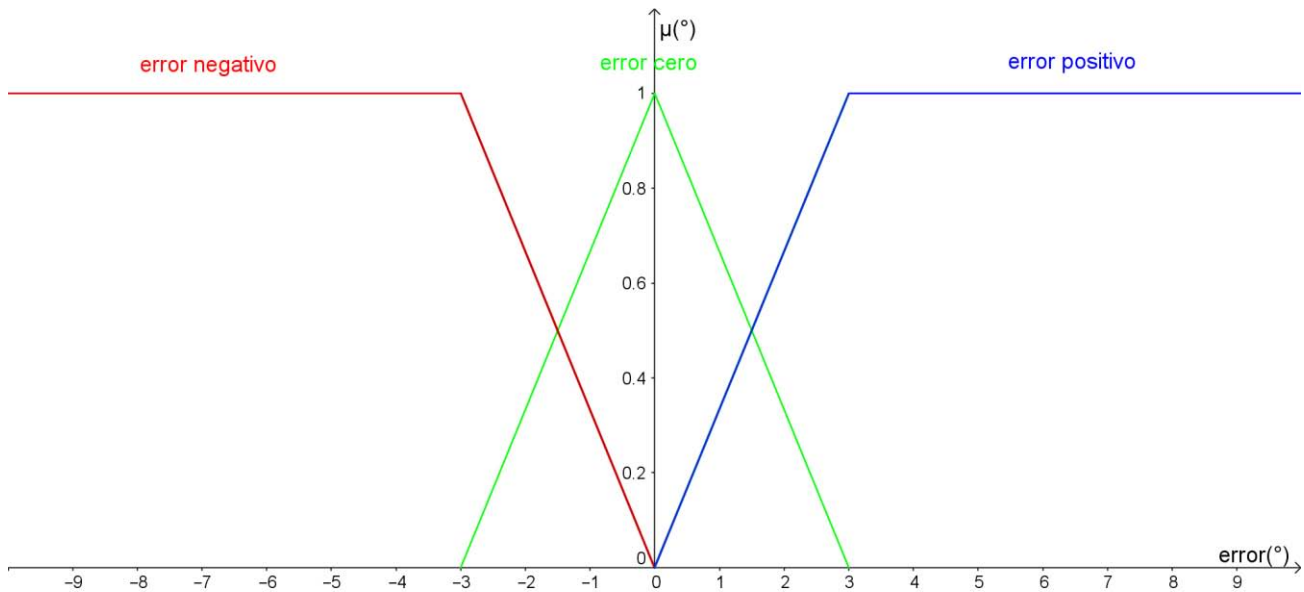


Figura 36. Universo de discurso para la variable lingüística *error* (°)

La función de membresía *error negativo* se encuentra definido de $[-360, 0]$, y está integrada por dos funciones, una recta con pendiente $m = 0$, que va de $[-360, -3]$ y otra recta con pendiente negativa $m = -\frac{1}{3}$ que va de $[-3, 0]$, (fig. 36).

Se define como la ec. (72):

$$error\ negativo(error) = \begin{cases} 1, & si\ -3 \leq error \\ \frac{0-x}{0-(-3)}, & si\ -3 < error \leq 0 \\ 0, & si\ error > 0 \end{cases} \quad (72)$$

La función de membresía *error cero* es una función de tipo triangular, y está integrada por dos funciones, una recta con pendiente positiva $m = \frac{1}{3}$ que va de $[-3, 0]$ y otra recta con pendiente negativa $m = -\frac{1}{3}$ que va de $[0, +3]$.

Se define como la ec. (73):

$$erro\ zero(error) = \begin{cases} 0, & si\ -3 \leq error \\ \frac{x-(-3)}{0-(-3)}, & si\ -3 \leq error \leq 0 \\ \frac{3-x}{3-0}, & si\ 0 \leq error \leq 3 \\ 0, & si\ error \geq 3 \end{cases} \quad (73)$$

La función de membresía error positivo está integrada por dos funciones, una recta con pendiente positiva $m = \frac{1}{3}$ que va de $[0, +3]$ y otra recta con pendiente $m = 0$ que va de $[3, +360]$, (fig. 36).

Se define como la ec. (76):

$$error\ positivo(error) = \begin{cases} 0, & si\ error \leq 0 \\ \frac{x-0}{3-0}, & si\ -3 < error \leq 0 \\ 1, & si\ error > 3 \end{cases} \quad (74)$$

3.2.2.2. Conjunto difuso de(t)

La siguiente variable de entrada requerida por el sistema difuso es el cambio en el error, ésta variable mide los cambios bruscos del error o su predicción a futuro, además de evitar posibles picos de ruido que pudieran disparar alguna regla equivocadamente, para corregir la inercia del motor debido a la carga de la lanzadera y como un apoyo para disminuir el sobrepaso máximo que pudiera haber en el proceso de sintonización y estabilización de la señal de retroalimentación del motor con la parte proporcional.

Las funciones de membresía se muestran de forma gráfica en la figura 37.

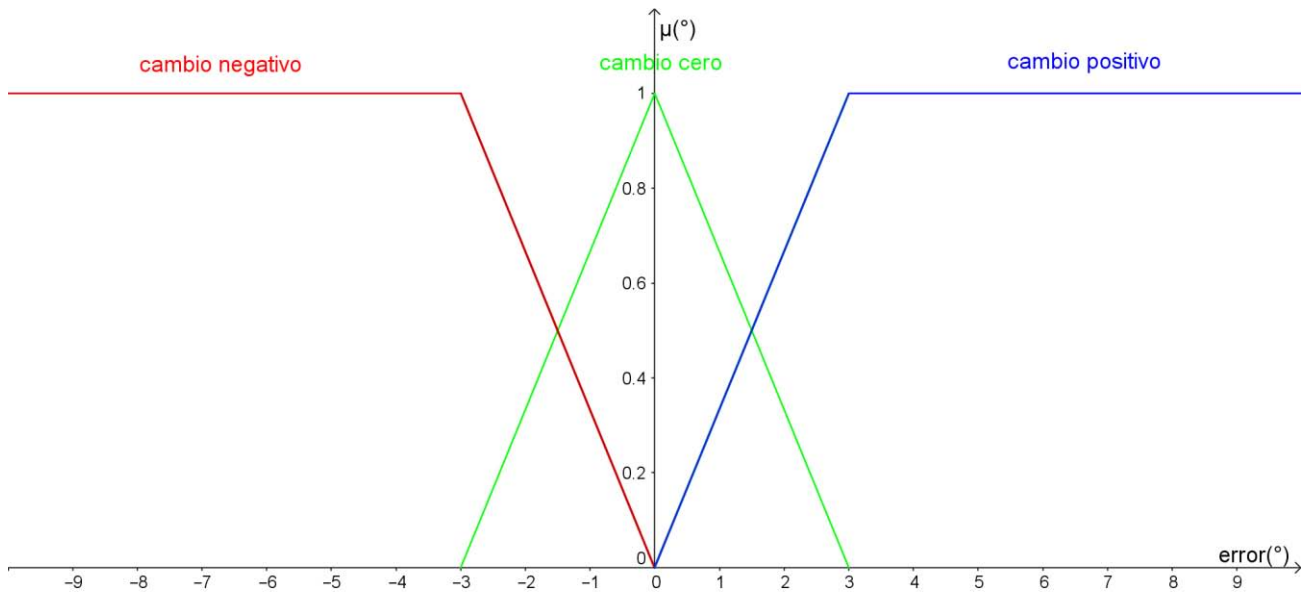


Figura 37. Universo de discurso para la variable lingüística “cambio en el error”

La función de membresía cambio negativo se encuentra definido de $[-360, 0]$, y está integrada por dos funciones, una recta con pendiente $m = 0$ que va de $[-360, -3]$ y otra recta con pendiente negativa $m = -\frac{1}{3}$ que va de $[-3, 0]$, (fig. 37).

Se define como la ec. (75):

$$cambio\ negativo(error) = \begin{cases} 1, & si\ -3 \leq error \\ \frac{0-x}{0-(-3)}, & si\ -3 < error \leq 0 \\ 0, & si\ error > 0 \end{cases} \quad (75)$$

La función de membresía cambio cero, se encuentra definido de $[-3, +3]$, y está integrada por dos recta, una recta con pendiente positiva $m = \frac{1}{3}$ de $[-3, 0]$ y la siguiente con pendiente negativa $m = -\frac{1}{3}$ de $[0, +3]$, (fig. 37).

Se define como la ec. (76):

$$cambio\ cero(error) = \begin{cases} 0, & si\ -3 \leq error \\ \frac{x-(-3)}{0-(-3)}, & si\ -3 \leq error \leq 0 \\ \frac{3-x}{3-0}, & si\ 0 \leq error \leq 3 \\ 0, & si\ error \geq 3 \end{cases} \quad (76)$$

La función de membresía cambio positivo se encuentra definido de $[0, 360]$ y está integrada por dos funciones, una recta con pendiente positiva $m = \frac{1}{3}$ que va de $[0, +3]$ y otra recta con pendiente $m = 0$ que va de $[+3, +360]$, (fig. 37).

Se define como la ec. (77):

$$cambio\ positivo(error) = \begin{cases} 0, & si\ error \leq 0 \\ \frac{x-0}{3-0}, & si\ -3 < error \leq 0 \\ 1, & si\ error > 0 \end{cases} \quad (77)$$

3.2.3. Reglas difusas.

El diseño de las reglas *IF - THEN* es el que corresponde al bloque de base del conocimiento (fig. 25). Estas son el conocimiento empírico obtenido por un experto en nuestro caso para realizar un proceso de orientación de la lanzadera.

La tabla 4 muestran las reglas. Éstas se encuentra separa en tres secciones. 1) La 1ra sección es el error negativo o cuando la retroalimentación se encuentra por arriba de la referencia, la salida correspondiente debido a este error es la activación del motor y dirección en sentido contrario. La siguiente sección corresponde al error en cero o cuando la referencia es igual a la retroalimentación, la salida correspondiente es activación cero o dicho de otra manera “paro del motor”, al estar activada esta salida la siguiente variable de

salida no importa su valor ya que el motor se encuentra en paro total. La última sección de la tabla corresponde al error positivo o cuando la retroalimentación se encuentra por debajo de la referencia, la salida debido a esto es la activación del motor con dirección hacia el lado contrario correspondiente.

Tabla 4. Reglas para la activación del sistema difuso

Error & Cambio-Error
<p>1. IF: (error es neg & cambio-error es neg) THEN: (activación es uno & dirección es cero)</p> <p>2. IF: (error es neg & cambio-error es cero) THEN: (activación es uno & dirección es cero)</p> <p>3. IF: (error es neg & cambio-error es pos) THEN: (activación es uno & dirección es cero)</p>
<p>4. IF: (error es cero & cambio-error es neg) THEN: (activación es cero & dirección es cero)</p> <p>5. IF: (error es cero & cambio-error es cero) THEN: (activación es cero & dirección es cero)</p> <p>6. IF: (error es cero & cambio-error es pos) THEN: (activación es cero & dirección es cero)</p>
<p>7. IF: (error es pos & cambio-error es neg) THEN: (activación es uno & dirección es uno)</p> <p>8. IF: (error es pos & cambio-error es cero) THEN: (activación es uno & dirección es uno)</p> <p>9. IF: (error es pos & cambio-error es pos) THEN: (activación es uno & dirección es uno)</p>

3.2.4. Defuzzificación del subsistema 1

Se definieron dos funciones constantes de defuzzificación para la salida, la tabla 5 resume las salidas para cada variable, de donde las funciones $f(x)$ se obtuvieron de cambiar las salidas digitales por funciones constantes, (fig. 38).

Tabla 5. Activación del driver para el motor a pasos

Entradas del Driver	
Activación (Salida digital)	Dirección (Salida digital)
1 – Activa giro del motor	1 – Giro un horario
0 – No activa el motor	0 – Giro antihorario

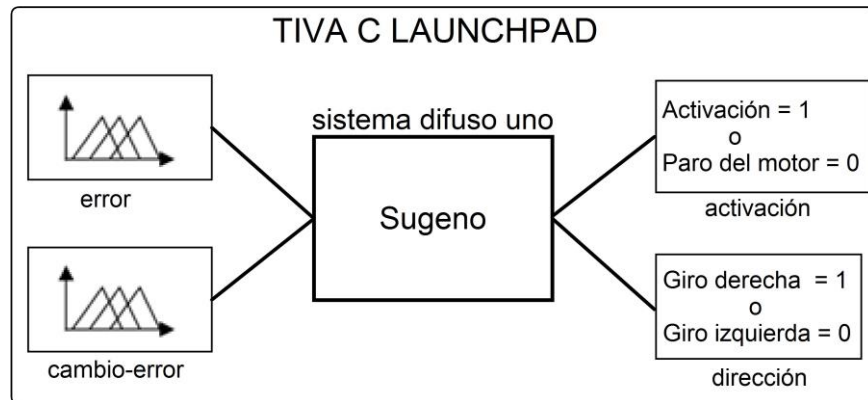


Figura 38. Diagrama a bloques del sistema difuso uno

3.3. Subsistema 2

Para el diseño del sistema de ángulo de ataque para el segundo motor se utilizó igualmente el método de inferencia de *Sugeno* y como variable de entrada el peso del UAV.

El diagrama a bloques de la figura 39 muestra el esquema del sistema difuso utilizado.

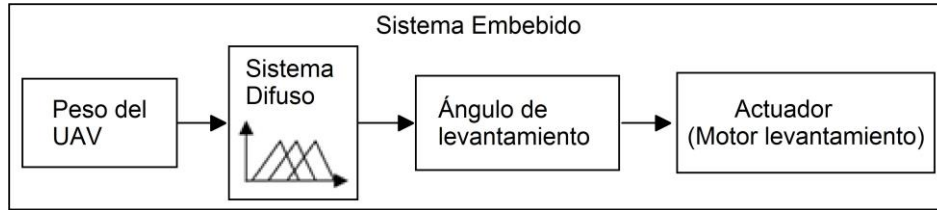


Figura 39. Sistema difuso para el motor de ángulo de ataque

El sistema requiere como entrada el peso del UAV. A la salida se tiene el ángulo al cual tiene que girar el motor el cual se manda al actuador correspondiente.

3.3.1. Funciones de Membresía

Las funciones de membresía se muestran en la figura 40. Está compuesta por tres funciones con tres variables lingüísticas, poco peso, peso medio y peso máximo.

