



**UNIVERSIDAD MICHOACANA DE
SAN NICOLÁS DE HIDALGO**

**FACULTAD DE INGENIERÍA ELÉCTRICA
DIVISIÓN DE ESTUDIOS DE POSGRADO**

**MODELADO, ESTIMACIÓN PARAMÉTRICA Y CONTROL DE
UN SISTEMA DE LEVITACIÓN MAGNÉTICA DE UNO Y DOS
GRADOS DE LIBERTAD**

TESIS

Que para obtener el grado de:

MAESTRO EN CIENCIAS EN INGENIERÍA ELÉCTRICA

Presenta:

Ing. Marco Antonio Rodríguez Martínez

Director de Tesis:

Dr. Roberto Tapia Sánchez

Morelia, Michoacán, agosto de 2020



A Dios por permitirme llegar aquí, a mis padres quienes siempre me han apoyado en este largo camino, así como a mi hermano quien siempre ha sido un ejemplo a seguir.

A mi novia Esmeralda Carrillo que siempre me ha apoyado en todo. A mis amigos de generación de Maestría, que compartieron esta experiencia.

Al Dr. Roberto Tapia Sánchez por permitirme realizar este trabajo bajo su tutela, por su paciencia, consejos y apoyo en este trabajo el cual agradezco sinceramente.

Agradezco al Consejo Nacional de Ciencia y Tecnología el apoyo brindado para cursar los estudios de Maestría en la División de Estudios de Posgrado de la Facultad de Ingeniería Eléctrica, a través de la beca que me fue otorgada durante la estancia del mismo.

Resumen

En esta tesis se presenta el modelado matemático de un sistema de levitación magnética de un grado de libertad tanto para masas con propiedades magnéticas y paramagnéticas. Se realiza la implantación física la cual abarca desde la creación del electroimán, la instrumentación y caracterización de los sensores, la eliminación del ruido haciendo uso del Filtro de Kalman. Adicionalmente se procede a realizar una estimación paramétrica embebida del sistema usando el método de mínimos cuadrados y Filtro de Kalman para la validación del modelado matemático, posteriormente se realiza la estimación del parámetro de la fuerza electromagnética usando redes neuronales a nivel simulación e implementación en el microcontrolador. El control del sistema se hace bajo dos técnicas de control no convencionales. Modos deslizantes de primer orden y modos deslizantes de orden superior “SuperTwisting”, se realiza una comparativa entre embebido y Matlab ® y se determina porque en este tipo de plantas la acción de control debe de ser rápida. Se extiende el estudio a un sistema de levitación magnética de dos grados de libertad donde se realiza la estimación paramétrica del sistema mediante tres técnicas (mínimos cuadrados, mínimos cuadrados recursivos, y Filtro de Kalman-Bucy). Finalmente se hace el diseño de controladores robustos para el seguimiento de trayectoria.

Palabras Clave: Modos Deslizantes, SuperTwisting, Levitación Magnética, ANN, Estimación Paramétrica.

Abstract

This thesis presents the mathematical modeling of one and two freedom degree magnetic levitation system for both masses with magnetic and paramagnetic properties. Physical implantation is carried out, showing the creation of the electromagnet, the instrumentation and characterization of the sensors, the extraction of noise using the Kalman filter. In addition, an embedded parametric estimation of the system is carried out using the least squares method and the Kalman filter for the validation of the mathematical modeling. Besides, the parametric estimation of the electromagnetic force parameter is carried out using neural networks, at the simulation level and with its implementation in the microcontroller. The control of the system is done considering two unconventional control techniques: first order sliding modes and higher order "SuperTwisting" sliding modes, a comparison is made between Embed vs Matlab determined because these controls are appropriate for this type of plants with fast dynamics. The study is extended to a two degrees of freedom magnetic levitation system where the parametric precision of the system is performed using three techniques (Least Squares, Recursive Least Squares, and Kalman-Bucy Filter). Finally, the design of robust controls for tracking is done.

Keywords: Sliding Modes, SuperTwisting, Magnetic Levitation, ANN, Parametric Estimation.

Lista de Publicaciones

“Implementation for a Magnetic Levitation System: Embedded Parametric Estimation and Control”

Marco Antonio Rodríguez Martínez, Roberto Tapia Shancez

2019 IEEE International Autumn Meeting on Power, Electronics and Computing (ROPEC 2019). Ixtapa, Mexico.

Contenido

Dedicatoria	III
Resumen	V
Abstract	VII
Lista de Publicaciones	IX
Contenido	XI
Lista de Figuras	XV
Lista de Tablas	XVII
Lista de Símbolos	XIX
1. Introducción	1
1.1. Estado del arte	2
1.2. Justificación	4
1.3. Objetivos de la tesis	4
1.3.1. Objetivo general	4
1.3.2. Objetivos particulares	5
1.4. Planteamiento del problema	5
1.5. Descripción de capítulos	6
2. Modelado del sistema de levitación magnética de 1 y 2 GDL	7
2.1. Introducción	7
2.2. Modelado de un sistema de levitación magnética de un grado de libertad	7
2.3. Proceso de modelado	9
2.4. Modelado usando objeto ferromagnético	9
2.5. Modelado usando objeto con propiedades magnéticas	11
2.6. Linealización del sistema	11
2.7. Modelo en espacio de estado	12
2.8. Modelado de un sistema de levitación magnética de dos grados de libertad	14
2.8.1. Linealización del sistema	18
2.9. Conclusiones	18
3. Implementación del sistema de levitación magnética de 1 GDL	19
3.1. Introducción	19
3.2. Elementos del sistema	19
3.2.1. Imán de neodimio	20

3.2.2.	Diseño del actuador – electroimán	21
3.2.3.	Microcontrolador	23
3.2.4.	Control de electroimán	24
3.2.5.	Puente H L298N	26
3.2.6.	Sensor de efecto hall	27
3.2.7.	Lectura de posición	29
3.2.8.	Sensor se corriente	31
3.3.	Obtención del parámetro K	34
3.4.	Estimación paramétrica	35
3.4.1.	Método de mínimos cuadrados	35
3.4.2.	Procedimiento de estimación de R y L	36
3.4.3.	Obtención de los vectores	37
3.4.4.	Resultados de identificación	39
3.5.	filtro de Kalman	40
3.5.1.	Implementación del filtro de Kalman	42
3.5.2.	Obtención de los vectores adicionando filtro de Kalman	43
3.5.3.	Resultados de identificación utilizando el filtro de Kalman	44
3.6.	Control por modos deslizantes de primer orden	45
3.6.1.	Consideraciones de implementación	47
3.6.2.	Implementación modos deslizantes en Matlab - Arduino	48
3.6.3.	Implementación modos deslizantes en Arduino embebido	50
3.7.	Control por modos deslizantes de orden superior	51
3.7.1.	Supertwisting	52
3.7.2.	Implementación SuperTwisting en Matlab - Arduino	52
3.7.3.	Implementación SuperTwisting en Arduino embebido	54
3.8.	Implementación añadiendo filtro de Kalman	56
3.8.1.	Implementación control con modos deslizantes y filtro de Kalman en Arduino embebido	56
3.8.2.	Implementación control SuperTwisting y filtro de Kalman en Arduino embebido	58
3.9.	Estimación del parámetro de la fuerza electromagnética	59
3.9.1.	Estimación con redes neuronales	61
3.9.2.	Red neuronal embebida	65
3.9.3.	Red neuronal reconstruida	66
3.10.	Conclusiones	68
4.	Simulación, estimación paramétrica y control del modelo del sistema de levitación magnética de 2 GDL	71
4.1.	Introducción	71
4.2.	Estimación paramétrica	71
4.2.1.	Mínimos cuadrados	72
4.2.2.	Obtención de los vectores	73
4.2.3.	Mínimos cuadrados recursivos	75
4.2.4.	Filtro Kalman-Bucy	77
4.2.5.	Resultados de simulación KBF	79

4.3. Simulación control por modos deslizantes de primer orden	81
4.4. Simulación control por modos deslizantes de orden superior	84
4.4.1. Conclusiones	87
5. Conclusiones	89
5.1. Conclusiones generales	89
5.2. Trabajos futuros	90
A. Codigos para implementación	91
A.1. Filtro de Kalman	91
A.2. Control con modos deslizantes	93
A.3. Control Supertwisting	95
A.4. Control con modos deslizantes y filtro de Kalman	97
A.5. Control SuperTwisting y filtro de Kalman	100
A.6. Red neuronal embebida	103
A.7. Reconstrucción red neuronal	108
B. Vectores de identificación	111
B.1. Vectores de identificación del sistema con un conversor ADC de 8 bits . . .	112
B.2. Vectores de identificación del sistema con filtro de Kalman	121
Referencias	131

Lista de Figuras

2.1. Sistema de levitación magnética de un grado de libertad	8
2.2. Sistema de levitación magnética de dos grados de libertad	15
2.3. Modelo del electroimán.	15
3.1. Sistema de levitación magnética de 1 GDL	20
3.2. Campo magnético imán N42	21
3.3. Elaboración del electroimán	22
3.4. Prueba del electroimán	22
3.5. Medición de la resistencia e inductancia	23
3.6. Estructura del puente H	25
3.7. Puente H sentido de giro.	25
3.8. Driver L298N	26
3.9. Conexión de alimentación L298N.	27
3.10. Estructura interna y pinout del sensor de efecto hall.	29
3.11. ADC valores vs distancia.	30
3.12. Sensor de corriente ACS712.	31
3.13. Dinámica voltaje vs corriente del sensor de corriente.	33
3.14. Diagrama de la identificación del sistema embebida.	37
3.15. Señal de entrada-voltaje	38
3.16. estado x1-corriente	38
3.17. Derivada corriente	39
3.18. Algoritmo KF	41
3.19. Implementación del filtro de Kalman con una masa estática	42
3.20. Implementación del filtro de Kalman con una masa en movimiento	43
3.21. Identificación del sistema con el filtro de Kalman	44
3.22. Esquema de simulación para el control con modos deslizantes del sistema de levitación magnética de un grado de libertad	48
3.23. Respuesta del error para el control con modos deslizantes Matlab-Arduino	49
3.24. Respuesta de la acción de control para el control con modos deslizantes Matlab-Arduino	49
3.25. Señales de control y error para el control con modos deslizantes embebido.	50
3.26. Señal de error para el control con modos deslizantes embebido	51

3.27. Esquema de simulación para el control SuperTwisting del sistema de levitación magnética de un grado de libertad	53
3.28. Respuestá del error para el control SuperTwisting Matlab-Arduino	53
3.29. Respuestá de la acción de control para el control SuperTwisting Matlab-Arduino	54
3.30. Señales de control y error para el control SuperTwisting embebido.	55
3.31. Señal de error para el control SuperTwisting embebido.	55
3.32. Control con modos deslizantes y filtro de Kalman - Arduino	56
3.33. Señal de error para el control con modos deslizantes y filtro de Kalman - Arduino	57
3.34. Control SuperTwisting y filtro de Kalman - Arduino	58
3.35. Señal del error para el control con modos deslizantes y filtro de Kalman - Arduino	59
3.36. Red neuronal para aproximar el parámetro electromagnetico.	61
3.37. Función de activación lineal	62
3.38. Función de activación sigmoide	63
3.39. Red neuronal implementada en Matlab Simulink	64
3.40. Error cuadrático medio de la red neuronal	64
3.41. Red neuronal embebida	66
3.42. Reconstrucción de la red neuronal	68
3.43. Salida - reconstrucción red neuronal	68
4.1. Entradas al sistema de levitación magnética de dos grados de libertad.	73
4.2. Sistema de levitación magnética de dos grados de libertad.	74
4.3. Resultados de identificación	75
4.4. Diagrama de identificación usando el método de mínimos cuadrados recursivos	77
4.5. Filtro Kalman-Bucy	79
4.6. Kalman Bucy simulación un ciclo	80
4.7. Kalman Bucy simulación tiempo extendido	80
4.8. Salida del sistema en ambos ejes x,y	81
4.9. Referencia y salida del sistema para el eje x	82
4.10. Referencia y salida del sistema para el eje y	82
4.11. Señal de control para el eje x	83
4.12. Señal de control para el eje y	83
4.13. Salida del sistema en ambos ejes x,y con una acción de control SuperTwisting	85
4.14. Referencia y salida del sistema para el eje x con una acción de control SuperTwisting	85
4.15. Referencia y salida del sistema para el eje y con una acción de control SuperTwisting	86
4.16. Señal de control SuperTwisting para el eje x	86
4.17. Señal de control SuperTwisting para el eje y	87

Lista de Tablas

2.1. Parámetros del sistema.	17
2.2. Variables del sistema.	17
3.1. Pinout y valores del sensor de efecto hall	29
3.2. Lectura ADC- distancia	30
3.3. Modelos sensor de corriente	32
3.4. PinOut sensor de corriente	32
3.5. Lectura voltaje vs amperes	33
3.6. Resultados de identificación	39
3.7. Resultados de identificación con filtro de Kalman	44
3.8. Dependencia entre posición de equilibrio y la corriente de equilibrio	60
3.9. Validación de Red Neuronal	63
4.1. Resultados de identificación usando el método de mínimos cuadrados recursivos	77
B.1. Valores identificación 1	112
B.2. Valores identificación 2	113
B.3. Valores identificación 3	114
B.4. Valores identificación 4	115
B.5. Valores identificación 5	116
B.6. Valores identificación 6	117
B.7. Valores identificación 7	118
B.8. Valores identificación 8	119
B.9. Valores identificación 9	120
B.10. Valores identificación KF 1	121
B.11. Valores identificación KF 2	122
B.12. Valores identificación KF 3	123
B.13. Valores identificación KF 4	124
B.14. Valores identificación KF 5	125
B.15. Valores identificación KF 6	126
B.16. Valores identificación KF 7	127
B.17. Valores identificación KF 8	128
B.18. Valores identificación KF 9	129

Lista de Símbolos

$1GDL$	Un grado de libertad
$2GDL$	Dos grados de libertad.
KF	Filtro de Kalman.
seg	Ssegundos.
bps	Baudios por segundo.
ADC	Conversor Analógico – Digital.
PWM	Modulación por ancho de pulso
g	Fuerza de atracción gravitacional.
cm	Centímetros.
m	Metros.
A	Amperios.
C	Grados Centígrados.

Capítulo 1

Introducción

Se le llama levitación al fenómeno físico que se presenta cuando un objeto es mantenido de manera estable en el aire en una posición particular, sin sujeción mecánica alguna. Para que exista este fenómeno de levitación se deben cumplir dos condiciones:

- La presencia de una fuerza que contrarreste el peso del cuerpo.
- La existencia de una fuerza variable adicional que se oponga a cada desplazamiento del objeto.

A lo largo del tiempo, para el hombre ha sido de interés materializar este fenómeno y para ello ha empleado diversos principios físicos, entre los que cabe destacar la levitación electrostática (requiere que el cuerpo levitante esté cargado eléctricamente), levitación aerodinámica (presente en helicópteros), levitación acústica (se realiza usando ondas sonoras en objetos de pequeña masa), la levitación óptica (empleando radiación láser en objetos de masa pequeña), y por supuesto, la levitación magnética. La levitación magnética ha sido objeto de estudio por parte de investigadores desde mediados del siglo XX hasta la actualidad. Existen diversos fenómenos magnéticos donde está presente el fenómeno de la levitación: por un lado se encuentra la levitación debida a la superconductividad, en la cual por medio de el efecto Meissner se expulsan todas las líneas de flujo magnético del interior de un material superconductor a temperaturas por debajo de la temperatura crítica del material, logrando convertir el material superconductor en un diamagnético perfecto. Mediante el diamagne-

tismo de los materiales se produce la levitación: un material diamagnético se magnetiza de manera opuesta al campo que intenta magnetizarle, produciéndose de esta manera una repulsión entre el campo generador y el generado. Adicionalmente, la levitación magnética puede enmarcarse en dos principios [1] que abarcan toda una gama de aplicaciones de ingeniería desarrolladas hasta hoy: la levitación por repulsión y por atracción. La levitación por repulsión, conocida como EDS (Electrodynamic Suspension) se basa en la repulsión de dos campos magnéticos, cada uno puede ser generado por inducción electromagnética o con imanes permanentes. Los sistemas se desarrollan bajo este principio para lo cual son estables en un eje y tienen un punto de equilibrio natural. La levitación por atracción por su parte, conocida como EMS (Electromagnetic Suspension) se logra cuando un campo magnético atrae un cuerpo que se opone a la gravedad, produciéndose un equilibrio inestable entre la fuerza magnética y la fuerza gravitacional. Así, un sistema basado en este principio es inviable sin el uso de un módulo de control.

Los sistemas de levitación magnética han sido objeto de estudios en las últimas décadas debido a la importancia de trabajar en la disminución de la fricción de Coulomb debido al contacto mecánico [2]. Estos sistemas se han vuelto populares en sistemas de transportación terrestre de alta velocidad, son usados para rodamientos magnéticos y sistemas de aislamiento de vibraciones [3]. De manera particular, los rodamientos magnéticos en la industria de ingeniería han tenido gran demanda [4], donde se necesita alcanzar un posicionamiento con tolerancias de error en el orden de nanómetros, ancho de banda y rigidez, en la industria medica [5] en los trenes de levitación magnética [6] cuyo interés es reducir los costos por mantenimiento constante de partes por causas de desgaste y la necesidad de un cambio continuo de lubricantes empleados para disminuir efectos de fricción.

1.1. Estado del arte

En la ingeniería, la levitación magnética se ha consolidado como una forma de reducir el consumo energético, debido a que permite el desplazamiento relativo entre dos superficies eliminando la fricción, la cual es la principal responsable de las pérdidas de energía en sistemas electromecánicos tales como trenes, rodamientos, entre otros [7]. Así

mismo, la levitación magnética se ha empleado en el desarrollo de trampas magnéticas que hacen uso del efecto Meissner esto en materiales superconductores como lo son el helio líquido el cual es usado para repeler las líneas de campo y confinar el líquido en una región del espacio al ser restringido magnéticamente en todas las direcciones [8].

La levitación magnética ha sido usada con éxito en cañones electromagnéticos empleándose para mover el proyectil, evitando así el contacto de este con la superficie de la guía, anulando la fricción mecánica [9].

En medicina la aplicación de la levitación magnética ha aportado en el desarrollo de corazones artificiales implementados mediante bombas centrifugas donde el impulsor es levitado magnéticamente, esto aumenta el desempeño del sistema y minimiza el consumo de energía debido a la ausencia de fricción [10].

En la industria manufacturera, se han desarrollado máquinas de control numérico (CNC) cuyo soporte vertical es levitado magnéticamente, esto aumenta considerablemente la velocidad de desplazamiento de la pieza móvil y la precisión. También este fenómeno físico aparecen en el campo de los ambientes virtuales en lo que se conoce como tecnologías del tacto [11].

Se han diseñado sistemas de interacción con el tacto los cuales permiten tocar, sentir, manipular y alterar objetos levitados magnéticamente donde la posición y ángulos es monitoreada mediante sensores; esta información es enviada a interfaces virtuales donde se encuentra un entorno de realidad virtual. Esta aplicación tiene un enorme potencial en la industria del entretenimiento y la manipulación de artefactos [12].

En el ámbito académico, la levitación magnética representa uno de los sistemas dinámicos y altamente no lineales e inestables, debido a la naturaleza de los campos electromagnéticos y al objetivo de la levitación como tal para compensarlo con la fuerza gravitacional. Por esta razón durante décadas ha sido una plataforma ideal de trabajo para investigadores, en la cual el objetivo es implementar técnicas de control tanto clásicas como modernas [13-21].

1.2. Justificación

La levitación magnética desde hace ya varias décadas ha constituido una demostración que permite ilustrar el uso potencial de la levitación en aplicaciones enfocadas a la industria, el comercio, el transporte, la medicina o el entretenimiento. En la ingeniería eléctrica por ejemplo se encuentran aplicaciones asociadas al modelado de sistemas dinámicos [23], control de sistemas inestables [24], rodamientos de bajo roce o fricción [25], sistemas mecánicos de almacenamiento de energía [26], trampas magnéticas para fluidos con propiedades diamagnéticas, como el Helio líquido [27], levitación de microrobots en tres dimensiones [28] y sistemas de transporte de alta velocidad [29-30]. Las implicaciones económicas y sociales que este tipo de tecnología podría tener en una sociedad son enormes. Involucrando tanto la física como la ingeniería: la electrónica, el control, la computación y la instrumentación. Las razones anteriormente expuestas resultan motivadoras para continuar trabajado en este tipo de problemas, buscando hacer aportes adicionales tanto en el modelado e implementación física, como la aplicabilidad de los sistemas de levitación magnética.

1.3. Objetivos de la tesis

1.3.1. Objetivo general

Modelar y analizar un sistema de levitación magnética de uno y dos grados de libertad, así como implementar diferentes técnicas de control en él, además de realizar la identificación paramétrica de sus componentes.

1.3.2. Objetivos particulares

- Obtener el modelo del sistema de levitación magnética de un grado de libertad.
- Implementación del sistema. (Creación del electroimán, la caracterización de los sensores, la eliminación del ruido haciendo uso del Filtro de Kalman).
- Estimación paramétrica del sistema. (Uso de mínimos cuadrados y Filtro de Kalman).
- Estimación del parámetro de la fuerza electromagnética usando redes neuronales a nivel simulación e implementación.
- Control del sistema. (Uso de dos técnicas de control no convencional- Modos deslizantes de primer orden y modos deslizantes de orden superior “SuperTwisting”).
- Comparativa Embebido vs Matlab ®
- Obtención y simulación del sistema de levitación magnética de dos grados de libertad.
- Estimación paramétrica del sistema. (Uso de mínimos cuadrados, mínimos cuadrados recursivos, y Filtro de Kalman-Bucy).
- Diseño de controladores robustos para el seguimiento de trayectoria.