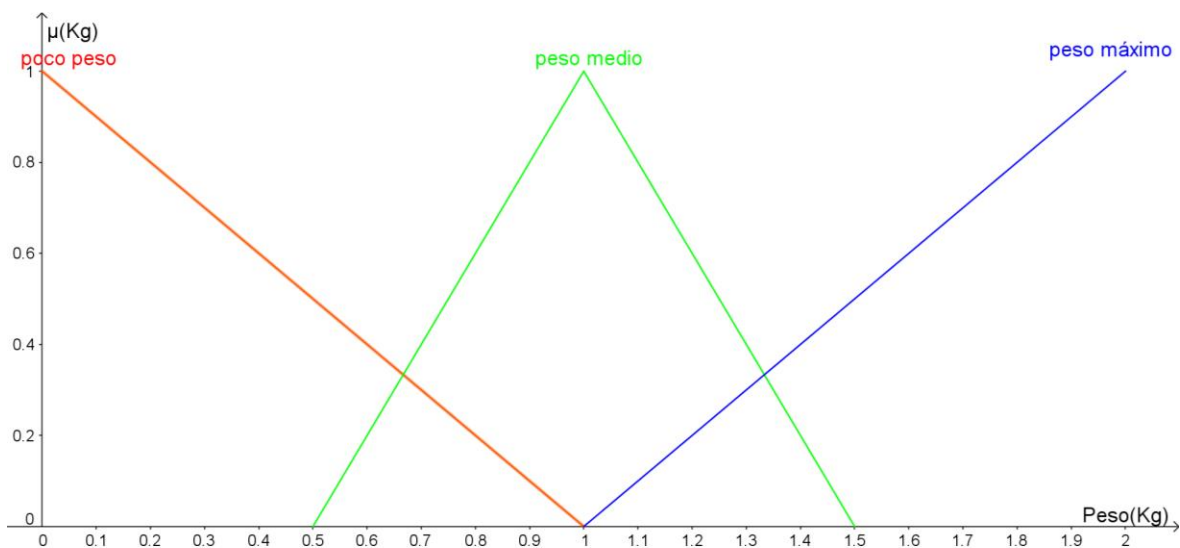


Figura 40. Universo de discurso para la variable de entrada "peso del UAV"

La función de membresía poco peso se encuentra definido de $[0, +1]$, y está integrada por una función con pendiente negativa $m = -1$, (fig. 40).

Se define como la ec. (78):

$$poco\ peso\ (peso) = \begin{cases} 0 & si\ 0 \leq error \\ \frac{1-x}{1} & si\ 0 < error \leq 1 \\ 0 & si\ error > 1 \end{cases} \quad (78)$$

La función de membresía peso medio, se encuentra definido de $[+0.5, +1.5]$, y está integrada por dos rectas, la primer con pendiente positiva $m = +1$ que va de $[+0.5, +1]$ y la siguiente con pendiente negativa $m = -1$ que va de $[+1, +1.5]$, (fig. 40).

Se define como la ec. (79):

$$cambio\ cero\ (error) = \begin{cases} 0, & si\ 0.5 \leq error \\ \frac{x-0.5}{1-1.5}, & si\ 0.5 \leq error \leq 1 \\ \frac{1.5-x}{1.5-1}, & si\ 1 \leq error \leq 1.5 \\ 0, & si\ error \geq 1.5 \end{cases} \quad (79)$$

La función de membresía peso máximo se encuentra definido de $[+1, +2]$, y está integrada por una función con pendiente positiva $m = +1$, (fig. 40).

Se define como la ec. (80):

$$cambio\ cero(error) = \begin{cases} 0, & si\ 1 \leq error \\ \frac{x-1}{1}, & si\ 1 < error \leq 2 \\ 0, & si\ error > 2 \end{cases} \quad (80)$$

3.3.1. Defuzzificación del Subsistema 2

La función para la defuzzificación se representa a través de una recta $f(x) = 7.5x$. Ésta función se obtuvo de forma teórica a partir de definir el ángulo de ataque de un micro-uav en específico, que para este caso se utilizó el modelo micro-Cessna 172, con un peso de 2 kg y un ángulo de ataque de 15°, obtenido a partir de su tipo de perfil alar el NACA 0012, (fig. 41).

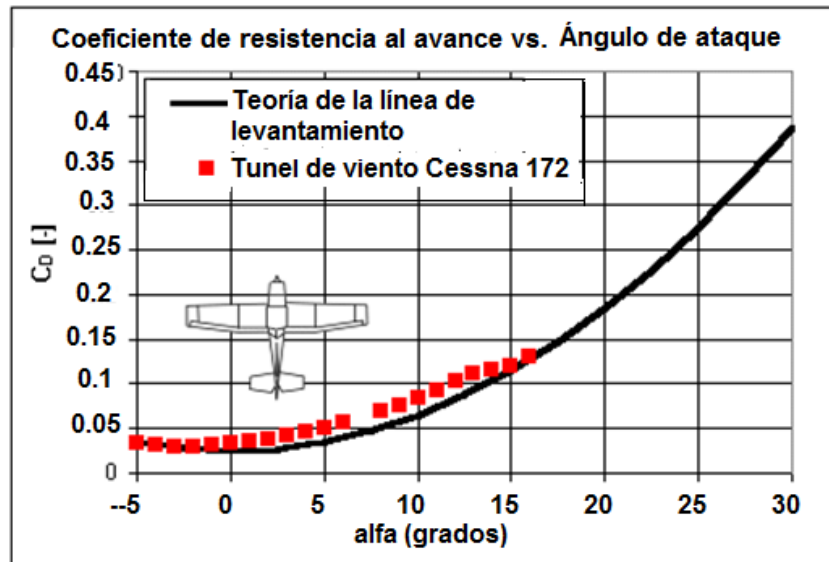


Figura 41. Coeficiente de arrastre vs. ángulo de ataque del micro-UAV Cessna 172 [62]

La función de defuzzificación se obtuvo a partir del peso del UAV, que es igual a 2 kg, y además con su ángulo de ataque, igual a 15°, con estos dos valores se puede obtener un punto en un plano, y para un peso de 0 kg corresponde un ángulo de 0°, obteniendo un segundo punto en el plano, con los cuales al aplicar la ecuación de la recta entre dos puntos, ec. (80), se obtiene la función $f(x)$ que la define, (fig. 42).

$$y - y_0 = \frac{y_1 - y_0}{x_1 - x_0} (x - x_0) \quad (81)$$

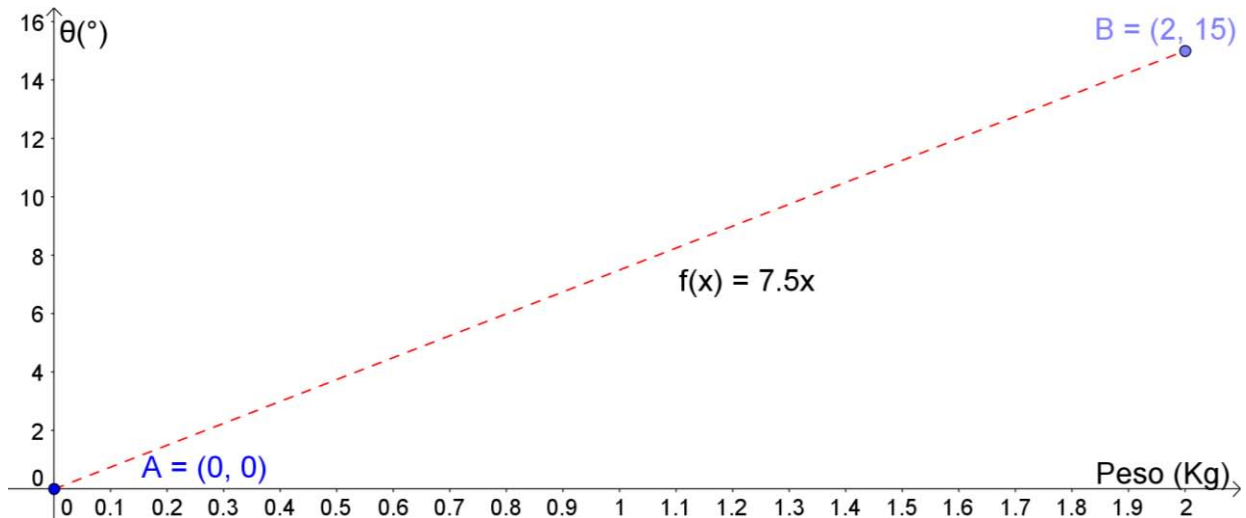


Figura 42. Función para la defuzzificación, recta $f(x) = 7.5x$

3.3.1. Diseño de las reglas difusas.

El diseño de las reglas *IF-THEN* es la parte más importante del sistema difuso, ya que corresponde al bloque de “base del conocimiento”. Estas reglas son el conocimiento empírico obtenido por un experto para el levantamiento de la lanzadera.

La tabla 6 muestra las reglas para la activación del motor de ángulo de ataque. Son tres reglas correspondientes a las tres variables lingüísticas asociadas a los tres conjuntos difusos, y consiste en ser evaluado el valor de entrada en la función $f(x)$.

Tabla 6. Reglas para la activación del sistema difuso para el motor de ángulo de ataque

Peso
1. IF: Poco peso, THEN: $f(x) = 7.5x$
2. IF: Peso medio, THEN: $f(x) = 7.5x$
3. IF: Peso máximo, THEN: $f(x) = 7.5x$

CAPÍTULO 4. ELEMENTOS DEL SISTEMA

El sistema se muestra en la figura 43, está constituido por dos sensores que proporcionan la dirección del viento y el peso del UAV, una tarjeta electrónica modelo *TIVA C LAUNCHPAD*, en donde se encuentra embebido el sistema difuso, dos *drivers* controladores de motores a pasos para igual cantidad de motores a pasos que posicionan la lanzadera en los respectivos ángulos.

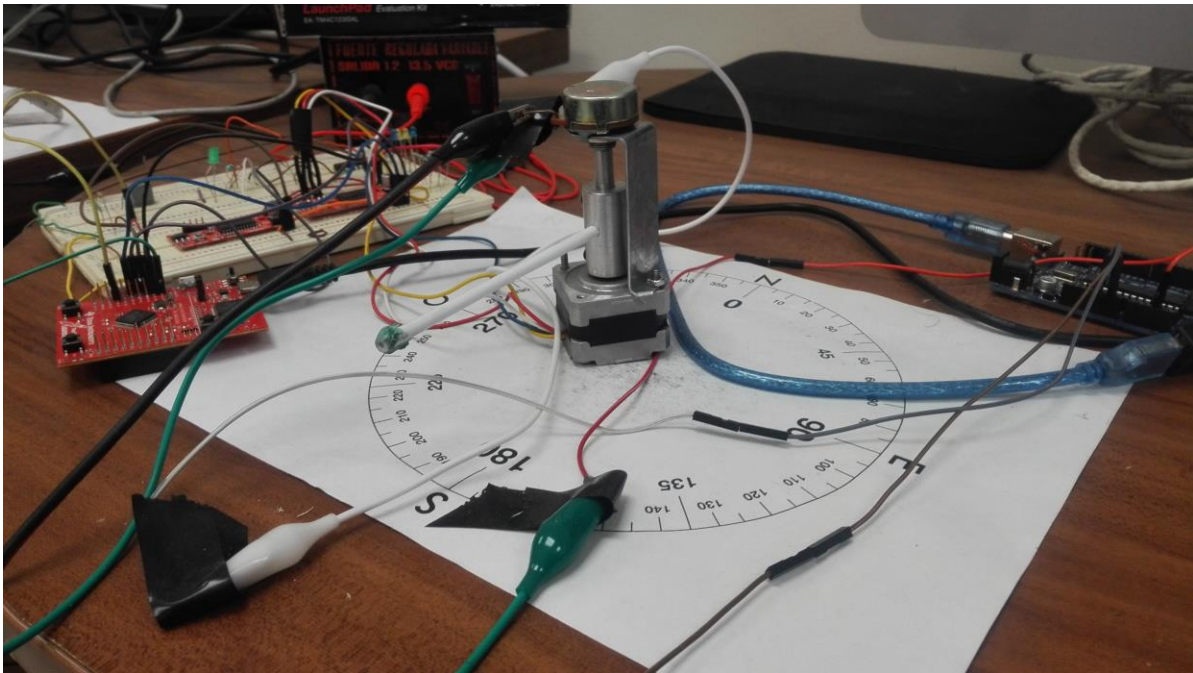


Figura 43. Instrumentación Electrónica (vista superior)

4.1. Tarjeta electrónica

Como microcontrolador para embeber el sistema difuso se utilizó la tarjeta electrónica *TIVA-C Launch Pad* de la empresa *Texas Instrument©*, (fig. 44).

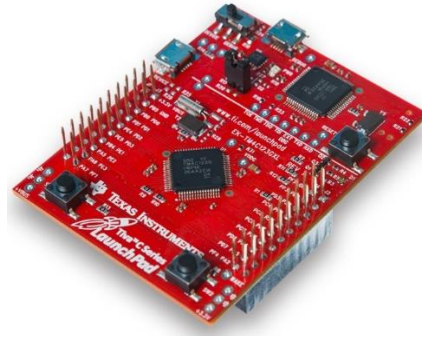


Figura 44. TIVA TM4C123GXL [63] [Anexo A].

4.2. Sensores

4.2.1. Sensor de dirección de viento

Integra tanto elementos analógicos como digitales para la medición del viento, éste gira al potenciómetro y cambia su valor de resistencia que registra datos que son enviados para su posterior procesamiento.

Se propuso el potenciómetro lineal modelo COM-09939 de la empresa *SparkFun*, (fig. 45), (ANEXO B).



Figura 45. Potenciómetro lineal modelo COM-09939.

4.2.1.1. Funcionamiento del sensor de dirección de viento

-Señal de salida: La señal de salida del sensor es una variación de voltaje a través de un potenciómetro de 10 K Ω . La forma de alimentación es con una fuente de 6 VDC aplicado en

la terminal de excitación, y la salida es sensada en la terminal de voltaje. 0V indica que se encuentra en el Norte y 3V al Sur, a partir de ahí la tensión se reparte hasta los 360°, (tabla 7).

Tabla 7. Valores de salida analógica del potenciómetro

Voltaje (V)	Grados (°)	Dirección
0	0	N
1.5	90	Este
3V	180	S
4.5	270	OE

-Compensación de la banda muerta: Para medir un giro completo del potenciómetro, existe una región llamada banda muerta que está constituida por un conjunto de valores de resistencia que oscila entre $\pm 1\Omega$ que equivale a $\approx 0V$, así al tener un cruce por cero se sabe que ha dado una vuelta completa, (fig. 46).

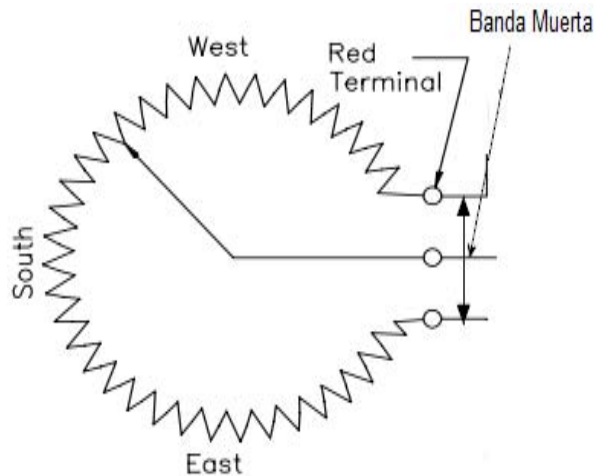


Figura 46. Banda muerta

4.2.2. Sensor de peso

Los sensores de peso o celdas de carga, son aquellos dispositivos electrónicos desarrollados con la finalidad de medir cambios eléctricos provocados por una variante en la magnitud de un peso aplicado sobre el sensor.