1.4. Planteamiento del problema

Se busca diseñar y construir un sistema de levitación magnética de repulsión de un grado de libertad tanto para masas con propiedades magnéticas como aquellas que no, así mismo el modelado matemático de un sistema de levitación magnética de dos grados de libertad, controlado por algoritmos no convencionales, usar compensación neuronal, donde la red neuronal hace la función de compensador de parámetros electromagnéticos que pueden ser medidos fácilmente, estimación de parámetros a partir de vectores de entrada/salida del modelo matemático. Para ello se harán consideraciones tanto físicas, como electrónicas y de control, integrándolas en el desarrollo de un prototipo funcional. Las consideraciones físicas se hacen sobre variables mecánicas como posición, velocidad, aceleración; y electromagnéticas como corriente, voltaje, magnetización, entre otras. Por otro lado, las consideraciones

electrónicas abarcan tópicos relacionados con la instrumentación, como la elección del sensor o el tipo de amplificación. Finalmente las consideraciones de control se hacen sobre las técnicas a emplear en cada uno de los aspectos relacionados con el sistema.

1.5. Descripción de capítulos

En el Capítulo 2 se presenta el desarrollo del modelo matemático de un sistema de levitación magnética de uno y dos grado de libertad, tipo atractivo usado para suspender un objeto ferromagnético o Magnético.

En el Capítulo 3 se realiza la implementación del sistema de levitación magnética de un grado de libertad, la cual abarca desde la creación del electroimán, análisis de los elementos controladores, la caracterización de los sensores, la eliminación del ruido haciendo uso del Filtro de Kalman (KF), estimación paramétrica del sistema por medio del método de mínimos cuadrados con y sin KF, control del sistema por distintas leyes de control y en combinación con el KF, tanto con Simulink como Embebido. Finalmente se realiza una estimación del parámetro electromagnético por medio de la recreación de una red neuronal a partir del entrenamiento de esta en Matlab ®, obteniendo sus pesos y bias para su implementación embebida en el microcontrolador.

En el Capítulo 4 se presentan los resultados de las simulaciones del sistema de levitación magnética de dos grados de libertad, así como la estimación paramétrica en la cual se aplican 3 técnicas distintas: mínimos cuadrados, mínimos cuadrados recursivos y filtro de Kalman-Bucy. Siendo la primer técnica fuera de línea y las otras dos en línea respectivamente. El control es aplicado por dos técnicas de control modos deslizantes de primer orden y Modos deslizantes de orden superior SuperTwisting

Finalmente en el Capítulo 5 se muestran las conclusiones a las cuales se ha llegado con esta tesis, así como los trabajos futuros.

Capítulo 2

Modelado del sistema de levitación magnética de 1 y 2 GDL

2.1. Introducción

En este capítulo se presenta el modelado del sistema de levitación magnética de uno y dos grados de libertad. En específico, se presenta el modelo no lineal, así como su linealización. Finalmente el modelo en espacio de estados es planteado.

2.2. Modelado de un sistema de levitación magnética de un grado de libertad

Esta sección está enfocada al desarrollo del modelo matemático de un sistema de levitación magnética de un grado de libertad, tipo atractivo usado para suspender un objeto ferromagnético o magnético. El sistema de levitación magnética considerado es constituido por una bola ferromagnética suspendida en un campo magnético, el cual está controlado por medio de la manipulación de la diferencia de potencial en el devanado eléctrico.

El análisis está enfocado solo en el movimiento de traslación realizado en el plano vertical. La bola ferromagnética en el sistema se encuentra influenciada por dos fuerzas:

- La fuerza de gravedad (g).
- La fuerza de sustentación electromagnética (f_e) producida por el campo generado por la bobina (Electroimán).

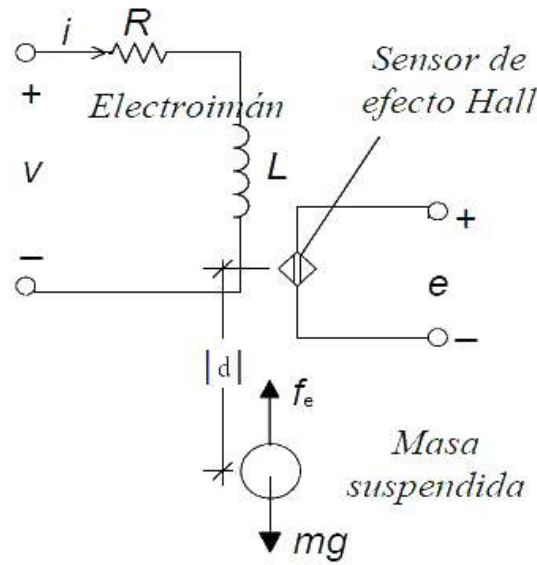


Figura 2.1: Sistema de levitación magnética de un grado de libertad

Los sistemas de suspensión magnéticos son sistemas que cuentan con dinámicas no lineales las cuales son inestables en lazo abierto, por lo que una retroalimentación de control es necesaria para estabilizarlos [34]. Se puede desarrollar el modelo matemático del sistema de levitación por medio de las ecuaciones diferenciales apropiadas de acuerdo a los principios mecánicos y eléctricos. En el modelo matemático propuesto, las variables de estado usadas son la posición de la bola (d), la velocidad de la bola y la corriente que pasa a través del circuito electromagnético (i), como se muestra en la Figura 2.1.

2.3. Proceso de modelado

Se considera que la posición d de la bola se incrementa de manera positiva en dirección hacia abajo del electroimán. Usando las leyes fundamentales de la mecánica clásica, mediante un diagrama de cuerpo libre; expresado mediante la ley de Newton para el equilibrio de fuerzas tiene la siguiente forma no lineal

$$m \frac{d^2[d(t)]}{dt^2} = mg - f_e \quad (2.1)$$

Donde:

- m = Masa del objeto a levitar.
- g = Fuerza de gravedad.
- $d(t)$ = Posición del objeto levitado (Distancia entre el electroimán y el objeto).
- f_e = Fuerza electromagnética.

Dependiendo del objeto y forma de éste, la f_e cambia debido a su dependencia de la corriente, posición y del tiempo $f_e(i, x, t)$, de este modo, si el objeto no fuera ferromagnético sino con propiedades magnéticas, éste induciría corriente en la bobina y afectaría a la f_e .

2.4. Modelado usando objeto ferromagnético

La Ecuación (2.1) se mantiene de la misma manera. De este modo la expresión para el balance energético dentro del sistema se muestra en la expresión (2.2):

$$dW_e = dW_{mec} + dW_t + dW_m \quad (2.2)$$

Donde:

- dW_e = Variación de la energía eléctrica.
- dW_{mec} = Variación de la energía mecánica.
- dW_t = variación de la energía térmica.

- dW_m = variación de la energía magnética
- Φ = flujo magnético

Asi mismo dW_m cuando el flujo magnético está variando y los cuerpos se están moviendo dentro del campo magnético está dada por la Ecuación (2.3):

$$dW_m = i\Phi - f_e dx. \quad (2.3)$$

De este modo haciendo uso del balance energético se puede plantear de acuerdo a [35] y teniendo en cuenta que para el problema de levitación magnética usualmente se toma como masa un cuerpo metálico con características ferromagnéticas, este cuerpo al ser sometido a la acción de la fuerza magnética producida por el electroimán flota en el aire a una distancia d . La fuerza magnética sobre la masa metálica está dada por la Ecuación (2.4).

$$f_e = K \left(\frac{i}{d} \right)^2 \quad (2.4)$$

Donde:

- d = Distancia entre la masa y el electroimán.
- i = La corriente del electroimán.
- K Constante de la fuerza electromagnética.

Donde: K está definida como $\frac{1}{2}\mu_0 AN^2$

- μ_0 = permeabilidad magnética en el vacío = $4\pi 10^{-7} N/A^2$
- A = Área total donde fluye el flujo magnético.
- N = Número de vueltas de cable de cobre esmaltado para construir el electroimán.

Como puede notarse el cambio de la dirección de corriente o inversión de polarización en el electroimán no influirá en la acción de la dirección de la fuerza sobre la masa. está Ecuación (2.4) solo es válida para campos uniformes y para una masa metálica ferromagnética.

2.5. Modelado usando objeto con propiedades magnéticas

En el sistema de levitación magnética del presente proyecto se usa un imán de neodimio; es decir, un cuerpo con magnetización propia, el cual al ser sometido al campo magnético generado por el electroimán no se rige por la Ecuación (2.4) debido a que el cambio de la polarización del electroimán sí genera cambios en la acción de la fuerza sobre el imán. En este caso la fuerza generada por el electroimán está dada por la Ecuación (2.5).

$$f_e = K \left(\frac{i}{d^4} \right) \quad (2.5)$$

Además, la Ecuación eléctrica del circuito del electroimán está determinada por la Ecuación (2.6).

$$V = Ri + L \frac{di(t)}{dt} \quad (2.6)$$

De este modo, la Ecuación (2.1) queda expresada como la Ecuación (2.7):

$$m\ddot{d} = mg - K \left(\frac{i}{d^4} \right) \quad (2.7)$$

2.6. Linealización del sistema

Para linealizar la Ecuación (2.7) se aplica la expansión por Series de Taylor alrededor de un punto de operación, se desprecian los términos de orden superior; por tanto la aproximación lineal [37] estará determinada como:

$$g(x_{10}, x_{20}, \dots, x_{n0}) + \left(\frac{\partial g(x)}{\partial x_1} \right) \Big|_{x=x_0} (x_1 - x_{10}) + \left(\frac{\partial g(x)}{\partial x_2} \right) \Big|_{x=x_0} (x_2 - x_{20}) + \dots + \left(\frac{\partial g(x)}{\partial x_n} \right) \Big|_{x=x_0} (x_n - x_{n0}) \quad (2.8)$$

Despejando (2.7) se obtiene (2.9):

$$g(x) = \ddot{d} = g - \left(\frac{Ki}{md^4} \right) \quad (2.9)$$

Aplicando la serie de Taylor y truncando tenemos:

$$g(i_0, d_0) = g - \left(\frac{Ki_0d_0^{-4}}{m} \right) \quad (2.10)$$

$$\left(\frac{\partial g(i_0, d_0)}{\partial i_0} \right) \Big|_{i_0, d_0} = \frac{-Kd_0^{-4}}{m} \quad (2.11)$$

Finalmente tenemos:

$$\ddot{d} = g - \left(\frac{Ki_0}{md_0^4} \right) - \left(\frac{K}{md_0^4} \right) (i - i_0) + \left(\frac{4Ki_0}{md_0^5} \right) (d - d_0) \quad (2.12)$$

Obteniendo los puntos de equilibrio de (2.7) y (2.6) respectivamente

$$d_0 = \left(\frac{Ki_0}{gm} \right)^{1/4} \quad (2.13)$$

$$i_0 = \left(\frac{v_0}{R} \right) \quad (2.14)$$

2.7. Modelo en espacio de estado

De acuerdo a las ecuaciones del sistema anteriormente descritas se seleccionan las variables de estado como se muestra:

$$x_1(t) = i - i_0 \text{ (Corriente)}$$

$$x_2(t) = \dot{d} - \dot{d}_0 \text{ (Velocidad)}$$

$$x_3(t) = d - d_0 \text{ (Distancia)}$$

La señal de entrada respectivamente es:

$$u = v - v_0 \text{ (Voltaje)}$$

Finalmente se obtiene:

$$\dot{x}_1(t) = -\frac{R}{L}x_1(t) + \frac{1}{L}u \quad (2.15)$$

$$\dot{x}_2(t) = -\frac{K}{md_0^4}x_1(t) + \frac{4Ki_0}{md_0^5}x_3(t) \quad (2.16)$$

$$\dot{x}_3(t) = x_2(t) \quad (2.17)$$

De acuerdo a el modelo en espacio de estados:

$$\dot{x}(t) = Ax(t) + Bu$$

$$y = Cx(t) + Du$$

Tenemos:

$$A = \begin{pmatrix} -\frac{R}{L} & 0 & 0 \\ -\frac{K}{md_0^4} & 0 & \frac{4Ki_0}{md_0^5} \\ 0 & 1 & 0 \end{pmatrix}$$

$$B = \begin{pmatrix} \frac{1}{L} \\ 0 \\ 0 \end{pmatrix}$$

$$C = \begin{pmatrix} 0 & 0 & 1 \end{pmatrix}$$

$$D = \begin{bmatrix} 0 \end{bmatrix}$$

2.8. Modelado de un sistema de levitación magnética de dos grados de libertad

El sistema de posicionamiento magnético de dos grados de libertad utilizado en esta tesis, es el mismo que se presenta en [31], el cual se basa en [32] y [33]. Este sistema consiste en una rejilla móvil que es posicionada por cuatro electroimanes idénticos. El sistema no presenta un acoplamiento entre los dos ejes de acción de la plataforma. Asimismo, este sistema cuenta con todos los sensores necesarios para la medición de posición, velocidad de la rejilla y de las corrientes en los cuatro electroimanes. Los electroimanes utilizados en estos sistemas tienen la característica de presentar una fuerza de atracción de la siguiente forma:

$$F_m(i, d_0) = K_f \left(\frac{i}{d_0} \right)^2 \quad (2.18)$$

Donde:

- F_m es la fuerza electromagnética
- i es la corriente del electroimán
- d_0 es la distancia entre el electroimán y la rejilla
- K_f es la constante de actuación del electroimán.

El modelo matemático de la planta se obtiene a partir de analizar la Figura 2.2. Asimismo, hay que separar las componentes mecánicas y eléctricas del sistema. La parte eléctrica de la planta está conformada por los 4 electroimanes, cada electroimán se modela con el circuito de la Figura 2.3, el cual está formado por una resistencia e inductancia conectadas en serie con una fuente de voltaje.

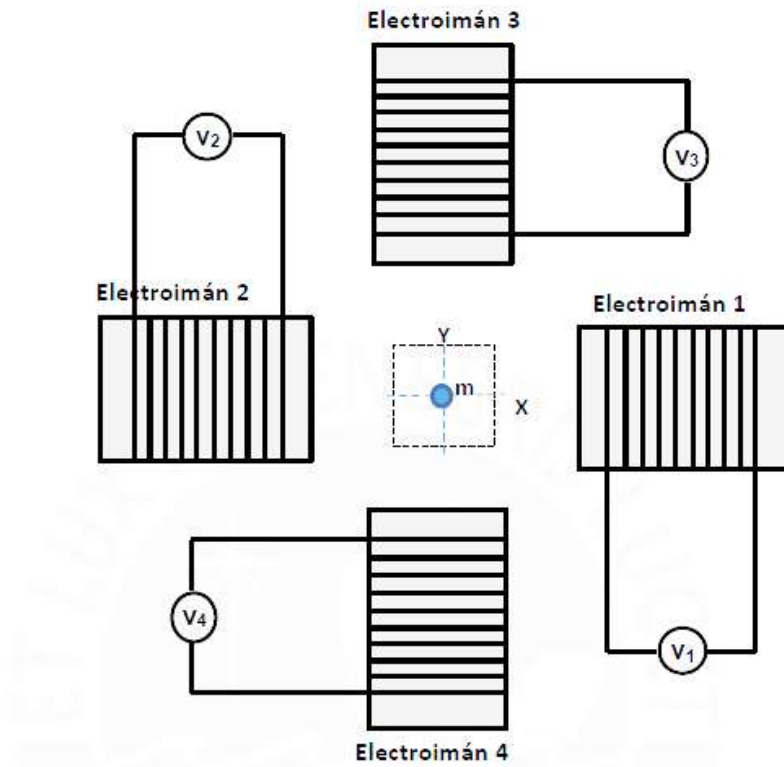


Figura 2.2: Sistema de levitación magnética de dos grados de libertad

La Figura 2.2, muestra dos electroimanes en cada eje con el fin de ampliar el rango de movimiento en cada grado de libertad, además de para su posterior implementación y poder ser utilizado en sistemas de eliminación de fricción con el fin de rotar el objeto levitado.

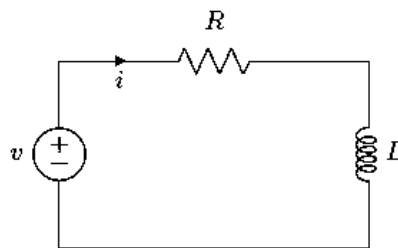


Figura 2.3: Modelo del electroimán.

Considerando V como el voltaje de alimentación de un electroimán, i como la

corriente que fluye por el circuito, R como la resistencia del electroimán y L la inductancia del electroimán, se puede escribir la ley de voltajes de Kirchoff como:

$$V = Ri + L \frac{di(t)}{dt} \quad (2.19)$$

Para la parte mecánica, la planta está compuesta por la rejilla metálica, la cual se modela con una masa puntual " m " que se mueve bajo la acción de las fuerzas electromagnéticas generadas por los electroimanes y por una fuerza de amortiguamiento característica del medio. Adicionalmente, debido a que los ejes X e Y no se encuentran acoplados, cada uno de estos se puede analizar por separado. Aplicando la segunda ley de Newton a la rejilla metálica se puede escribir para el eje X:

$$m\ddot{x} = -\beta\dot{x} - K_f[(\frac{i_1}{d-x})^2 - (\frac{i_2}{d+x})^2] \quad (2.20)$$

Donde i_1 e i_2 son las corrientes de los electroimanes 1 y 2 respectivamente y x es la posición en el eje X de la rejilla metálica. Asimismo β es una constante de amortiguación, K_f es la constante de actuación de los electroimanes y d es la distancia entre dos electroimanes en un mismo eje.

Para simplificar las variables en el modelo se hace uso de los conceptos de voltajes y corriente diferenciales, los cuales se definen de la siguiente manera:

$$\begin{aligned} u &= v_1 - v_2 \\ \Delta i &= i_1 - i_2 \end{aligned} \quad (2.21)$$

Asimismo, se pueden obtener las corrientes y voltajes no diferenciales a partir de sus cantidades diferenciales de la siguiente manera:

$$\begin{aligned} v_1 &= v_0 + \frac{u}{2} \\ v_2 &= v_0 - \frac{u}{2} \end{aligned} \quad (2.22)$$

$$\begin{aligned} i_1 &= i_0 + \frac{x_3}{2} \\ i_2 &= i_0 - \frac{x_3}{2} \end{aligned} \quad (2.23)$$

Donde: i_o y v_o son condiciones iniciales del sistema. El procedimiento descrito se aplica al eje Y con los electroimanes 3 y 4. Finalmente, de este modo se puede formar el modelo espacio estado no lineal de la planta:

$$\dot{x} = f(x, u) \quad (2.24)$$

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -\frac{\beta}{m}x_2 + \frac{k_f}{m}\left[\left(\frac{i_0 + \frac{x_3}{2}}{d - x_1}\right)^2 - \left(\frac{i_0 - \frac{x_3}{2}}{d + x_1}\right)^2\right] \\ \dot{x}_3 &= -\frac{R}{L}x_3 + \frac{1}{L}u \\ \dot{x}_4 &= x_5 \\ \dot{x}_5 &= -\frac{\beta}{m}x_5 + \frac{k_f}{m}\left[\left(\frac{i_0 + \frac{x_6}{2}}{d - x_4}\right)^2 - \left(\frac{i_0 - \frac{x_6}{2}}{d + x_4}\right)^2\right] \\ \dot{x}_6 &= -\frac{R}{L}x_6 + \frac{1}{L}u_2 \end{aligned} \quad (2.25)$$

Los parámetros que caracterizan al sistema se muestran en la Tabla 2.2, así mismo las variables que caracterizan al sistema se muestran en la Tabla 2.1

Tabla 2.1: Parámetros del sistema.

Parámetro	Descripción	Valor
β	Coficiente de amortiguamiento de la rejilla	$0.1 \frac{Ns}{m}$
K_f	Constante de actuación de los electroimanes	$0.49 \frac{Nm^2}{A^2}$
i_0	Corriente nominal de los electroimanes	3 A
v_0	Voltaje nominal de los electroimanes	18 V
$2d$	Distancia entre dos electroimanes de un mismo eje	50 cm
R	Resistencia de los electroimanes	6Ω
L	Inductancia de los electroimanes	0.001 H
m	Masa móvil de prueba	50 g

Tabla 2.2: Variables del sistema.

Variable	Descripción
x_1	Posición (Eje X)
x_2	Velocidad (Eje X)
x_3	Corriente diferencial en los electroimanes (Eje X)
x_4	Posición (Eje Y)
x_5	Velocidad (Eje Y)
x_6	Corriente diferencial en los electroimanes (Eje Y)
u_1	Voltaje diferencial en los electroimanes (Eje X)
u_2	Voltaje diferencial en los electroimanes (Eje Y)

2.8.1. Linealización del sistema

Se linealiza la dinámica no lineal de la planta por una aproximación simple de Taylor de primer orden para obtener un modelo de la forma:

$$\Delta \dot{x} = A\Delta x + B\Delta u \quad (2.26)$$

Donde:

$$A = \frac{\partial f}{\partial x} \quad (2.27)$$

$$B = \frac{\partial f}{\partial u} \quad (2.28)$$

éstas matrices toman los siguientes valores en alrededor del punto de equilibrio:

$$A = \begin{pmatrix} 0 & 1 & 0 \\ \frac{4Kf}{m} & -\frac{\beta}{m} & \frac{2Kf}{m} \\ 0 & 0 & -\frac{R}{L} \end{pmatrix}$$

$$B = \begin{pmatrix} 0 \\ 0 \\ \frac{1}{L} \end{pmatrix}$$

$$C = \begin{pmatrix} 1 & 0 & 0 \end{pmatrix}$$

Del mismo modo se procede para el eje Y, obteniendo exactamente las mismas Matrices A , B y C respectivamente.