Se utilizó el sensor de peso lineal modelo COM-09939 ROHS, (Anexo C), (fig. 47).

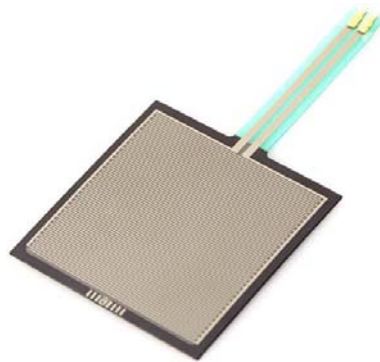


Figura 47. Sensor de peso modelo SEN-09376 ROHS

4.3. Filtro de Media Móvil.

Los sensores utilizados analógicos, tienen la ventaja de ser muy precisos y mejores en ciertos aspectos que los digitales, pero su gran desventaja es la capacidad que tiene de adquirir ruido blanco (ambiental), para evitar éste problema es necesario utilizar algún tipo de filtro para su reducción, por lo que se propone el filtro de media móvil.

“El filtro de media móvil, es el filtro más común utilizado en aplicaciones embebidas, ya que puede ser programado en cualquier dispositivo electrónico como DSP, FPGA o microcontrolador, principalmente porque es el más fácil de entender y usar” [64]. A pesar de su simplicidad, este filtro es el óptimo para una tarea en específico: reducir el ruido aleatorio al mismo tiempo que conserva la respuesta de la señal [65] [66].

La ecuación del filtro se muestra en la ecuación 82.

$$y[n] = \frac{1}{M} \sum_{i=0}^{M-1} x[n - 1] \quad (82)$$

Donde:

- $x[n-1]$ es la señal de entrada

- $y[n]$ es la señal de salida

- M es el número de muestras a filtrar.

La razón por la que existe un corrimiento en $x[n-1]$ se debe a que la primer muestra se deduce que es cero.

“Muchos científicos e ingenieros sienten culpa acerca de usar el filtro de media móvil, porque es muy simple, incluso si el problema está completamente resuelto, sigue existiendo un sentimiento de que algo más se debería hacer; esta situación es completamente irónica. No solamente éste filtro es muy bueno en muchas aplicaciones, es el óptimo para un problema en común, reducir el ruido aleatorio blanco mientras se mantiene la señal” [64].

La utilización de un filtro de media móvil consiste en aplicar a la señal de entrada la ec. (82). Para éste trabajo de tesis se aplicaron dos filtros de media móvil en cascada tomando 20 muestras cada uno.

4.4. Actuadores

En el movimiento mecánico del sistema se utilizaron motores a pasos modelo 5014-820 del fabricante *Applied Motion* (Anexo D), el modelo es de tamaño NEMA 14 con 4 cables de salida y 1.8° grados por paso, 3.2 V a 0.35 A, 8.5Ω y 150 gr. (fig. 48)



Figure 48. Motor a pasos

4.5. Etapa de potencia y *Driver* Controlador.

La etapa de potencia es el proceso que consiste en aumentar el nivel de una señal de entrada, incrementando para ello la amplitud mediante corrientes de polarización (voltaje negativo, voltaje positivo) utilizando un transistor o conjunto de transistores. Cuando se habla de motores, la etapa de potencia sirve para acoplar las salidas *TTL* de un microcontrolador a los voltajes y corrientes que utilizan los motores; normalmente los microcontroladores manejan corrientes $\approx 100 \text{ mA}$, menores a las requeridas por un motor, debido a esto se utiliza la etapa de potencia.

Para realizar la secuencia de accionamiento de un motor a pasos y aumentar la corriente de salida se utilizó un *driver* controlador, éste consta de una tarjeta electrónica que cuenta con un conjunto de transistores internos en configuración *Darlington-serie* que funciona como etapa de potencia y un microcontrolador que se encargan de realizar los ciclos de conmutado de las bobinas del motor para realizar el giro, (fig. 49).



Figura 49. *Driver* controlador de motores a pasos modelos *EasyDriver*© de la empresa *SparkFun*® [67]

La conexión de un *driver* se muestra en la figura 50.

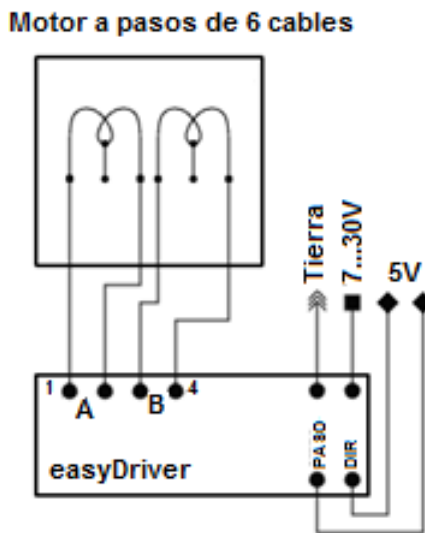


Figura 50. Conexiones para el *EasyDriver* [68]

El *driver* funciona a partir de dos pines de entrada para su control, entrada de activación y dirección, la entrada activación enciende el motor al recibir un 1 lógico y se detiene hasta recibir un 0, el pin de dirección proporciona el sentido del giro del motor, al recibir 1 gira en sentido horario y 0 en antihorario, (tabla 8).

Tabla 8. Activación del driver para el motor a pasos

Entradas del Driver	
Pin Activación (Salida digital)	Pin Dirección (Salida digital)
1 – Activa giro del motor	1 – Giro un horario
0 – No activa el motor	0 – Giro antihorario

La figura 51 muestra la instrumentación del sistema embebido en forma de diagrama eléctrico.

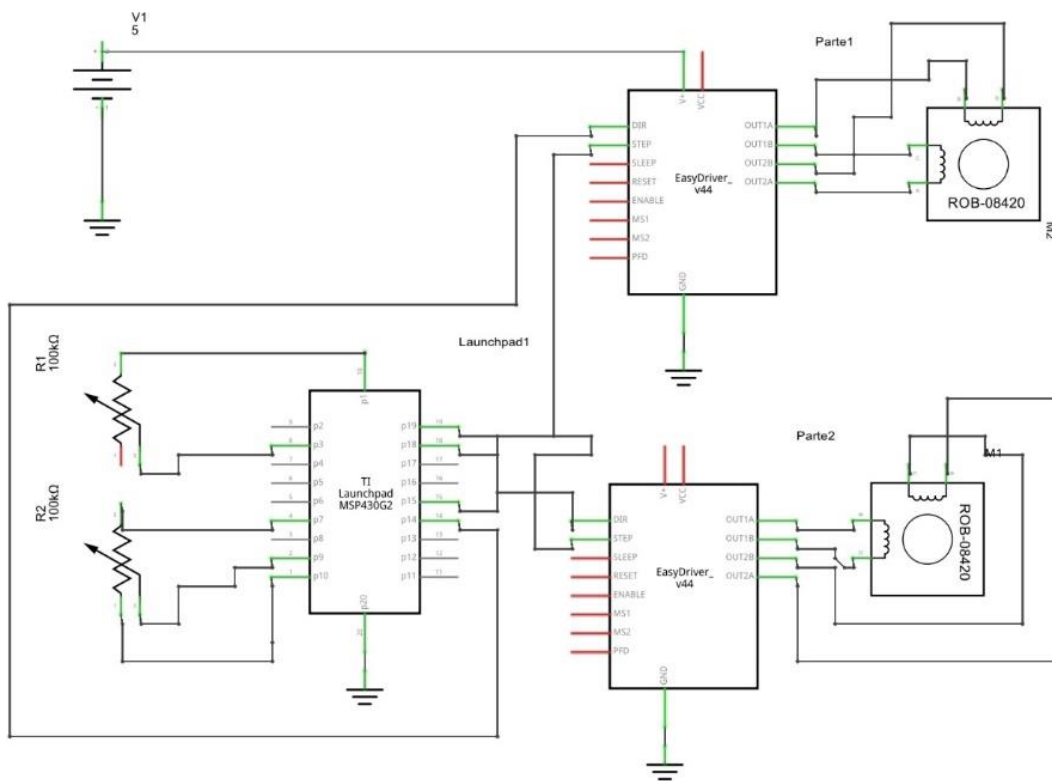


Figura 51. Sistema Instrumental

CAPÍTULO 5. RESULTADOS

5.1. Comprobación de las Reglas Sistema 1.

Para comprobar que las reglas son correctas se utilizó la herramienta interna del *Toolbox*TM de *Fuzzy Logic*© de *Matlab*®, además de un código en *Dev C* que permitirá comprobar los resultados del *Toolbox* con la programación en C. Éste mismo código en C fue embebido dentro de la tarjeta electrónica.

En la tabla 9 se muestran los resultados de la comparación entre el *Toolbox*TM y el código en C. Las columnas de “Entrada” muestran valores en grados de entradas al sistema difuso como prueba para su validación, las columnas de salida son valores binarios y representan la respuesta obtenida por el sistema difuso después de aplicar los valores de entrada. La columna de error muestra la comparación en porcentaje de los resultados obtenidos entre las columnas que abarcan el *Toolbox*TM y el lenguaje C (ver códigos en Anexo C).

Tabla 9. Comparación de resultados entre el *Toolbox* de *Matlab* y el código en lenguaje C

COMPARACIÓN DE RESULTADOS										
NO. REGLA	TOOLBOX				LENGUAJE C				ERROR	
	Entrada (°)		Salidas (b)		Entrada (°)		Salida (b)		Error	Derror
Error	Derror	Act	Dir	Error	Derror	Act	Dir	Error		
1	-10	-10	1	0	-10	-10	1	0	0%	0%
2	-10	0	1	0	-10	0	1	0	0%	0%
3	-10	10	1	0	-10	10	1	0	0%	0%
4	0	-10	0	0	0	-10	0	0	0%	0%
5	0	0	0	0	0	0	0	0	0%	0%
6	0	10	0	0	0	10	0	0	0%	0%
7	10	-10	1	1	10	-10	1	1	0%	0%
8	10	0	1	1	10	0	1	1	0%	0%
9	10	10	1	1	10	10	1	1	0%	0%

5.2. Comprobación de las Reglas Sistema 2.

En la tabla 10 se muestran los resultados de la comparación entre el *Toolbox*TM y en lenguaje C para el sistema 2. Las columnas de “Entrada” muestran valores reales de entradas al sistema difuso como prueba para su validación y tiene unidades angulares (grados), las columnas de salida son valores binarios y representa la salida obtenidas por el sistema difuso después de aplicar los valores de entradas. La columna de “error” muestra la comparación en porcentaje de los resultados obtenidos entre las columnas que abarcan el *Toolbox*TM y el lenguaje C.

Tabla 10. Comparación de resultados entre el *Toolbox* de *Matlab* y código en lenguaje C para el motor de ángulo de ataque

COMPARACIÓN DE RESULTADOS						
NO. REGLA	TOOLBOX		LENGUAJE C		ERROR	
	Entrada (kg)	Salida (°)	Entrada (kg)	Salida (°)	Ent	Sal
1	0	0	0	0	0%	0%
2	1	7.5	1	7.5	0%	0%
3	2	15	2	15	0%	0%

5.3. Simulación en *Simulink*®

En la figura 52 se muestra el esquema general desarrollado en el ambiente *Simulink*™ de *Matlab*® para la simulación de ambos sistemas difusos. La entrada o referencia (*Set Point*) es la señal entregada por los sensores, el primer sistema corresponde a la dirección del viento y para el segundo es el peso del UAV. Para el sistema uno se utilizó un bloque de memoria que genera un retraso en la señal de entrada provocando un cambio en el tiempo en el error, que será utilizada para la resta de la variable cambio en el error. El motor utilizado es un motor a pasos híbrido, la salida se manda directamente a una ventana de *plot* para visualizar los resultados. Para el primer sistema ésta salida se retroalimenta para cerrar el lazo.

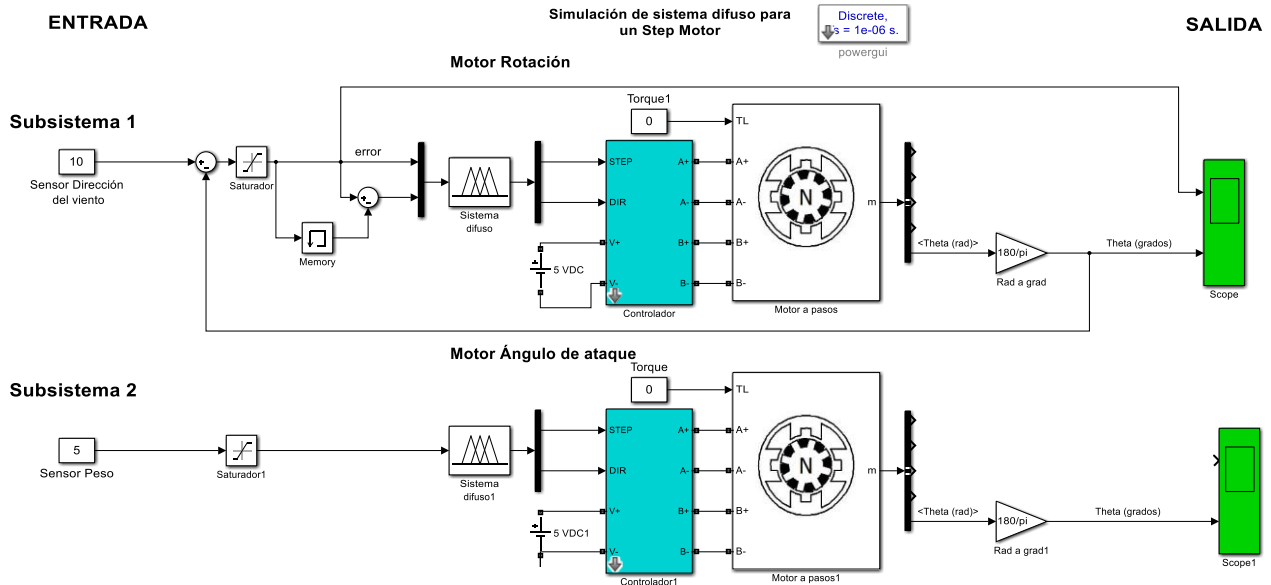


Figura 52. Esquema desarrollado para la simulación de los sistemas difusos en *Simulink*® de *Matlab*® incluyendo los actuadores

La simulación se corrió en Matlab R2015a®, con un tiempo de 2 segundos. Los valores utilizados para el motor son los siguientes:

- Inductancia de los devanados (H): $7.5e-3$
- Inductancia resistiva (Ohm): 8.3
- Paso por ángulo (grados): 0.7
- Flujo máximo (V-s): 0.005
- Inercia de arranque (N-m): 0.002
- Inercia Total (kg-m-m): $1.2e-7$
- Fricción total (kg-m/s): $1e-7 \cdot 1000$

Para validar los resultados se simuló a diferentes entradas, mostrándose para una entrada de 10° y 180° con el motor en un valor inicial de 0° y una segunda simulación con entrada de 10° y con el motor a un valor inicial de 80° .