2.9. Conclusiones

En este capítulo se presentó el desarrollo del modelo matemático de un sistema de levitación magnética de uno y dos grado de libertad, tipo atractivo usado para suspender un objeto ferromagnético o magnético, el planteamiento matemático en ambos casos se realizó por medio de la segunda ley de Newton y haciendo uso de un diagrama de cuerpo libre.

Capítulo 3

Implementación del sistema de levitación magnética de 1 GDL

3.1. Introducción

En este capítulo se presentan todos los elementos que componen al sistema, así como también la implementación del sistema. La cual abarca desde la creación del electroimán, análisis de los elementos controladores, la caracterización de los sensores, la eliminación del ruido haciendo uso del filtro de Kalman (KF). Además, se presenta la estimación paramétrica del sistema por medio de mínimos cuadrados con y sin filtro (KF), control del sistema por distintas leyes de control y en combinación con el KF implementados en Simulink y con un sistema embebido. Finalmente se hace una estimación del parámetro electromagnético por medio de la recreación de una red neuronal a partir del entrenamiento de esta en Matlab[®] obteniendo sus pesos y bias para su implementación embebida en el microcontrolador.

3.2. Elementos del sistema

En esta sección se presentan todos los elementos utilizados para la implementación del sistema de levitación magnética de un grado de libertad, de acuerdo a la Figura 3.1

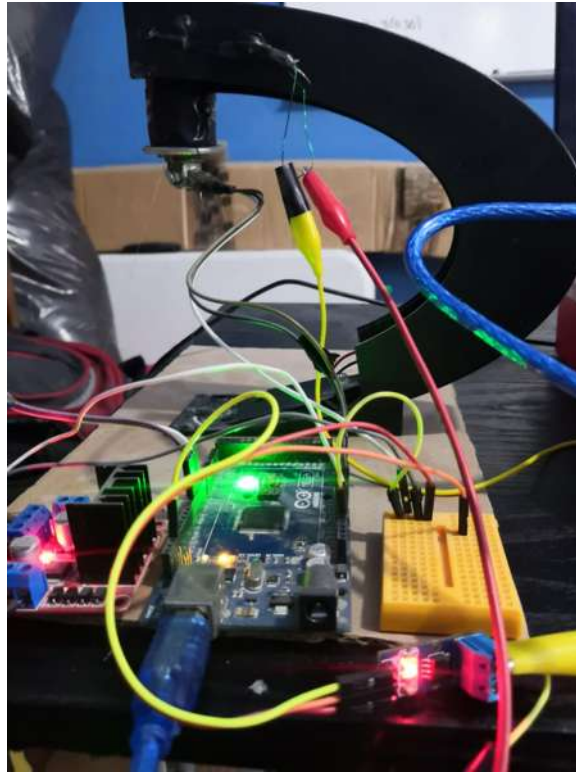


Figura 3.1: Sistema de levitación magnética de 1 GDL

3.2.1. Imán de neodimio

Para el sistema de levitación magnética se hace uso de imanes de neodimio como la masa a levitar, específicamente de imanes de tipo neodimio – hierro – boro, estos son los tipos de imanes que presentan o tienen mayor fuerza magnética.

Los imanes de Neodimio-hierro-boro son sensibles al calor. Si un imán se calienta por encima de su temperatura máxima de funcionamiento ($176^{\circ}F$ u $80^{\circ}C$), se perderá una fracción de su fuerza magnética. Si se calientan por encima de $590^{\circ}F$ ó $310^{\circ}C$, perderán la totalidad de sus propiedades magnéticas. Para el sistema de levitación magnética, las temperaturas de trabajo rondan los $25^{\circ}C$ y sus variantes dentro de $\pm 10^{\circ}C$ son ínfimos, por lo que se desprecian en las ecuaciones.

Para el presente trabajo se usa un imán del tipo N42, el cual fue mostrado en el proceso de modelado en el capítulo anterior. En la Figura 3.2 se muestra las líneas del campo magnético y cómo se comportan.

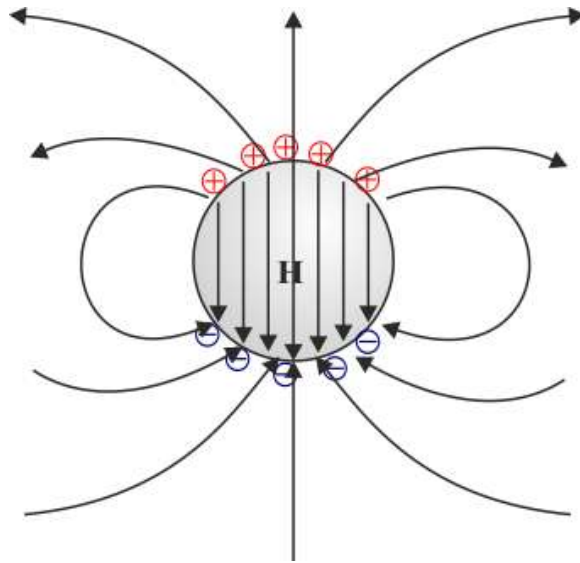


Figura 3.2: Campo magnético imán N42

3.2.2. Diseño del actuador – electroimán

El cable eléctrico es bobinado alrededor de una estructura metálica, la cual por sus propiedades de alta permeabilidad magnética permite concentrar las líneas de campo magnético generado por el electroimán. Cuando la corriente fluye a través de la bobina está se magnetiza. esta fuerza magnética dura solo mientras la electricidad fluye a través de los cables, y su magnitud varía dependiendo de la intensidad de la corriente eléctrica. Los electroimanes pueden generar campos electromagnéticos variables, por lo tanto pueden variar la fuerza atractiva sobre un cuerpo, mientras que los imanes normales tienen un nivel estático de fuerza y no se pueden “apagar”. Se hace uso de un tornillo de media pulgada, así como dos arandelas que ayudan al soporte del alambre magneto, de 0.25mm, como se muestra en la Figura 3.3. El objetivo es dar alrededor dar 3000 vueltas para obtener una fuerza capaz de soportar el peso del imán de neodimio.



Figura 3.3: Elaboración del electroimán

A si mismo se realiza una prueba para determinar si el electroimán es capaz de atraer a la masa desde una distancia considerable, como se muestra en la Figura 3.4. En este aspecto considerando que a menor calibre y mayor cantidad de vueltas se tiene mayor resistencia óhmica; no siendo de tanto interés en este prototipo ya que los parámetros en relación al número de vueltas se muestra en una siguiente sección de manera experimental.



Figura 3.4: Prueba del electroimán

Así mismo se hace la respectiva medición de la resistencia y de la inductancia por medio de un medidor RLC, el cual permite medir la inductancia a una frecuencia alta, obteniendo un un resultado viable de partida, el cual será comparado cuando se realice la estimación paramétrica. En la Figura 3.5 se muestra el resultado obtenido con el uso del medidor RLC.

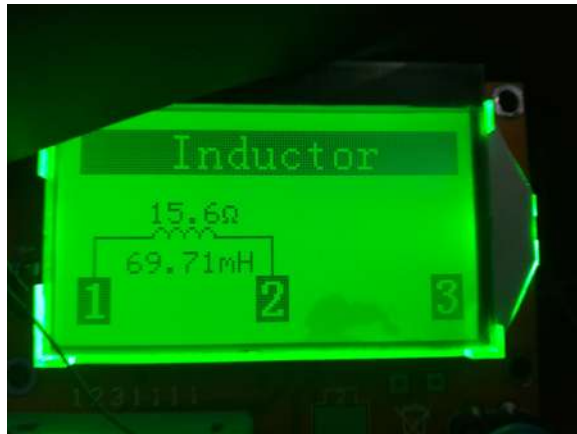


Figura 3.5: Medición de la resistencia e inductancia

3.2.3. Microcontrolador

Existen diferentes microcontroladores, tarjetas de desarrollo o dispositivos que pueden ser utilizados para implementar la ley de control entre estos dispositivos, se pueden utilizar una tarjeta de adquisición de datos DAQ de National Instruments, una FPGA o un Arduino Mega. Al realizar el comparativo entre las tres opciones en cuanto a costo y facilidad de operación se optó por la utilización del Arduino Mega, el cual tiene las siguientes características:

- Microcontrolador ATmega2560
- Voltaje de entrada: 7 a 12V
- 54 pines digitales I/O (14 salidas PWM)
- ADC 10 bits
- PWM 8 bits
- 16 entradas analógicas.
- 256kb de memoria flash.
- Velocidad de reloj 16MHz.

- **Factible:** Las placas Arduino son más accesibles y factibles comparadas con otras plataformas de microcontroladores. **Multi-Plataforma:** El software de Arduino funciona en los sistemas operativos Windows, Macintosh OSX y Linux. La mayoría de los entornos para microcontroladores están limitados a Windows. **Ambiente de programación sencillo y directo:** El ambiente de programación de Arduino es fácil de usar para los usuarios, Arduino está basado en el entorno de programación de processing.
- **Software ampliable y de código abierto:** El software Arduino está publicado bajo una licencia libre y preparada para ser ampliado por programadores y desarrolladores experimentados. El lenguaje puede ampliarse a través de librerías de C++ y modificarlo a través del lenguaje de programación AVR C en el que está diseñado.
- **Hardware ampliable y de código abierto:** Arduino está basado en los microcontroladores ATMEGA168, ATMEGA328 y ATMEGA1280. Los planos de los módulos están publicados bajo licencia creative commons, por lo que los diseñadores de circuitos pueden hacer su propia versión del módulo, ampliándolo u optimizándolo.

3.2.4. Control de electroimán

Para el control del electroimán se hace uso de un puente H, el cual es un circuito electrónico que generalmente se usa para permitir a un motor eléctrico de corriente directa (DC) girar en ambos sentidos, avance y retroceso. Son ampliamente usados en robótica y como convertidores de potencia. Los puentes H están disponibles como circuitos integrados, pero también pueden construirse a partir de componentes discretos. La Figura 3.6 muestra la estructura del puente H, con sus cuatro interruptores (S1,S2,S3,S4) y una fuente de voltaje de entrada (V_{in}), así como el motor de corriente DC (M).

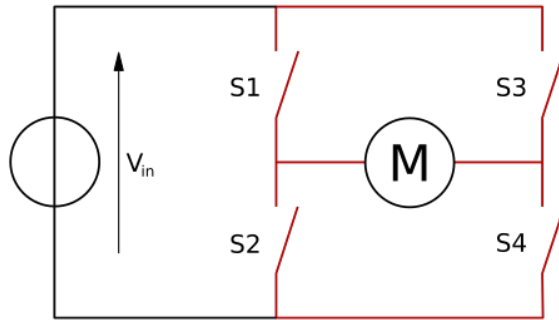
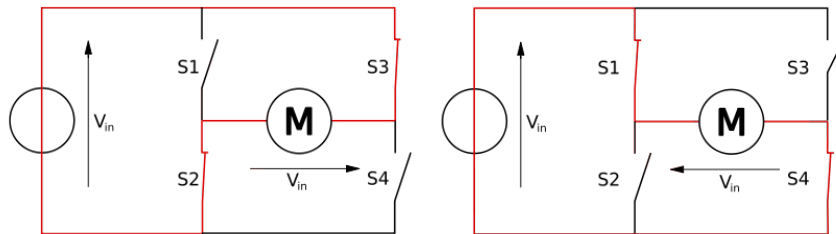


Figura 3.6: Estructura del puente H

El término "puente H" proviene de la típica representación gráfica del circuito. Un puente H se construye con 4 interruptores (mecánicos o mediante transistores). Cuando los interruptores S1 y S4 (Figura 3.7) están cerrados (y S2 y S3 abiertos) se aplica una tensión positiva en el motor, haciéndolo girar en un sentido. Abriendo los interruptores S1 y S4 (y cerrando S2 y S3), el voltaje se invierte, permitiendo el giro en sentido inverso del motor.