Los resultados obtenidos se muestran en dos gráficas, en la parte superior se encuentra el comportamiento del error debido a la entrada o referencia y en la parte inferior se encuentra el ángulo alcanzado con respecto a la referencia. El eje 'y' corresponde al ángulo del motor y el eje 'x' al tiempo en segundos alcanzado.

En la figura 53 se muestran los resultados a una entrada de 10° con el motor en una posición inicial de 0° .

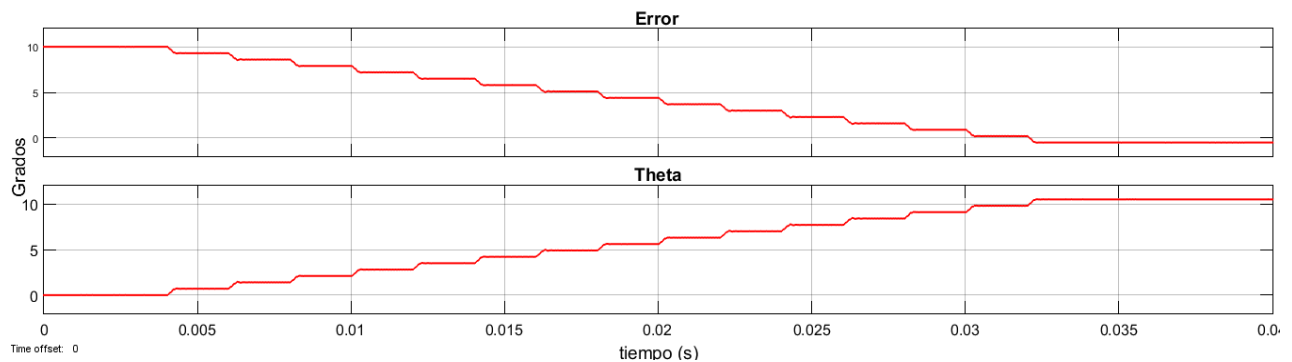


Figura 53. Resultados para una entrada de 10° con el motor a 0°

La figura 54 se muestra los resultados para una entrada de 180° con el motor en una posición inicial de 0° .

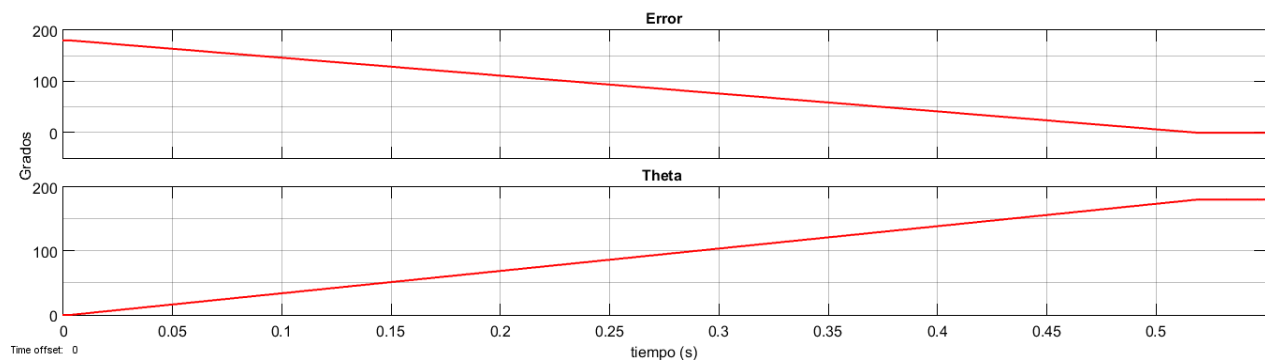


Figura 54. Resultados para una entrada de 180° con el motor a 0°

La figura 55 muestra los resultados a una entrada de 0° pero el motor en una referencia de 5° .

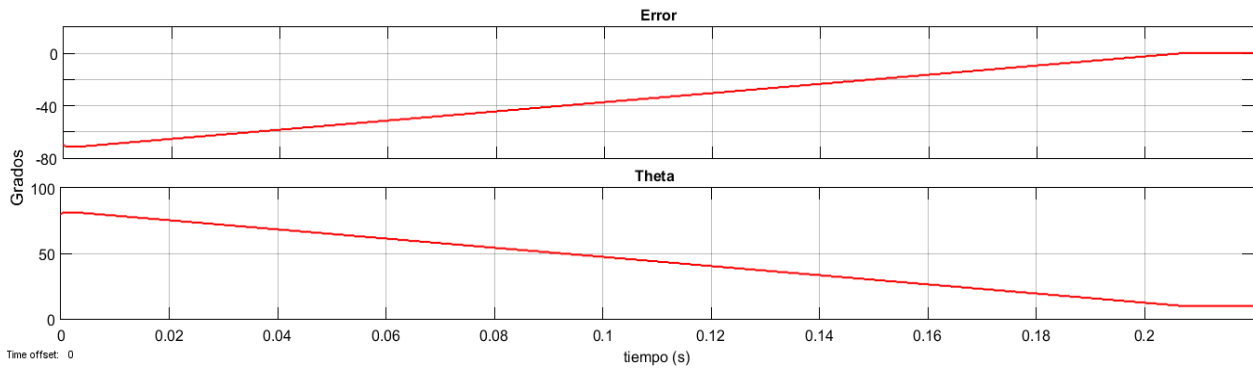


Figura 55. Resultados para una entrada de 10° con el motor en 80°

Debido a la resolución y a la escala en los resultados, no es posible apreciar el error en estado estable, así como el sobrepaso máximo, por lo cual se realizó un zoom cerca de la referencia, (fig. 56).

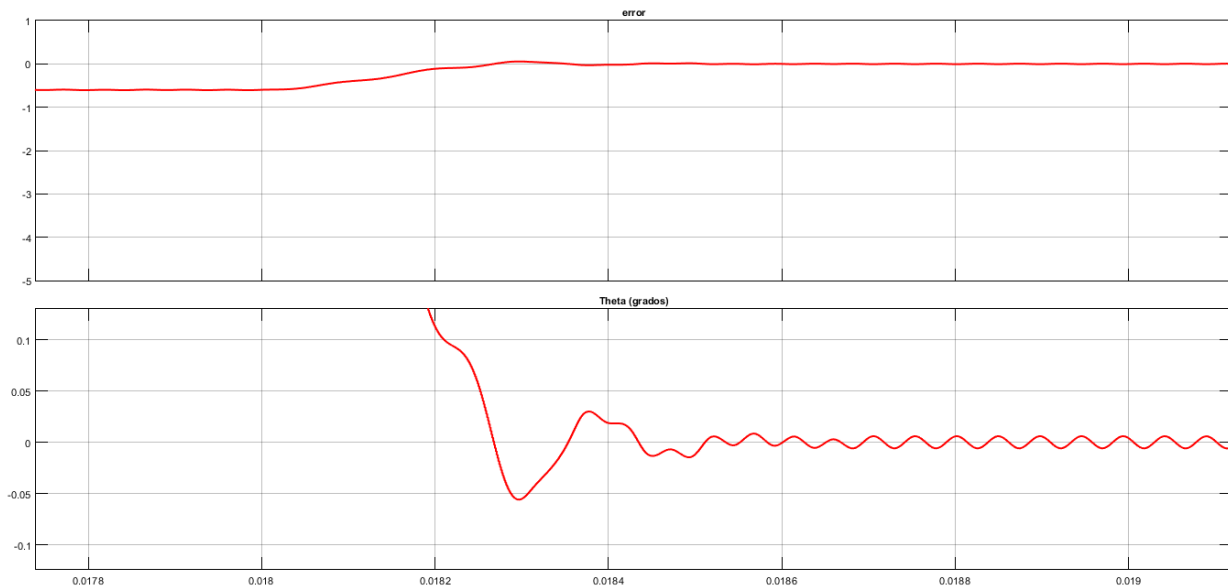


Figura 56. Sobrepaso máximo debido al error en estado estable

Al estar el error en cero (estado estable), el sobrepaso obtenido fue de 0.05° , es casi despreciable con respecto a su punto de control o *Set Point*.

La figura 57 se muestra el resultado para una entrada de 1 kg y el motor encontrándose en 0° . Para éste peso, si se evalúa en la función $f(x) = 7.5x$, lo que se encuentra es que le corresponde un valor de 7.5 , mismo obtenido en la figura 56, comprobándose así el resultado.

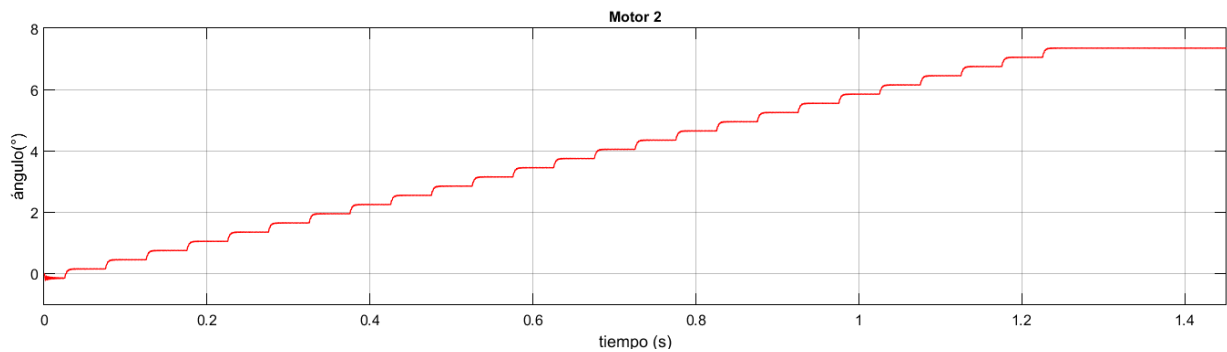


Figura 57. Resultados para una entrada de 1 kg con el motor en 0°

La figura 58 muestra el resultado para una entrada de 2 kg y el motor encontrándose en 0° . Éste es el peso máximo que tiene un micro-UAV, tanto el valor que le corresponde es el ángulo máximo, que para el modelo utilizado corresponde a un valor de 15° , de la figura 56 el valor máximo obtenido es 15° . Mismo resultado se obtiene al evaluar 2 kg en la función $f(x) = 7.5x$.

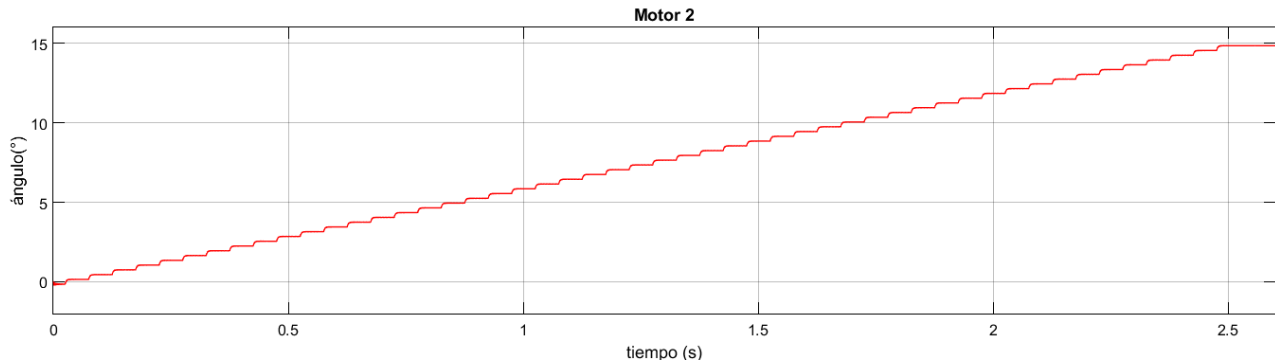


Figura 58. Resultados obtenidos para una entrada de 2 kg y el motor en una referencia de 0°

5.4. Resultados experimentales

Para validar las simulaciones se instrumentó con un motor a pasos las dos orientaciones, rotación sobre la vertical y ángulo de ataque, activando por partes los sensores de dirección de viento y peso del UAV. Se comprobó en él la respuesta experimental de los dos subsistemas difuso a las mismas entradas que la simulación. Las pruebas experimentales y resultados fueron desarrollados en laboratorio.

En la figura 59 se muestra el resultado experimental en laboratorio a una entrada de 10° con el motor en una posición inicial de 0° .

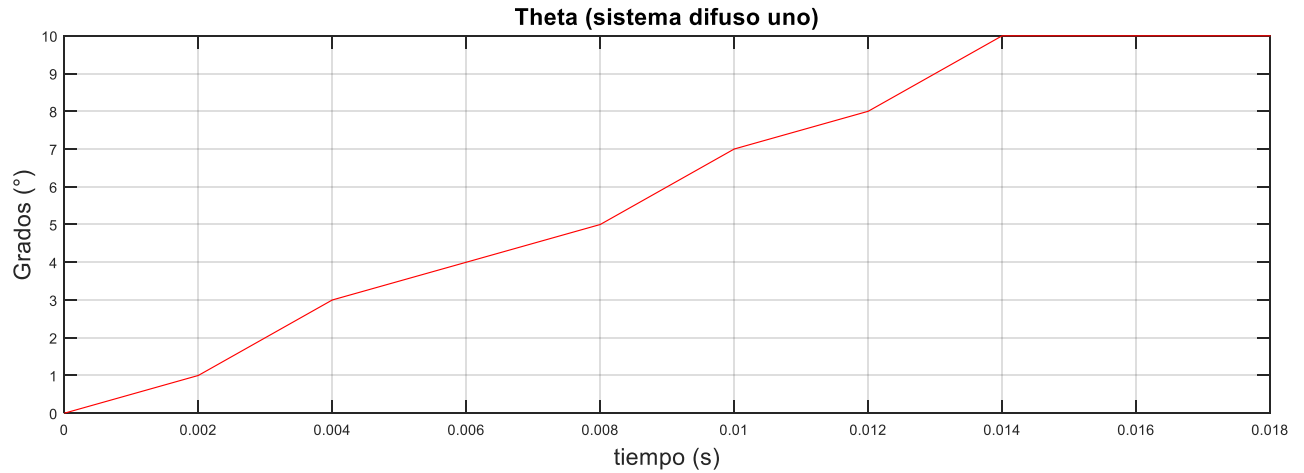


Figura 59. Resultado experimental para una entrada de 10° con el motor a 0°

En la figura 60 se muestra el resultado experimental en laboratorio para una entrada de 180° con el motor en una posición inicial de 0°.



Figura 60. Resultado experimental para una entrada de 180° con el motor a 0°

La figura 61 muestra el resultado para una entrada de 1 kg y el motor encontrándose en 0°.

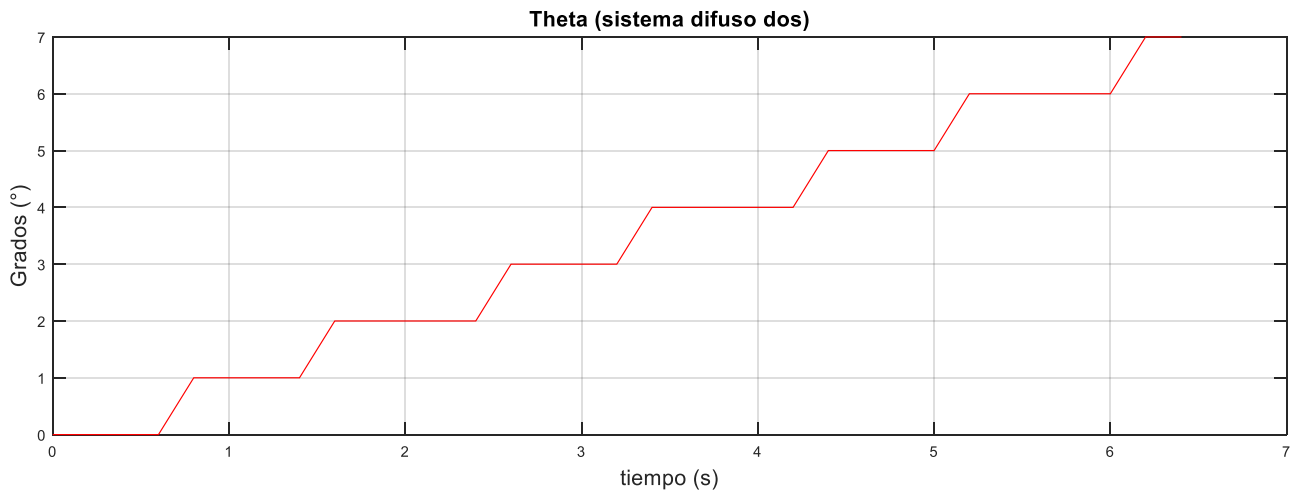


Figura 61. Resultados para una entrada de 1 kg con el motor en 0°

La figura 62 muestra el resultado para una entrada de 2 kg y el motor encontrándose en 0°.

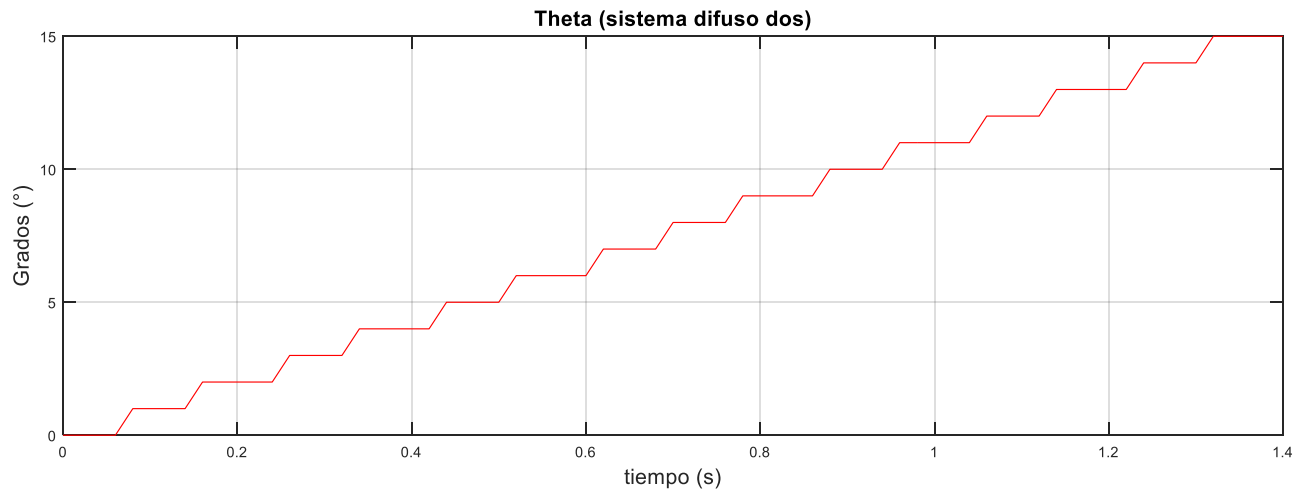


Figura 62. Resultados para una entrada de 2 kg con el motor en 0°

5.5. Comprobación de Resultados Embebidos

Se instrumentó un motor a pasos con un cople que une el eje del motor con el eje de un sensor y dos prisioneros para ajustarlos, el sensor utilizado fue un potenciómetro lineal de vuelta infinita (fig. 63).



Figura 63. Cople motor-sensor

Los resultados obtenidos de forma experimental en laboratorio se analizaron y compararon con los resultados obtenidos en simulación, se utilizó la fórmula del error experimental (ec. 83), el error experimental es una desviación del valor medido de una magnitud física respecto al valor real de dicha magnitud [69].

El cálculo del error experimental consiste en tomar el error absoluto (ec. 83) dividido entre el valor medido y multiplicado por 100%, (ec. 84). En este caso el valor medido es el obtenido de forma experimentalmente en el laboratorio y valor real es el obtenido en simulación.

$$Error\ absoluto = |valor\ medido - valor\ real| \quad (83)$$

$$Error\ exp\ (\%) = \frac{Error\ absoluto}{valor\ medido} \times 100 \quad (84)$$

La tabla 11 muestra la comparación de resultados del sistema 1 entre la simulación y la parte experimental de laboratorio para el giro sobre la vertical activando solamente el sensor de dirección de viento. La tabla se divide en dos resultados. La primera corresponde a los resultados en función del valor de salida del sistema difuso y el segundo en función del tiempo máximo alcanzado. La columna de entrada corresponde a las condiciones utilizadas para el experimento, en la columna simulación se muestran los resultados simulados obtenidos y la columna experimental los resultados obtenidos con $\pm 10^\circ$ debido a la incertidumbre del sensor de dirección de viento (Anexo B), la columna error exp muestra el error experimental en porcentaje.

Tabla 11. Comparación de resultados Sistema 1 en respuesta final y tiempo final

ENTRADA (°)	SIMULACIÓN (°)	EXPERIMENTAL (°) $\pm 10^\circ$	ERROR EXP (%)
Motor: 0 Entrada: 10	10	10	0
Motor: 0 Entrada: 180	180	181	0.5 (defecto)
ENTRADA (°)	SIMULACIÓN (s)	EXPERIMENTAL (s)	ERROR EXP (%)
Motor: 0 Entrada: 10	0.03	0.01	135.7 (exceso)
Motor: 0 Entrada: 180	0.52	0.21	147.6 (exceso)

Resultados de funcionamiento:

-Para una entrada de 10° y el motor a 0° la simulación del sistema difuso entregó un valor de salida de 10° con un tiempo de 0.033s, en la parte experimental el sistema difuso alcanzó el valor de 10° en un tiempo de 0.014s, dando como resultado un error experimental del 0%

con respecto de la simulación y un error por exceso de 135.7% en el tiempo alcanzado, es decir, el sistema difuso experimental sintonizó en un tiempo menor con respecto a la simulación.

-Para una entrada de 180° y el motor a 0° la simulación del sistema difuso entregó un valor de salida de 180° en un tiempo de 0.52s, en la parte experimental el sistema difuso alcanzó el valor de 181° en un tiempo de 0.21s, dando como resultado un error experimental 0.5% con respecto de la simulación y un error por exceso de 147.6% en el tiempo alcanzado, es decir, el sistema difuso experimental sintonizó en un tiempo menor con respecto a la simulación.

La tabla 12 muestra la comparación de resultados del sistema 2 entre la simulación y la parte experimental utilizando el sensor de peso del UAV. La tabla se divide en dos resultados. La primera corresponde a los resultados en función del valor de salida del sistema difuso y el segundo en función del tiempo máximo alcanzado. La columna entrada se encuentra las condiciones utilizadas para el experimento, la columna simulación se encuentra los resultados simulados obtenidos y la columna experimental los resultados experimentales obtenidos con $\pm 10^\circ$ debido a la incertidumbre del sensor de dirección de viento (Anexo B), la columna error exp muestra el error experimental en porcentaje.

Tabla 12. Comparación de resultados Sistema 2 en respuesta final y tiempo final

ENTRADA	SIMULACIÓN (°)	EXPERIMENTAL (°) $\pm 10^\circ$	ERROR EXP (%)
Motor: 0° Entrada: 1 kg	7.5	7.2	4.1 (defecto)
Motor: 0° Entrada: 2 kg	15	15	0
ENTRADA	SIMULACIÓN (s)	EXPERIMENTAL (s)	ERROR EXP (%)
Motor: 0° Entrada: 1 kg	1.2	0.6	79.4 (exceso)
Motor: 0° Entrada: 2 kg	2.4	1.3	110 (exceso)

De la tabla 12 para una entrada de 1 kg y el motor encontrándose a 0° la simulación del sistema difuso entregó un valor de salida de 7.5° con un tiempo de 1.22s, en la parte experimental el sistema difuso alcanzó un valor de 7.2° en un tiempo de 0.68s, dando como resultado un error experimental de 4.1% con respecto de la simulación, y un error por exceso en el tiempo alcanzado de 79.4%, es decir, el sistema difuso experimental sintonizó en un tiempo menor con respecto a la simulado.

De la tabla 12 para un entrada de 2 kg y el motor encontrándose a 0° la simulación del sistema difuso entregó un valor de salida de 15° con un tiempo de 2.4s, en la parte experimental el sistema difuso alcanzó un valor de 15° en un tiempo de 1.3s, dando como resultado un error experimental de 0% con respecto de la simulación en el valor alcanzado, y un error por exceso en el tiempo alcanzado de 110%, es decir, el sistema difuso experimental sintonizó en un tiempo menor con respecto a la simulado.

5.5. Instrumentación Electrónica

En las figuras 64 y 65 se muestran el sistema instrumentado. En *protoboard* se encuentran los *drivers* controladores para los motores a pasos y una fuente de alimentación de 5 V.

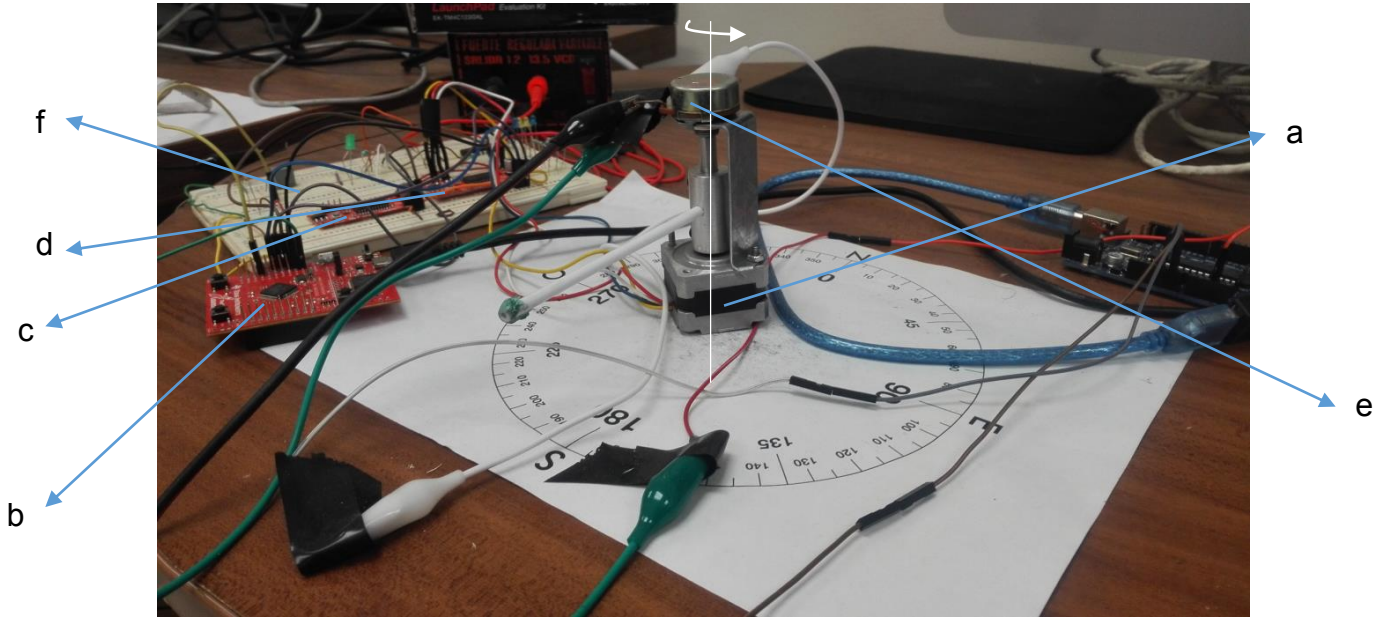


Figura 64. Sistema instrumentado

a) Motor, b) tarjeta electrónica, c) *driver* controlador, d) potenciómetro, f) *protoboard*

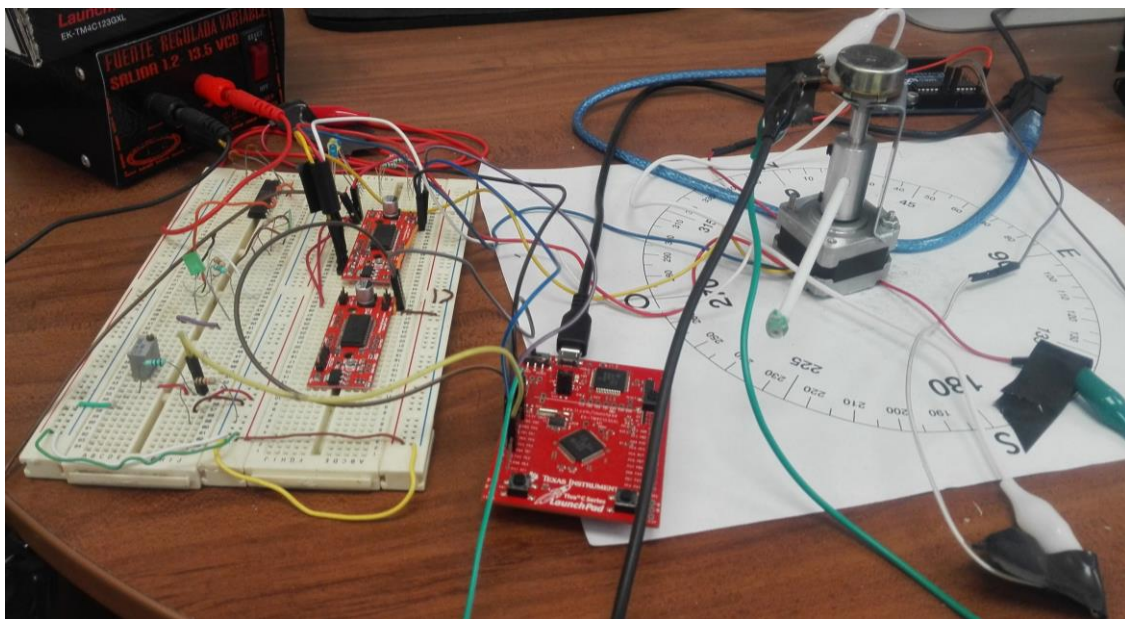


Figura 65. Instrumentación electrónica (vista frontal)

CAPÍTULO 6. CONCLUSIONES

6.1. Conclusiones y discusión

Se desarrolló una propuesta de sistema de ángulo de rotación y ángulo de ataque difuso basado en el método de inferencia de Sugeno para ser integrado como sistema inteligente a una lanzadera de μ -UAV de ala fija. Los sistemas difusos desarrollados mostraron a través de simulaciones su correcto funcionamiento y precisión al momento de controlar al alcanzar un tiempo máximo de 0.033 segundos para el primero y 0.034 para el segundo, con un error en estado estacionario igual a cero sin la presencia de sobrepaso. Para el caso experimental, en los subsistemas 1 y 2 se encontraron mejoras en 2.4 veces y 2.1 respectivamente que en la parte simulada; en ambos subsistemas el comportamiento de la respuesta del motor se mantuvo. El filtro de media móvil eliminó en su totalidad el ruido de los sensores, en el caso del peso se utilizó un segundo en cascada.