(a) *Aplicación de un voltaje positivo.* (b) *Aplicación de un voltaje invertido.*

Figura 3.7: Puente H sentido de giro.

En este caso no se hace uso de un motor, sino del electroimán, el cual es una bobina la cual dependiendo del sentido de la corriente se obtiene la diferente polaridad de "Norte" y "Sur" en este, lo que permite a que finalmente la masa magnética pueda ser controlada a conveniencia, intercambiando la polaridad del electroimán lo cual afectara directamente a la masa siendo atraída o repelida respectivamente.

3.2.5. Puente H L298N

El puente H L298N es un módulo que permite controlar dos canales, así como el sentido de corriente y su intensidad de hasta 2 amperios. El módulo cuenta con todos los componentes necesarios para funcionar sin necesidad de elementos adicionales, entre ellos diodos de protección y un regulador LM7805 que suministra 5V a la parte lógica del integrado L298N. Cuenta con jumpers de selección para habilitar cada una de las salidas del módulo (A y B). La salida A está conformada por OUT1 y OUT2 y la salida B por OUT3 y OUT4. Los pines de habilitación son ENA y ENB respectivamente. En la Figura 3.8 se muestra el driver utilizado.

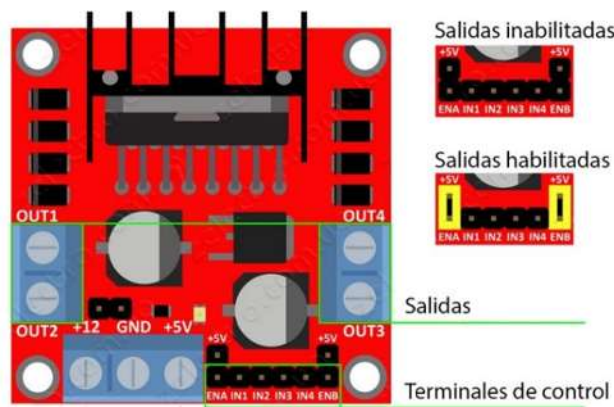


Figura 3.8: Driver L298N

- Conexión de alimentación

Este módulo se puede alimentar de 2 maneras gracias al regulador integrado LM7805.

Cuando el jumper de selección de 5V se encuentra activo, el módulo permite una alimentación de entre 6V a 12V DC. Como el regulador se encuentra activo, el pin marcado como +5V tendrá un voltaje de 5V DC.

Cuando el jumper de selección de 5V se encuentra inactivo, el módulo permite una alimentación de entre 12V a 35V DC. Como el regulador no está funcionando, se tiene que conectar el pin de +5V a una tensión de 5V para alimentar la parte lógica del L298N. Usualmente esta tensión es la misma de la parte de control.

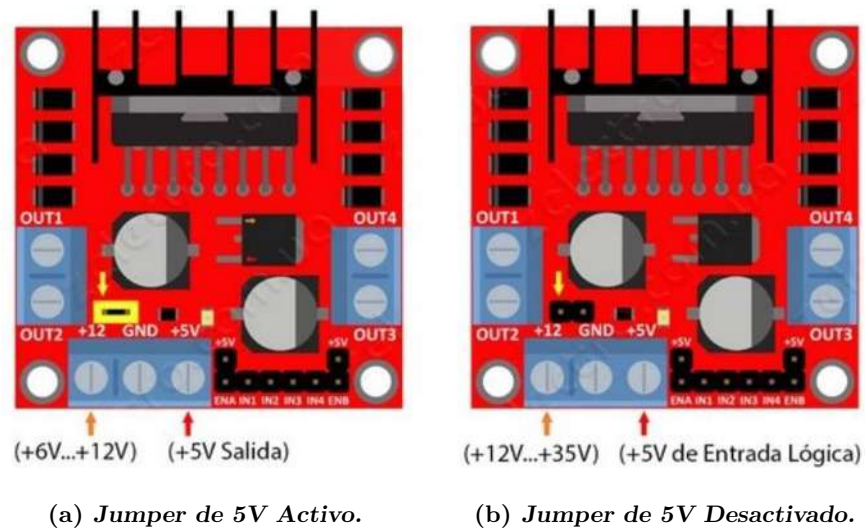


Figura 3.9: Conexión de alimentación L298N.

Como se observa en el módulo, las entradas Enable son las que permiten el control de la intensidad de la corriente en la bobina, la cual controlamos por medio del PWM del arduino, el cual es de 8 bits.

3.2.6. Sensor de efecto hall

Para poder determinar la posición de la masa (la cual posee propiedades magnéticas) se hace uso de un sensor de efecto hall analógico y lineal, el cual según el comportamiento y dirección de los vectores del campo de las fuerzas magnéticas, proporciona la lectura de la posición de la masa con respecto al electroimán. Se hace uso del sensor 49E el cual es un sensor de efecto Hall lineal económico, este es dispositivo pequeño y versátil que es operado por el campo magnético desde un imán permanente o un electroimán. La tensión de salida de la fuente se establece por la tensión de alimentación y varía en proporción a la fuerza de la fuente del campo magnético. El circuito integrado presenta una salida de ruido baja, lo que hace que no sea necesario utilizar un filtrado externo. También incluye resistencias de película delgada para proporcionar una mayor estabilidad de la temperatura y exactitud. Este sensor lineal de efecto Hall tiene un rango de temperatura de operación de -40°C a 100°C , apropiado para entornos comerciales de consumo e industriales.

Características:

- lineal y alta sensibilidad: 1.4 3.0 mV / GS
- Construcción en miniatura.
- Consumo de energía de 4 mA a 5 Vcc para eficiencia energética
- Funcionamiento en baja tensión hasta 2.3 Vcc.
- Salida de fuente de corriente única.
- Salida lineal para flexibilidad de diseño de circuito.
- El bajo nivel de ruido elimina virtualmente la necesidad de filtrado
- Rango de temperatura de $-40^{\circ}C$ a $100^{\circ}C$
- Responde a gauss positivo o negativo.

El sensor usa una técnica de compensación de desplazamiento dinámico con un reloj de alta frecuencia interna, lo cual reduce el desplazamiento residual de tensión. El reloj de alta frecuencia permite una mayor velocidad de muestreo, lo que se traduce en mayor precisión y capacidad de procesamiento de la señal más rápido. esta técnica produce que los dispositivos cuenten con una salida de tensión estable. En la Figura 3.10 se muestra la estructura interna del sensor.

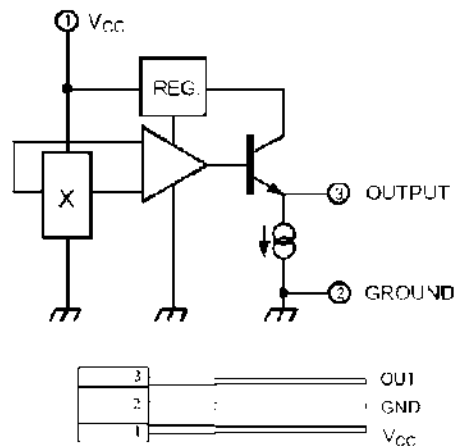


Figura 3.10: Estructura interna y pinout del sensor de efecto hall.

La Tabla 3.1 muestra la distribución de pines y los valores reales del sensor de efecto hall.

Tabla 3.1: Pinout y valores del sensor de efecto hall

Término	Número de terminal	Descripción	Implementación
Vcc	1	Voltaje de alimentación	4.89 v
Gnd	2	Tierra	0v
Out	3	Voltaje de salida de sensor	(0v-4.89v) en 10 bits

3.2.7. Lectura de posición

Como se muestra en la Tabla 3.1 específicamente en la terminal número 3 del sensor la cual es la señal analógica de salida del sensor, se obtendrá un rango de lecturas desde 0 a 4.89V en un proceso de cuantización en un rango de 0 a 1023 valores presentes en el ADC (Conversor analógico digital) del microcontrolador. La resolución del ADC limita la implementación, esto debido a la cantidad de valores procesados a partir de cuales se obtienen las ecuaciones que caracterizan la respuesta del sensor, la cual es implementada en el microcontrolador, estas lecturas se muestran en la Tabla 3.2.

Tabla 3.2: Lectura ADC- distancia

ADC Valor	Distancia CM
873	0
858	0.5
810	1
764	1.5
753	2

La gráfica de tendencia (Figura 3.11), muestra la obtención de los datos donde se utiliza una aproximación lineal de las lecturas reales del sensor las cuales no son del todo lineales, tal y como lo especificó, el fabricante, está aproximación se realiza con la ecuación punto pendiente, directamente en Excel ® ® . Se podría aproximar a cualquier orden “N” deseado sin embargo en este caso se utiliza el de primer orden, ya que se observa que se obtiene una buena aproximación.

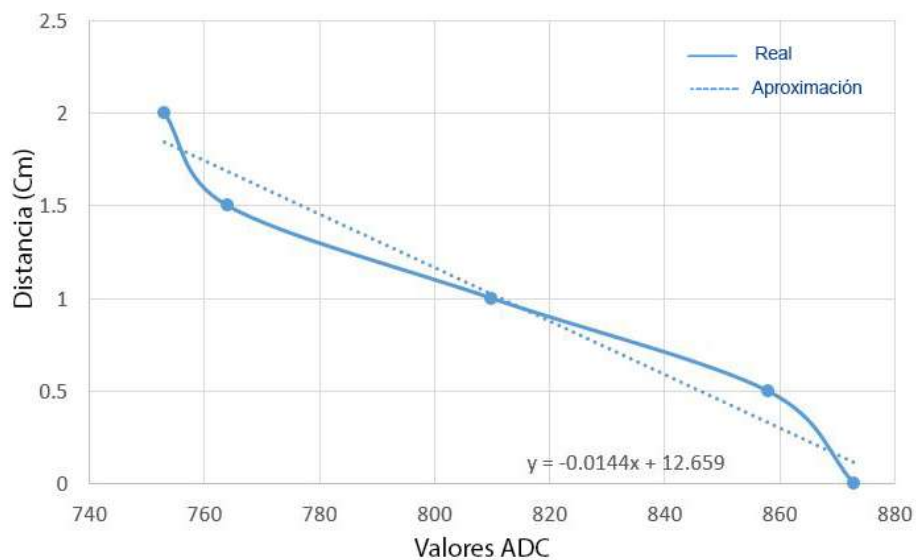


Figura 3.11: ADC valores vs distancia.