Se enlistan las actividades y resultados del sistema

- Se diseñó un sistema inteligente utilizando el método de inferencia de Sugeno para los ángulos de rotación y ataque para una lanzadera de micro-uav's de ala fija.
- Se diseñaron dos subsistemas difusos para el ángulo de rotación y ángulo de ataque de dos motores a pasos utilizando el *Toolbox*TM de lógica difusa de *Matlab*®.
- Se simularon los dos subsistemas con entradas variables y como salida dos motores a pasos utilizando *Simulink*© de *Matlab*®.
- Se comprobó la controlabilidad de los subsistemas al orientar ambos actuadores en función de diferentes entradas, demostrando así el correcto funcionamiento de los subsistemas difusos.
- El sobrepaso máximo en la simulación fue menor al 1%

- Se probaron y calibraron dos sensores para medir la dirección del viento y el peso del UAV.
- Se programó un código en lenguaje C para los dos subsistemas difusos.
- Se compararon los resultados de las salidas obtenidas entre el código en lenguaje C y el diseño en *Matlab*®, dando como resultado un 0% de error entre ambos.
- Se programaron los dos subsistemas difusos dentro de una tarjeta electrónica *TIVA C LAUNCHPAD* utilizando el código en lenguaje C
- Se programó dentro de la tarjeta electrónica dos filtros de media móvil en cascada tomando 20 muestras cada uno permitiendo eliminar el ruido en su totalidad
- La comparación de los resultados experimentales en laboratorio para el subsistema 1 con la simulación mostró que la parte experimental tuvo un error máximo de 0.5%
- La comparación de los resultados experimentales para el subsistema 2 con la simulación mostró que la parte experimental tuvo un error máximo 4%
- La comparación de los resultados experimentales en laboratorio para el subsistema 1 con la simulación mostró que sintoniza 147.6% o 2.4 veces mejor que la parte simulada
- La comparación de los resultados experimentales para el subsistema 2 con la simulación mostró que sintoniza 110% mejor que la parte simulada

6.2. Trabajo a futuro

El trabajo a futuro es instrumentar los dos subsistemas con dos motores para ejecutar la rotación sobre la vertical y ángulo de ataque he integrarla a la lanzadera de micro-UAV's.

6.3. Publicaciones y presentaciones en foros académicos

J. Balanzá, N. Kémper, G. Velasco, 2016. Sistema Difuso Aplicado a Lanzaderas de μ -UAVs de Ala Fija en el Monitoreo de Zonas Verdes, 2da Convención Internacional de Ciencias Técnicas, Santiago de Cuba, Cuba. 21-23 de Junio.

J. Balanzá, N. Kémper, G. Velasco, 2015. Sistema Difuso Embebido Aplicado a una Lanzadera de Micro-UAV's de Ala Fija, Memoria Congreso de Instrumentación SOMI XXX, Cd. de Durango, Durango. 28-30 Octubre.

C. Ortega, J. Álvarez, G. Velasco, J. Balanzá 2015. Implementación de un sistema embebido para el control de trayectorias de un mini cuadro-rotor, Memoria Congreso de Instrumentación SOMI XXX, Cd. de Durango, Durango. 28-30 Octubre.

BIBLIOGRAFÍA

- [1] R. Austin, 2011. ***Unmanned Aircraft Systems: UAVS Design, Development and Deployment***. John Wiley & Sons, ISBN 978-0-470-05819-0, pp. 331.
- [2] **Imagen sistema aéreo no tripulado**, NASA. [Online] https://www.nasa.gov/sites/default/files/images/693260main_graphic1176_882.jpg. [Acesado: 10-Oct-2015].
- [3] **Imagen drone de ala fija walkera e-eyes**, *Walkera e-eyes*. [Online]. Disponible en: <http://www.indoheli.com/image/cache/data/eyes/Eeyes8-500x500.jpg>. [Acesado: 10-Oct-2015].
- [4] **Imagen drone con despegue tipo carrera**, *Hydra Technologies de Mexico*. [Online]. Disponible en: https://lahistoriadeldia.files.wordpress.com/2012/08/drone_azteca-0b497.jpg [Acesado: 10-Oct-2015].
- [5] **Imagen drone con despegue tipo RATO**, *Ejército de los Estados Unidos*. [Online]. Disponible en: http://www.airforce-technology.com/projects/rq5a-hunter-uav/images/hunter_4.jpg. [Acesado: 10-Oct-2015].
- [6] **Imagen drone con despegue tipo VTOL**, *Ejército de los Estados Unidos*. [Online]. Disponible en: <http://images6.alphacoders.com/413/413917.jpg>. [Acesado: 08-Oct-2015].
- [7] **Lanzamiento manual de drone de ala fija**, *Dr. J. Frye Instituto de California* [Online]. Disponible en: http://gis.uww.edu/wordpress/wp-content/uploads/2014/08/IMG_8022.jpg. [Acesado: 06-Oct-2015].
- [8] Z. Novakovic, Nikola Medar, 2013. **Analysis of a UAV Bungee Cord Launching Device**, *Sci. Tech. Rev.*, vol. 63, no. 3, pp. 41–47.
- [9] L. Gherman, 2012. **An Electromagnetic Launch System for UAVs**, *Air Force Acad.*

J., vol. 21, no. 2, p. 5.

- [10] V. Silkov, 2014. **Calculation of the Characteristics of a UAV Launch from a Ramp**, Cent. Sci. Inst. Armed Forces, vol. 18, no. 4, pp. 12–17.
- [11] B. Miller, 2014. C. Valoria, C. Warnock, J. Coutlee, **Lightweight UAV Launcher**, California Polytechnic State University.
- [12] M. Bernard, J. Rosenbaum, G. Petter, J. Gessler, 2003. **Portable catapult launcher for small aircraft**, US 6851647 B1.
- [13] B. Reck, 2003. **First design study of an electrical catapult for unmanned air vehicles in the several hundred kilogram range**, *IEEE Trans. Magn.*, vol. 39, no. 1, pp. 310–313.
- [14] B. Nainesh, 2012. **Design case study Catapult launcher for Unmanned Air Vehicles (UAV)**. Southampton, United Kingdom, vol. 2, pp. 40.
- [15] M. Kondratiuk, Z. Gosiewski, 2013. **Laboratory Stand of an Electromagnetic Multi-Coil Launcher for Micro Aerial Vehicles**, *Solid State Phenom.*, vol. 198, pp. 334–339.
- [16] M. Kondratiuk, Z. Gosiewski, 2013. **Simulation Model of an Electromagnetic Multi-Coil Launcher for Micro Aerial Vehicles**, *Solid State Phenom.*, vol. 198, pp. 406–411.
- [17] M. Kondratiuk, Z. Gosiewski, 2014. **Comparison of Electromagnetic Coil Launcher Model with Real-Device Characteristics**, *Solid State Phenom.*, vol. 214, pp. 58–66.
- [18] Z. Gosiewski, M. Kondratiuk, 2009. **Selection of Coils Parameters in Magnetic Launchers**, *Solid State Phenom.*, vol. 147–149, pp. 438–443.
- [19] Emmanuelli A, 2012. **Pneumatic OWL (Oblique Wildlife Launcher)**, Ph.D, University of Southampton, Southampton UK.

- [20] Kaluzhina I, 2011. **Double corded, rubber, launching platform for the unmanned aerial vehicle**, *Open Inf. Comput. Integr. Technol.*, vol. 51, no. UDC 629.7.014–519.
- [21] B. Skurdal, R. Gaigler, 2008. **Multi-Mission Electromagnetic Launcher**, *14th Symposium on Electromagnetic Launch Technology*, pp. 1–4.
- [22] X. Liu, C. Xia, S. Ma, 2013. **The Modeling and Simulation of UAV Pneumatic Launch System**, *Appl. Mech. Mater.*, vol. 299, pp. 27–30.
- [23] J. Francis, 2010. **Launch System for Unmanned Aerial Vehicles for use on RAN Patrol Boats**, *University of New South Wales*, pp. 8.
- [24] D. John, T. Brosseau, 2010. **Rifle launcher for small unmanned aerial vehicles (UAVs)**, US 7849628.
- [25] M. Stephen, 2003. **Pneumatic launcher for a lightweight air vehicle**, EP 1594736 B1.
- [26] E. Young, J. Reynolds, M. Curran, 2001. **Miniature aircraft catapult**, California Polytechnic State University, US.
- [27] Bae Systems Company, 2008. **Launch system**, EP 2119998 A1.
- [28] G. Bladimir, 2006, **Catapulta para aviones no tripulador**, RU 2493055.
- [29] L. Santamaría, N. Kémper, G. Velasco, 2014. **Desarrollo de un sistema difuso aplicado al despegue de micro-UAV**, *memoria de congreso Instrumentación XXIX (SOMI)*.
- [30] L. Santamaría, N. Kémper, G. Velasco, 2014. **Sistema difuso para la gestión de lanzamiento de micro-UAV's de ala fija**, memoria de congreso Simp. Mex. En Vehículos Aéreos no Tripul.
- [31] L. Santamaría, 2014. **Desarrollo de un sistema difuso aplicado al despegue de**

micro-UAV, Universidad Nacional Autónoma de México, Tesis Lic, pp.100.

- [32] L. Li, J. Hong, H. Wu, P. Li, X. Li, 2008. **Section Crossing Drive with Fuzzy-PI Controller for the Long Stroke Electromagnetic Launcher**, in *2008 14th Symposium on Electromagnetic Launch Technology*, pp. 1–5.
- [33] V. Sarac, 2013. **Application of Matlab/Simulink in hybrid stepper motor modeling**, Faculty of Mathematics & Natural Science-FMNS, Macedonia, vol.1.
- [34] J. Grimbleby, 1995. **Simple algorithm for closed-loop control of stepping motors**, *IEEE Proc. - Electr. Power Appl.*, vol. 142, no. 1, p. 5.
- [35] F. Betin, D. Pinchon, A. Jaafari, 1998. **Robust speed control of a stepping motor drive using fuzzy logic**, *IEEE International Conference on Control Applications*, vol. 2, pp. 948–952.
- [36] F. Betin, D. Pinchon, G. Capolino, 2000. **Fuzzy logic applied to speed control of a stepping motor drive**, *IEEE Trans. Ind. Electron.*, vol. 47, no. 3, pp. 610–622.
- [37] J. Barman, P. Konwar, G. Das, 2014. **Design of a PD like Fuzzy Logic Controller for Precise Positioning of a Stepper Motor**, *ADBU Journal of Engineering Technology*, vol. 1, pp. 4.
- [38] A. Kabde, A. Savio, 2014. **Position Control of Stepping Motor**, *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering-IJAREEIE*, Vol. 3, Issue 4, ISSN: 2278-8875, pp. 5.
- [39] S. Roshan, L. Rupak, 2014. **Simulation Of Fuzzy Logic Controller For Stepper Motor Drive For Abstrction Of Maximum Power From Solar Panel – A Review**, *Int. J. Emerg. Technol. Adv. Eng.*, vol. 4, no. 4, pp. 146–148.
- [40] A. Mohamed, A. Zaki, 1998. **Microstepping Control of Hybrid Stepper Motor Using Fuzzy Logic for Robotics Application**, Electronics Research Institute of Egipt, ISBN: 0-96661-35-0-3, pp. 4.

- [41] X. Kong, B. Zhang, X. Zhang, G. Li, 2009. **Design of Step Motor Fuzzy Control System and its Application**, *Appl. Mech. Mater.*, vol. 16–19, pp. 135–139.
- [42] P. Melin, O. Castillo, 2002. **Intelligent system for control of a stepping motor drive using a hybrid neuro-fuzzy approach**, *Proceedings of the IEEE International Symposium on Industrial Electronics ISIE-02*, vol. 1, pp. 305–309 vol.1.
- [43] P. Melin, O. Castillo, 2003. **Intelligent control of a stepping motor drive using a hybrid neuro-fuzzy approach**, *Soft Comput.*, vol. 8, no. 8, pp. 546–555.
- [44] P. Melin, O. Castillo, 2001. **Adaptive control of a stepping motor drive using a hybrid neuro-fuzzy approach**, *10th IEEE International Conference on Fuzzy Systems. (Cat. No.01CH37297)*, vol. 1, pp. 155–158.
- [45] C. Chen, 2010. **TSK-Type Self-Organizing Recurrent-Neural-Fuzzy Control of Linear Microstepping Motor Drives**, *IEEE Trans. Power Electron.*, vol. 25, no. 9, pp. 2253–2265.
- [46] A. Dannenberg, 2005. **Fuzzy Logic Motor Control with MSP430x14x**, Texas Instrument, Dallas, TX, Application Report SLAA235, pp. 8.
- [47] D. Waba, C. Patil, 2013. **Fuzzy Control Based Solar Tracker Using Arduino UNO**, *International Journal of Engineering and Innovative Technology (IJEIT)*, Vol. 2, Issue 12, ISSN: 2277-3754.
- [48] J. Anderson, 2010. **Fundamentals of Aerodynamics**, 15 Edicion McGraw-Hill Education, ISBN 978-0-07-339810-5.
- [49] P. Fahlstrom, T. Gleason, 2012. **Introduction to UAV Systems**, 4 Edición John Wiley & Sons, ISBN 978-1-119-97866-4.
- [50] J. Prado, 2007. **Sistema de simulación para pruebas de algoritmos de orientación y control de satélites pequeños**, Tesis Doctorado, Programa de Maestría y Doctorado en Ingeniería FI-UNAM.

- [51] J. Synge, A. Schild, 2012. **Tensor Calculus**, Dover Publications, ISBN: 0-486-63612-7, pp. 321.
- [52] L. Reznik, 1997. **Fuzzy Controllers Handbook: How to Design Them, How They Work**. Elsevier Ltd, ISBN 0-7506-3429-4, pp. 285.
- [53] L. Zadeh, 1973. **Outline of a New Approach to the Analysis of Complex Systems and Decision Processes**, IEEE Trans. Syst. Man. Cybern., vol. SMC-3, no. 1, pp. 28–44.
- [54] E. Mamdani, S. Assilian, 1975. **An experiment in linguistic synthesis with a fuzzy logic controller**, Int. J. Man. Mach. Stud., vol. 7, no. 1, pp. 1–13.
- [55] M. Sugeno, G. Kang, 1988. **Structure identification of fuzzy model**, Fuzzy Sets Syst., vol. 28, no. 1, pp. 15–33.
- [56] A. Piña, 2009. **Síntesis de Sistemas de Control Borroso Estables por Diseño**, Tesis Doctoral, Universidad de Huelva, pp. 263.
- [57] A. Kulkarni, 2001. **Computer Vision and Fuzzy-Neural Systems**, Prentice Hall, ISBN: 978-0135705995, pp. 504.
- [58] M. Sugeno, K. Murakami, 1984. **Fuzzy parking control of model car**, The 23rd IEEE Conference on Decision and Control, pp. 902–903.
- [59] M. Triola, 2004. **Estadística**. Pearson Educación, ISBN 9789702612872, pp. 140.
- [60] **Fuzzy Inference Process - MATLAB & Simulink**. [Online]. Available: <http://www.mathworks.com/help/fuzzy/fuzzy-inference-process.html#FP348>. [Accessed: 29-Oct-2015].
- [61] R. Cayssials, 2014. **Sistemas embebidos en FPGA**, 1ra edició. Marcombo: AlfaOmega, pp. 210.

- [62] **Imagen ángulo de ataque del Cessna 172.** [Online]. Disponible en: http://www.lapizarradeyuri.com/wp-content/uploads/2010/12/vuelo_angulo_ataque.jpg. [Acesado: 10-Oct-2015].
- [63] **Tiva™ C Series TM4C123G LaunchPad Evaluation Board Guía de Usuario.** [Online]. Disponible en: <http://www.ti.com/lit/ug/spmu296/spmu296.pdf>. [Acesado: 10-Oct-2015].
- [64] **DSP and Filters.** [Online]. Available: http://www.analog.com/media/en/technical-documentation/dsp-book/dsp_book_Ch15.pdf. [Accessed: 10-Oct-2015].
- [65] S. Shome, S. Vadali, U. Datta, S. Sen, A. Mulkherje, 2012. **Performance Evaluation of Different Averaging based Filter Designs Using Digital Signal Processor and its Synthesis on FPGA**, Int. J. Signal Process. Image Process. Pattern Recognit., vol. 5, pp. 75–92.
- [66] R. Hernandez, 2012. **Sensor Solar para Aplicación Espacial**, Tesis Licenciatura UNAM, pp. 109.
- [67] **Easy Driver Hook-up.** [Online]. Available: <https://learn.sparkfun.com/tutorials/easy-driver-hook-up-guide>. [Accessed: 10-Oct-2015].
- [68] **Schmalzhauz Easy Driver Web Page,** [Online]. Disponible en: <http://www.schmalzhaus.com/EasyDriver/>. [Acesado: 10-Oct-2015].
- [69] V. Athani, 1997. **Stepper Motors: Fundamentals, Applications And Design.** New Age International, pp. 200.
- [70] **Reporte Técnico Motor a pasos 5014-820 NEMA 14.** [Online]. Available: <http://s3.amazonaws.com/applied-motion-pdf/5014-820.pdf>. [Accessed: 10-Oct-2015].
- [71] **Step Motor Wiring Diagram.** [Online]. Available: <http://www.applied-motion.com/sites/default/files/StepMotorWiring-4-lead.pdf>. [Accessed: 10-Oct-2015].

ANEXO A. CARACTERÍSTICAS DE LA TARJETA ELECTRÓNICA TIVA C LAUNCHPAD

Procesador	ARM Cortex M4F 32-bit CPU
Velocidad de reloj	80 MHz
Memoria RAM	32 Kbytes de memoria RAM para almacenamiento de datos
Memoria ROM	2 Kbytes de EEPROM para almacenamiento de datos no volátiles
Memoria flash	256 Kbytes de memoria flash para almacenamiento de datos no volátiles
LED	LED RGB
Push Bottons	Dos botones de aplicación general
Headers de pines de salida	0.1-in (2.54-mm)
Reset	Botón de Reset

ANEXO B. CARACTERÍSTICAS DEL SENSOR POTENCIÓMETRO.

Obtenido directamente de la hoja de datos del fabricante.

Característica	Valor
Ángulo rotacional total	300° ± 10°
Tolerancia total de la Resistencia	± 20%
Terminales	A, B, C. B: variacional
Tensión máxima	Terminal B: 150 AC Otras terminales: 100V AC
Resistencia total	1MΩ
Ciclos de vida	15,000 ciclos

ANEXO C. CARACTERÍSTICAS DEL SENSOR DE PESO.

Obtenidos directamente de la hoja de datos del fabricante.

Force Sensitivity Range	< 100 g to > 10 kg
Pressure Sensitivity Range	< 1.5 psi to > 150 psi (< 0.1 kg/cm ² to > 10 kg/cm ²)
Part-to-Part Force Repeatability	± 15% to ± 25% of established nominal resistance
Single Part Force Repeatability	± 2% to ± 5% of established nominal resistance
Force Resolution	Better than 0.5% full scale
Break Force (Turn-on Force)	20 g to 100 g (0.7 oz to 3.5 oz)
Stand-Off Resistance	> 1MΩ
Switch Characteristic	Essentially zero travel
Device Rise Time	1-2 msec (mechanical)
Lifetime	> 10 million actuations
Temperature Range	-30°C to +70°C
Maximum Current	1 mA/cm ² of applied force
Sensitivity to Noise/Vibration	Not significantly affected
EMI / ESD	Passive device

ANEXO D. TEORÍA MOTOR A PASOS

Actuadores

Un actuador es un dispositivo que transforma algún tipo de energía, como la eléctrica, a energía mecánica; los motores a pasos son ejemplo de actuadores. Estos motores son dispositivos donde el movimiento de rotación se realiza en forma incremental utilizando pares de bobinas, que van desde los 90° hasta $\pm 1.8^\circ$.

Principio de funcionamiento de los motores a pasos

Los motores a pasos están compuestos por dos partes: estator, que está fijo a la carcasa del motor y rotor, que se encuentra en medio del motor y es móvil (fig. 66). El estator se encuentra en la periferia del motor y es la encargado de generar el flujo principal, mientras que el rotor se encuentra en el centro de este, unido al eje, y su función es reaccionar a la excitación del estator produciendo el movimiento de rotación [69].

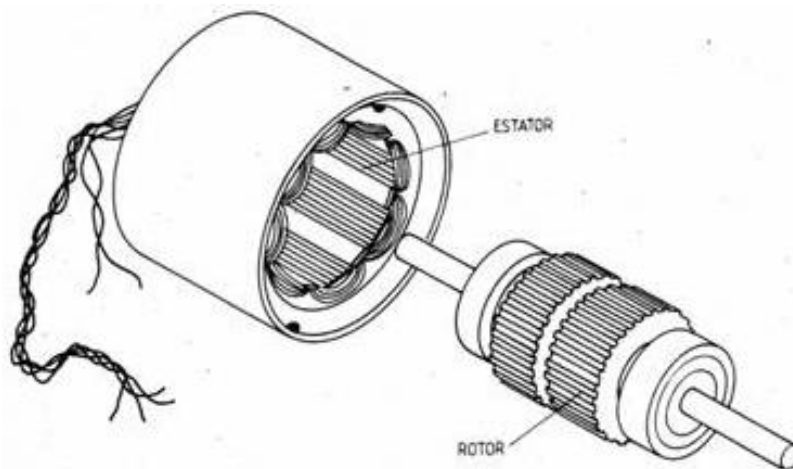


Figura 66. Estator y rotor de un motor a pasos [69]

El principio de funcionamiento de estos consiste en alimentar sólo algunas de las bobinas del estator con corriente continua, generando un flujo con una dirección fija, que al

interactuar con el rotor del motor produce un par en función del ángulo de desfase de la posición anterior o de equilibrio [69].

Al excitar al estator, se crearán los polos *NORTE-SUR (N - S)*, provocando la variación del campo magnético formado. La respuesta del rotor será seguir el movimiento de dicho campo, es decir, orientará sus polos *N - S* hacia los polos *S - N* del estator, respectivamente. Cuando el rotor alcanza esta posición de equilibrio, el estator cambia la orientación de sus polos y tratará de buscar la nueva posición de equilibrio, manteniendo dicha situación de manera continua, se conseguirá un movimiento giratorio y continuo del rotor, produciendo de esta modo el giro o avance del motor [69].

Cuando se efectúa un cambio de polaridad en las bobinas del estator se genera un paso y se le denomina ángulo de paso.

Motor a pasos híbridos

El motor híbrido consiste en un estator dentado y un rotor de tres partes. El rotor contiene dos piezas de polos separados por un magneto permanente, (fig. 67).

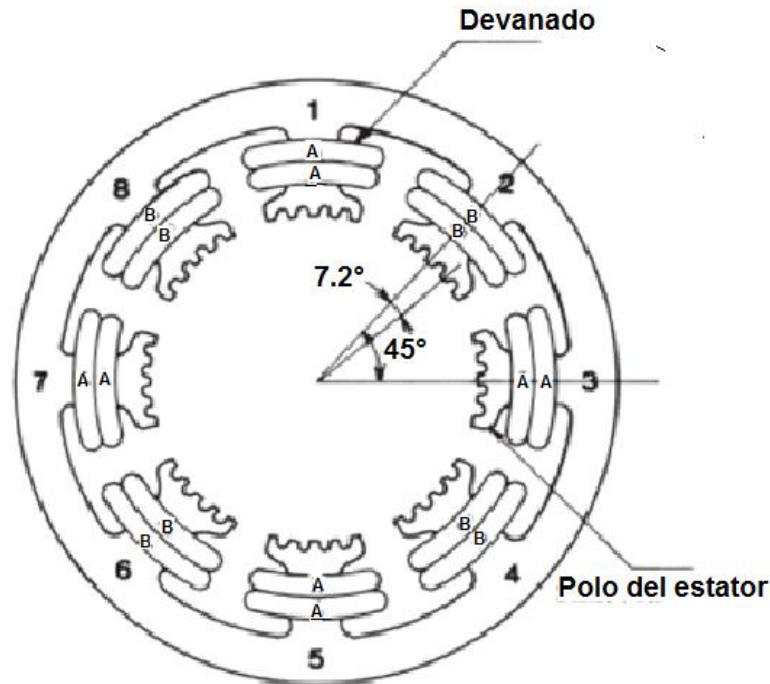


Figura 67. Estator de un motor a pasos híbrido [69]

Al excitar las bobinas del estator se alinean los dientes del estator con los del rotor de la carga apropiada. Paso 1) se excitan las bobinas impares, polo norte (N) en las bobinas 1 y 5 y, y polo sur (S) en las bobinas 3 y 7; y el rotor se alinea con el campo. Paso 2 se excitan las bobinas pares, polos N las 2 y 6, y polo S las 4 y 8; entonces el rotor busca alinear los dientes más próximos a los del estator se desplaza un ángulo que corresponde a un paso de ángulo. Paso 3 se alimentan nuevamente las bobinas impares, polo N en las bobinas 3 y 7, y polo S en las bobinas 1 y 5, el rotor se gira otro paso de ángulo, (fig. 68).

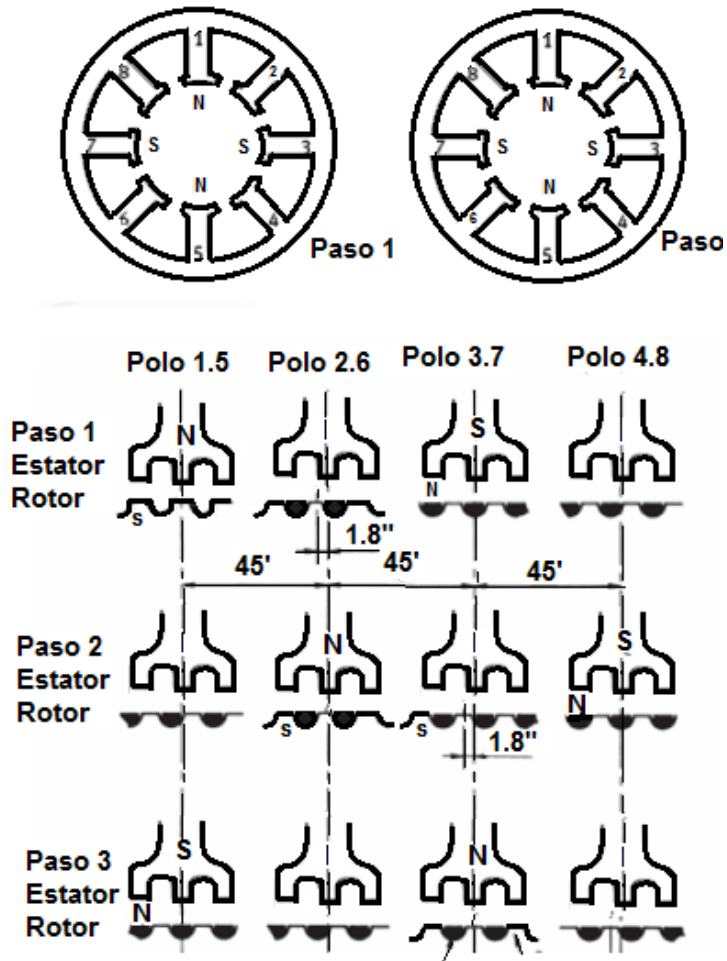


Figura 68. Principio de funcionamiento de un motor a pasos híbrido [70]

Conexión y Secuencia de Manejo.