Usando está información se obtiene la ecuación característica del sensor de efecto Hall representada por la Ecuación (3.1).

$$Dis = -0.0144(Val) + 12.659 \quad (3.1)$$

Donde se obtendrá la distancia (Dis) en (cm) como se requiere en el modelo, así mismo la variable "Val" la cual corresponde a los valores del voltaje leídos por el ADC provenientes del sensor.

3.2.8. Sensor de corriente

El sensor de corriente ACS712 es una solución económica para medir corriente. Internamente trabaja con un sensor de efecto Hall que detecta el campo magnético que se produce por inducción de la corriente que circula por la línea que se está midiendo. El sensor proporciona una salida de voltaje proporcional a la corriente, dependiendo de la aplicación puede usarse el ACS712-05A, ACS712-20A o el ACS712-30A, para rangos de 5, 20 o 30 amperios, respectivamente.

El ACS712 puede utilizarse en módulos, los cuales facilitan su conexión, ya que cuentan con una bornera para conectar la línea que se desea medir y 3 pines, dos para conectar la alimentación y un pin para la salida analógica. En la Figura 3.12 se muestra el modulo utilizado.



Figura 3.12: Sensor de corriente ACS712.

El rango de corriente que puede medir y la sensibilidad varían dependiendo del

modelo del integrado, existen tres modelos, los cuales son detallados en la Tabla 3.3.

Tabla 3.3: Modelos sensor de corriente

Modelo	Rango	Sensibilidad
ACS712ELCTR-05B-T	-5 a 5 A	185 mV/A
ACS712ELCTR-20A-T	-20 a 20 A	100 mV/A
ACS712ELCTR-30A-T	-30 a 30 A	66 mV/A

Todos los modelos usan la misma configuración de Pinout como se muestra en la Tabla 3.4.

Tabla 3.4: PinOut sensor de corriente

Termino	Número de terminal	Descripción
Gnd	1	Tierra
Vout	2	Voltaje de salida del Sensor
Vcc	3	Voltaje de Alimentación

Lectura de corriente

De la misma manera que se tiene en la lectura de la posición y como se muestra en la Tabla 3.4 específicamente en la terminal número 2 del sensor la cual es la señal analógica de salida del sensor, se obtiene un rango de lecturas desde 0A a 0.63A en un proceso de cuantización en una lectura de voltajes provenientes del sensor el rango de 0 a 1023 valores presentes en el ADC (Conversor analógico digital) del microcontrolador, pero a su vez estás lecturas de amperaje están determinadas por la tensión máxima de 12.29V aplicada sobre el electroimán la cual está regulada por medio de PWM el cual es de 8 Bits es decir en un rango de 0 a 255 valores, esto nos hace estar limitados por esta resolución. Al hacer uso del sensor se obtienen los valores de la Tabla 3.5, estos valores se utilizan para obtener la Ecuación que caracteriza dicho comportamiento y se implementa en el microcontrolador.

Tabla 3.5: Lectura voltaje vs amperes

Voltaje	Amperes
2.49	0
2.48	0.02
2.47	0.05
2.46	0.11
2.41	0.36
2.36	0.65

La gráfica de tendencia mostrada en la Figura 3.13 se obtiene usando una aproximación lineal de las lecturas reales del sensor, esta aproximación se hace con la Ecuación punto pendiente directamente en Excel [®]. Se podría aproximar a cualquier orden “N” deseado, pero en este caso se utiliza el de primer orden, debido a que se observa que se obtiene una buena aproximación.

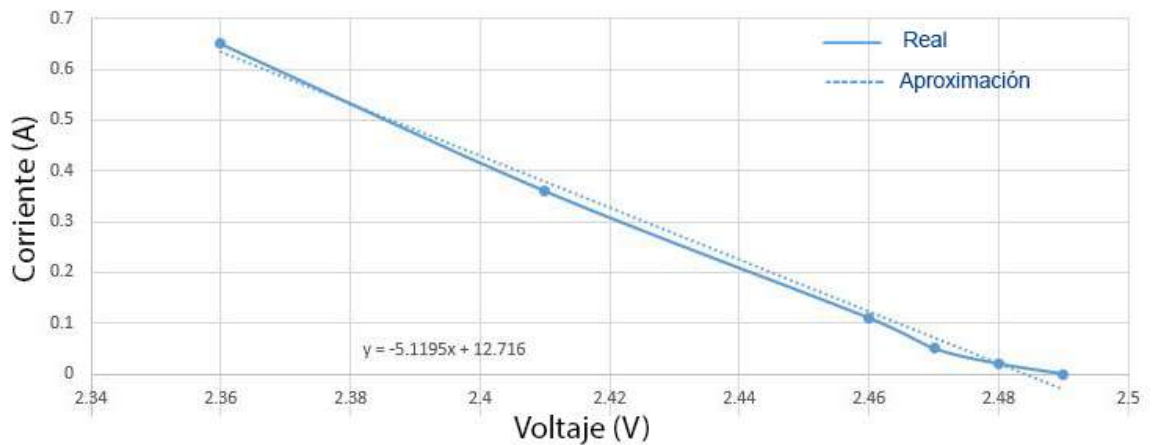


Figura 3.13: Dinámica voltaje vs corriente del sensor de corriente.

Usando esta información se obtiene la Ecuación característica del sensor de corriente representada por la Ecuación (3.2).

$$Amp = -5.1195(Voltaje) + 12.716 \quad (3.2)$$

Donde se obtiene la corriente en (Amperes) como se requiere en el modelo así

mismo la variable “Voltaje”, estos los valores del voltaje leídos y convertidos por el ADC que provienen del sensor.

3.3. Obtención del parámetro K

El valor numérico del parámetro K de la Ecuación (2.5) puede ser obtenido de forma teórica si se tienen absolutamente todos los parámetros del electroimán como se muestra en la obtención de la Ecuación de la fuerza del electroimán sobre la masa-imán [36]. Ya que para cada forma de la masa a levitar se produce al realizar un análisis vectorial de fuerzas. En este caso se utiliza un imán de Neodimio N42.

Para poder obtener el parámetro K se debe buscar el punto de equilibrio del sistema donde se igualen las fuerzas mostradas en la Ecuación (2.1) y (2.5), de este modo la ecuación de equilibrio resultante está dada por la Ecuación (3.3).

$$K \left(\frac{i}{d^4} \right) = mg \quad (3.3)$$

Hay infinidad de puntos de equilibrio, basta con encontrar una U_{eq} , lo cual es extremadamente difícil.

Datos:

- $m = 58.7 \text{ g}$
- $g = 9.81 \text{ m/s}^2$
- $d_{eq} = 1.8 \text{ cm}$
- $i_{eq} = 365.1 \text{ mA}$

De (3.3) tenemos:

$$\begin{aligned}
K &= \frac{mgd^4}{i} \\
&= \frac{(0.0587kg) (9.81m/s^2) (0.018m)^4}{0.3651A} \\
&= 1.65571 \times 10^{-7} \text{ kg} \cdot \text{m}^5 \cdot \text{s}^{-2} \cdot \text{A}^{-1}
\end{aligned}$$

3.4. Estimación paramétrica

El método de mínimos cuadrados es una técnica básica para la estimación de parámetros. El método es simple de aplicar, cuando el sistema a identificar es lineal respecto a los parámetros. En este caso los parámetros a identificar son la resistencia (R) y la inductancia (L) del electroimán, ya que la masa es un parámetro fácilmente medible sin un proceso tan elaborado de tratamiento y procesamiento de señales para obtenerlo.

3.4.1. Método de mínimos cuadrados

Karl Friedrich Gauss planteó el principio de mínimos cuadrados al final del siglo XVIII, e inicialmente fue utilizado para determinar órbitas de planetas y asteroides. El modelo matemático hipotético que puede utilizarse para describir los mínimos cuadrados puede se escribe de la forma:

$$y(i) = \varphi_1(i)\theta_1^0 + \varphi_2(i)\theta_2^0 + \dots + \varphi_n(i)\theta_n^0 = \varphi(i)^T \theta^0$$

Donde:

- y es la variable observada.
- $\theta_1^0 \dots \theta_n^0$ son los parámetros del modelo a ser determinados.
- $\varphi_1(i) \dots \varphi_n(i)$ son funciones conocidas.

Las variables φ se llaman variables de regresión, y $\varphi(i)^T$ es el regresor.

3.4.2. Procedimiento de estimación de R y L

De acuerdo a las ecuaciones diferenciales (2.15), (2.16), y (2.17), las cuales describen el comportamiento del sistema, en este caso los parámetros a identificar son la resistencia y la inductancia del electroimán, los cuales están presentes en la ecuación diferencial (2.15), la cual puede ser escrita de la siguiente forma:

$$\frac{di}{dt} = -\frac{R}{L}i + \frac{1}{L}u \quad (3.4)$$

Planteando (3.4) en la forma de mínimos cuadrados:

$$\dot{i} = -b1 * i + b2 * u \quad (3.5)$$

donde:

- i = corriente del electroimán
- u = voltaje del electroimán
- $b1 = \frac{R}{L}$
- $b2 = \frac{1}{L}$

El modelo anterior puede ser escrito como:

$$y(i) = \varphi(i)^T \theta$$

donde:

$$\varphi(i)^T = [i \quad u] \quad (3.6)$$

y

$$\theta^T = [b1 \quad b2] \quad (3.7)$$

donde: b_1, b_2 , son los parámetros a estimar. Así, los parámetros estimados están dados por:

$$\hat{\theta} = [\Phi^T \Phi]^{-1} \Phi^T Y \quad (3.8)$$

En este caso de estudio se obtiene:

$$\hat{\theta} = [\varphi_1^T \varphi_1]^{-1} \varphi_1^T (\dot{i}) \quad (3.9)$$

3.4.3. Obtención de los vectores

Para obtener los vectores como se observa en la Ecuación (3.5), se utiliza una señal tipo rampa de entrada ya que al ser un sistema muy rápido, usar una señal senoidal sería extremadamente difícil, además de las conversiones internas de valores y uso de las ecuaciones descritas en las secciones anteriores, se procede a realizar la identificación como se muestra en el diagrama de la Figura 3.14. Esto se realiza de forma embebida en el microcontrolador.