Para conseguir el giro del rotor en un determinado número de grados, las bobinas del estator deben ser excitadas secuencialmente. Un motor a pasos bipolar tiene cuatro bobinas conectadas en pares, llamadas A+ y A-, B+ y B-, (fig. 69). Para lograr las distintas polaridades del campo es necesario invertir las dos terminales de cada bobina, esto puede efectuarse mediante software o utilizando algún *driver* controlador.

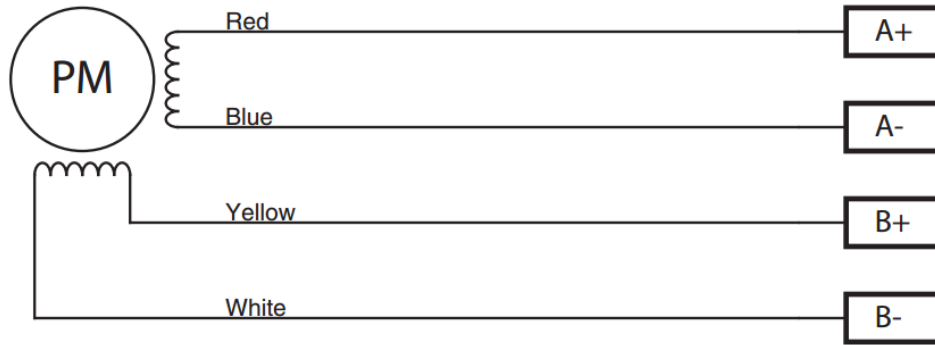


Figura 69. Conexión del motor a pasos híbrido bipolar modelo 5014-820 [71]

Existen tres métodos para el control de la secuencia de manejo de un motor a pasos bipolar, el cual depende de las secuencias de encendido de las bobinas.

-Paso simple: Consiste en activar cada bobina una a una y por separado, esta secuencia de encendido tiene la característica que no se obtiene mucha fuerza ya que sólo es una bobina a la vez, (fig. 70).

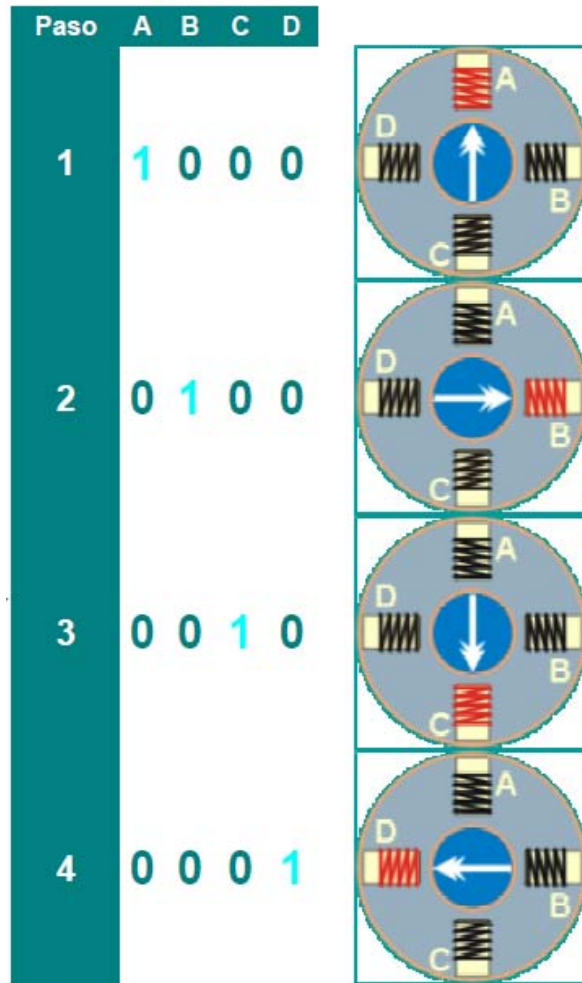


Figura 70. Secuencia de pasos simple para activar un motor bipolar [71]

-Paso doble: Se activan las bobinas de dos en dos, formándose un campo magnético que atraerá con más fuerza y retendrá el rotor del motor en el sitio, (fig. 71).

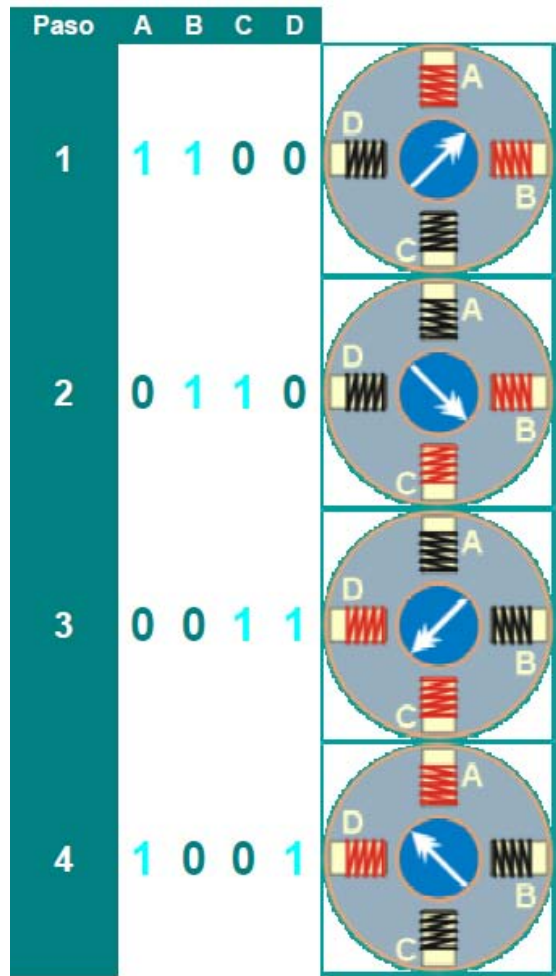


Figura 71. Secuencia de pasos doble para activar un motor a pasos bipolar [71]

ANEXO E. CÓDIGO EN LENGUAJE C

```
//El convertidor analógico digital del TIVA es de 12 bits, lo que da una resolución de (1212-1) = 4095
//La conversión de ADC a volts es la siguiente:
//Volts = (ADC*3.3)/4095;
//El valor de volts obtenido se convierte a grados con la siguiente conversión:
//Angulo = (volts*360)/3.3;
//La conversión de grados a pasos
//Pasos = (Angulo * 1600)/360
//El dato del número de pasos se obtuvo de la página:
http://www.schmalzhaus.com/EasyDriver/Examples/EasyDriverExamples.html, explicado
en el ejercicio 1
//360° en pasos es igual a 1600, de ahí se obtiene la conversión
//El dato obtenido después de realizar dicha operación se envía al motor.
```

```
int Distance = 0; // Variable para guardar el número de pasos que hemos dado
int angulo = 0;
int x, n, otro = 0;
float angulo1 = 0;
int angulo2 = 0;
int angulo3 = 0; //Variable para definir el ángulo que deseamos y la variable que lo
almacena de pasos a ángulo
```

```
float sensorValue2 = 0, sensorValue[60], sensorValue1 = 0; //Variables y arreglo utilizados
para el filtro de media móvil
float sensorValue2_2 = 0, sensorValue_1[60], sensorValue1_1 = 0; //Variables y arreglo
utilizados para el filtro de media móvil 2
```

```
float angulo1_1 = 0;
int angulo2_2 = 0;
int angulo3_3 = 0; //Variable para definir el ángulo que deseamos y la variable que lo
almacena de pasos a ángulo
```

```
float ne, ze, pe, nc, zc, pc, S1, S2, S3, S4, S5, S6, S7, S8, S9;
float ACTIVA_CERO = 0, ACTIVA_UNO = 0, ACTIVACION = 0, DIRECCION_CERO = 0,
DIRECCION_UNO = 0, DIRECCION = 0;
```

```
int error[2], pasos_angulo = 0, error_deriv = 0;
```

```
//VARIABLES SEGUNDO MOTOR
float poco_peso, peso_medio, peso_maximo, w = 0, defuzz = 0; //VARIABLES SEGUNDO
MOTOR
int Distance2 = 0; // Variable para guardar el número de pasos que hemos dado
```

```
int angulo_pasos = 0; //Variable para convertir de Ángulo a pasos
int pasos2_angulo = 0; //Variable para convertir de pasos a Ángulo
```

```
void setup()
```

```
{
```

```
    pinMode(2, OUTPUT);
    pinMode(3, OUTPUT);
    digitalWrite(2, HIGH);      //Dirección de giro
    digitalWrite(3, LOW);      //Activación motor
```

```
    Serial.begin(9600);
    delay(1);
```

```
    //SEGUNDO MOTOR
    pinMode(4, OUTPUT);
    pinMode(5, OUTPUT);
```

```
}
```

```
void loop()
```

```
{
```

```
    Distance = 0;
```

```
    //FILTRO MEDIA MÓVIL
```

```
    sensorValue1 = 0;      //Inicializando a cero al inicio del loop la variable acumulativa.
    Esta variable es la que suma los valores del ADC
```

```
    for(n = 0; n < 60; n++)
    {
```

```
        sensorValue[n] = 0;      //Se vuelve a inicializar la variable que guarda los valores
    del ADC a cero.
```

```
    }
```

```
    for(n = 30; n < 50; n++)
    {
```

```

        sensorValue[n] = analogRead(A3); //Se lee el valor del ADC y se guarda en 5
secciones de memoria
        delayMicroseconds(10);

    }

    for(n = 30; n < 50; n++)
    {

        sensorValue1 += sensorValue[n]; //Suma ponderada de los valores acumulados
en el arreglo

    }

    sensorValue2 = sensorValue1/20; //Se obtiene el promedio o la media móvil

//TERMINA FILTRO

    angulo1 = (sensorValue2 * 3.3) / 4095; //Conversión de ADC a volts
    angulo2 = ((angulo1 * 360) / 3.3) + 1; //Conversión de volts a grados móviles
constante de error, explicado arriba
    angulo3 = (angulo2 * 1600) / 360; //Conversión de grados a pasos

    angulo1_1 = (sensorValue2_2 * 3.3) / 4095; //Conversión de ADC a volts
    angulo2_2 = ((angulo1_1 * 360) / 3.3) + 1; //Conversión de volts a grados móviles
constante de error, explicado arriba
    //angulo3_3 = (angulo2_2 * 1600) / 360; //Conversión de grados a pasos

    //angulo2 = angulo2 - 90;

//Proporcional

//otro = 0 - angulo2;
//error[0] = otro + pasos_angulo;
error[0] = angulo2 - angulo2_2;
//Serial.println(error[0]);
Serial.println(angulo2);
angulo3_3 = abs((error[0] * 1600) / 360); //Conversión de grados a pasos

//error[1] = ((angulo2 - pasos) - error[0]) / 50e-3;
//error[0] = 0;
error[1] = 0;

ne = funcz(error[0]);

```

```

ze = funct(error[0]);
pe = funcn(error[0]);

//Cambio Error / Derivativo

error_deriv = (error[1]);
nc = funcz_der(error_deriv);
zc = funct_der(error_deriv);
pc = funcn_der(error_deriv);

//INFERENCIA: IMPLICACIÃ“N

S1 = min(ne, nc);
S2 = min(ne, zc);
S3 = min(ne, pc);
S4 = min(ze, nc);
S5 = min(ze, zc); //FUERZAS DE DISPARO
S6 = min(ze, pc);
S7 = min(pe, nc);
S8 = min(pe, zc);
S9 = min(pe, pc);

//AGREGACIONES, UNIONES

ACTIVA_CERO = COMPARA_MAX(S4, S5, S6);
ACTIVA_UNO = COMPARA_MAX2(S1, S2, S3, S7, S8, S9);

//AGREGACIONES, UNIONES

DIRECCION_CERO = COMPARA_MAX2(S1, S2, S3, S4, S5, S6);
DIRECCION_UNO = COMPARA_MAX(S7, S8, S9);

//DEFUZZ ACTIVACIÃ“N

ACTIVACION = (ACTIVA_CERO*0 + ACTIVA_UNO*1)/(ACTIVA_CERO +
ACTIVA_UNO);

//DEFUZZ DIRECCIÃ“N

DIRECCION = (DIRECCION_CERO*0 + DIRECCION_UNO*1)/(DIRECCION_CERO
+ DIRECCION_UNO);

//ENVÃ“O AL MOTOR

```

```

if(ACTIVACION >= 1 && DIRECCION >= 1)
{
    while(Distance != angulo3_3)
    {
        //2 = Direccion, 3 = Activaci3n

        digitalWrite(2, LOW);
        digitalWrite(3, HIGH);
        delayMicroseconds(100);    //Tiempo para realizar la operaci3n, el m3nimo es
de 100, m3s peque±o tiene errores en los desplazamientos
        digitalWrite(3, LOW);
        delayMicroseconds(100);    //Tiempo para realizar la operaci3n, el m3nimo es
de 100, m3s peque±o tiene errores en los desplazamientos
        Distance = Distance + 1;    // variable acumulativa, guardar cada paso

    }

//SEGUNDO MOTOR, SEGUNDO MOTOR, SEGUNDO MOTOR

    float peso = analogRead(A2);

//Primer filtro

if (peso <= 50)
{
    peso = 0;
}

//Termina primer filtro

float salida = (peso * 5) / 4095; //Conversi3n de ADC a volts
float salida2 = ((salida*1000) / 5) / 1000; //Conversaion de volts a gramos

    Distance2 = 0;

//Segundo filtro: Saturador

    if(salida2 >= 2)

```

```

    {
        salida2 = 2;
    }

    else if(salida2 <= 0)
    {
        salida2 = 0;
    }

//Termina saturador

Serial.println(salida2);

poco_peso = func_izq(salida2);    //Funci3n izquierda
peso_medio = func_cent(salida2); //Funci3n derecha
peso_maximo = func_der(salida2); //Funci3n izquierda

//AGREGACIONES, UNIONES

w = COMPARA_MAX(poco_peso, peso_medio, peso_maximo); //Fuerza de
disparo, peso

//DEFUZZ ACTIVACI3N

defuzz = w*(10*salida2)/w; //defuzz = wi*zi/wi, promedio ponderado
angulo_pasos = abs((defuzz * 1600) / 360); //Conversi3n de grados a pasos

    while(Distance2 != angulo_pasos)
    {

        //4 = Direccion, 5 = Activaci3n

        digitalWrite(4, HIGH);
        digitalWrite(5, HIGH);
        delayMicroseconds(300); //Tiempo para realizar la operaci3n, el m3nimo es
de 100, m3s peque1o tiene errores en los desplazamientos
        digitalWrite(5, LOW);
        delayMicroseconds(300); //Tiempo para realizar la operaci3n, el m3nimo es
de 100, m3s peque1o tiene errores en los desplazamientos
        Distance2 = Distance2 + 1; // variable acumulativa, guardar cada paso

    }

```

```
pasos2_angulo = (Distance2*360)/1600;
```

```
Serial.println(pasos2_angulo);
```

```
}
```

```
//FIN FIN FIN FIN FIN FIN FIN FIN
```