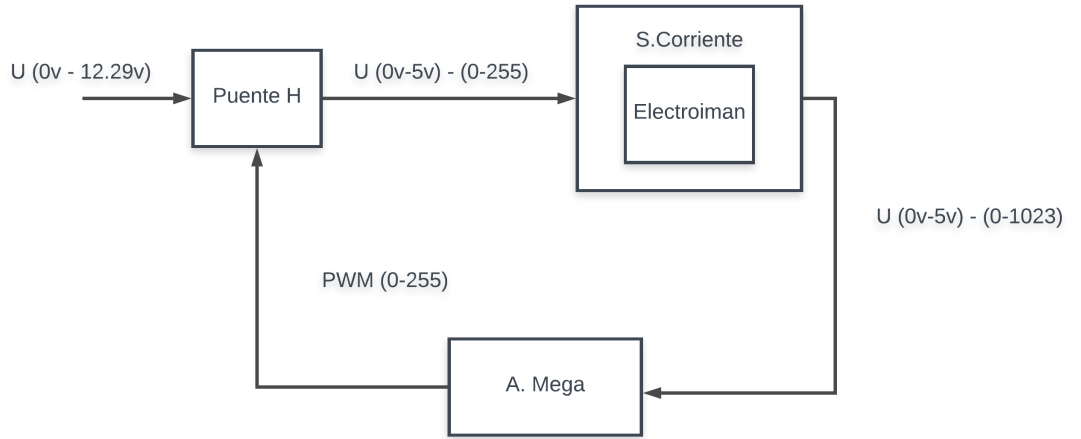


Figura 3.14: Diagrama de la identificación del sistema embebida.

Donde la u de entrada varía en un rango de 0 a 12.29V donde se alimentará al puente H el cual convertirá de 0 a 5V cuantizado en 8 bits, señal que se alimentará al electroimán por medio de PWM y se hará el sensado de esta corriente por medio del sensor

de corriente; donde estos valores están cuantizados por el ADC de 10 bits del Arduino Mega el cual cerrara el ciclo mandando la acción PWM al puente H cuantizado en 8 bits.

Así, la Figura 3.15 muestra la señal de entrada tipo rampa aplicada en la entrada del puente H.

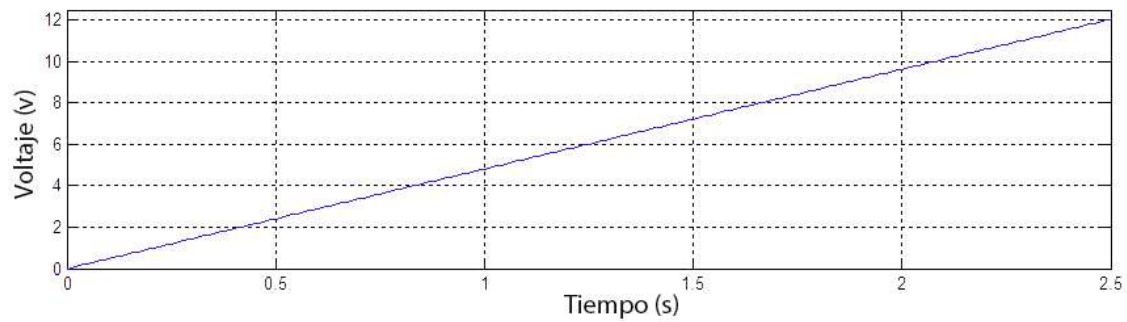


Figura 3.15: Señal de entrada-voltaje

La corriente y la derivada de la corriente son mostradas en la Figura 3.16 y 3.17, respectivamente.

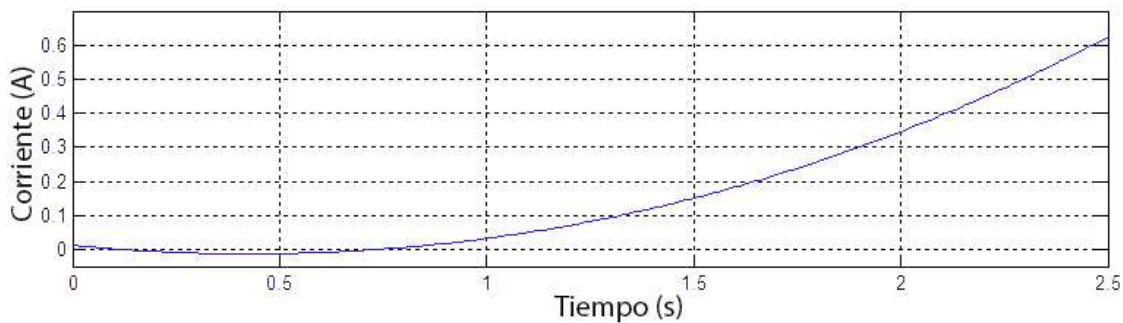


Figura 3.16: estado x1-corriente

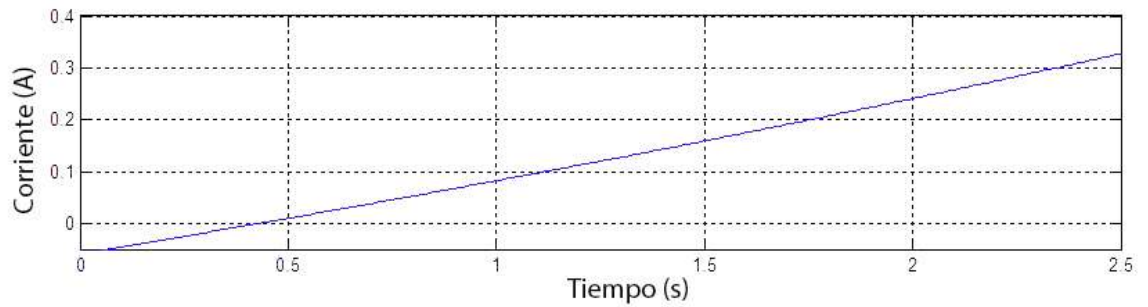


Figura 3.17: Derivada corriente

En el Apéndice B se muestran los resultados muestra a muestra, siendo un total de 256 debido al conversor analógico-digital de 8 bits presente en el arduino.

A primera instancia no se observa la presencia de ruido debido a la mínima cantidad de muestras “256” pero es evidente los errores en las mediciones de la corriente como se observa con corrientes negativas ante presencia de voltaje positivo, esto es debido a el modelado de la dinámica del sensor de corriente como se muestra en una sección previa de la obtención de la Ecuación característica de este.

3.4.4. Resultados de identificación

De acuerdo a la Ecuación (3.5) se obtienen los parámetros de la Tabla 3.6

Tabla 3.6: Resultados de identificación

Parámetro	Valore Real	Valor Estimado
R	15.6 Ohms	17.4114 Ohms
L	69.71 mH	63.7235 mH

Se observa claramente un error en la estimación, esto es debido a la cantidad limitada de muestras, ruido en las muestras, mala caracterización de los sensores. Estos factores son difíciles de eliminar ya que siempre estarán presentes tanto en limitaciones físicas como el ADC del microcontrolador así como la no linealidad presente en los sensores. Una de las características que pueden ser mejoradas para la estimación, corresponde a la

reducción del ruido en las muestras. Utilizando el filtro de Kalman el ruido presente puede ser atenuado.

3.5. filtro de Kalman

El filtro de Kalman (KF) es una potente herramienta matemática que juega un importante papel cuando se incluyen medidas del mundo real en el sistema con el que se trabaja. Fue inventado por Rudolph Emil Kalman a finales de la década de 1950, con la finalidad de filtrar y predecir sistemas lineales.

El filtro de Kalman, básicamente es un conjunto de ecuaciones matemáticas que implementan un estimador óptimo del tipo predictor-corrector. Procesa todas las medidas disponibles, sin importar su precisión, para estimar el valor actual de las variables de interés. Esto es posible gracias a:

- El conocimiento del sistema y dispositivos dinámicos de medida.
- A la descripción estadística de los ruidos del sistema, errores de medida e incertidumbre en los modelos dinámicos.
- A cualquier información disponible acerca de las variables de interés.

El KF es un algoritmo de procesamiento de datos, y no es más que un programa de ordenador. Por ello, se tiene que trabajar sobre medidas discretas en vez de trabajar en tiempo continuo. Además, al ser un algoritmo recursivo, no requiere ir almacenando todos los datos previos para procesarlos de nuevo cada vez que se tome una nueva medida. La Figura 3.18 muestra las etapas para la aplicación del KF

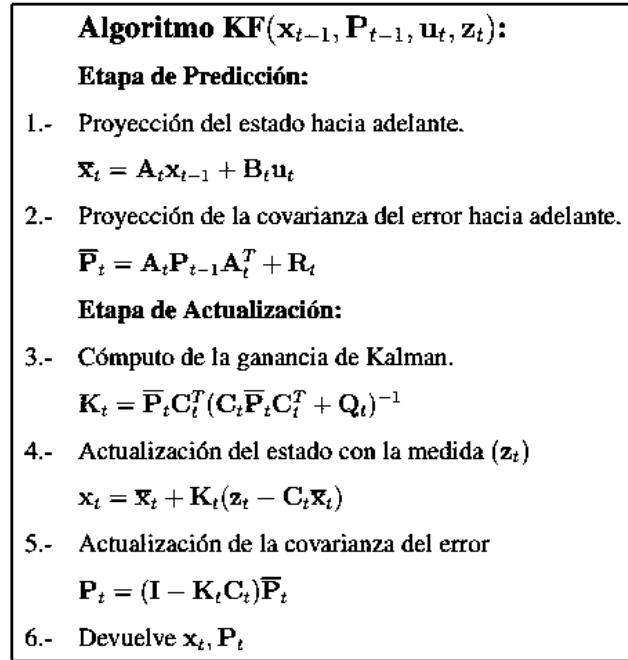


Figura 3.18: Algoritmo KF

El KF representa una creencia o confianza en el estado x_t en el tiempo t que viene dada por la media, x_t y la covarianza, P_t . La entrada que recibe el KF es la creencia en el tiempo $t - 1$, representada por x_{t-1} y P_{t-1} . Para actualizar dicha creencia, el KF requiere también las señales de control (u_t) y las observaciones del entorno que proporcionan los sensores (z_t). La salida del KF sería la creencia en el instante de tiempo t , representada por x_t y P_t

- A_t Es la matriz (nxn) que relaciona el estado en el instante $t - 1$ con el estado en el instante t , en ausencia de señales de control.
- B_t Es la matriz (n x l) que relaciona las señales de control, con el estado actual.
- R_t Es la matriz (nxn) que representa a la covarianza del ruido del proceso.
- C_t Es la matriz (mxn) que relaciona el estado actual con las observaciones del entorno.
- Q_t Es la matriz (nxn) que representa a la covarianza del ruido de las observaciones.

- Kt Es la matriz (nxm) que representa a la ganancia de Kalman. La ganancia de Kalman indica la confianza en las características observadas, empleando la incertidumbre de las mismas junto con una medida de la calidad de los datos proporcionados

3.5.1. Implementación del filtro de Kalman

La implementación de este algoritmo es llevada a cabo de manera embebida en el microcontrolador con el fin de reducir o atenuar el ruido presente en los sensores y realizar una mejor estimación paramétrica y un mejor control.

A continuación en la Figura 3.19 y 3.20 se presenta al KF en tiempo real estimando el valor del sensor de efecto hall el cual es el que proporciona la posición de la masa. En la Figura 3.19 se muestran los resultados obtenidos para la masa estática, lo cual permite observar las fluctuaciones presentes por el ruido en la medición del sensor. Para la masa en movimiento o excitada se observa una gran mejora como se presenta en la Figura 3.20

El código de implementación es mostrado en el Apéndice B.

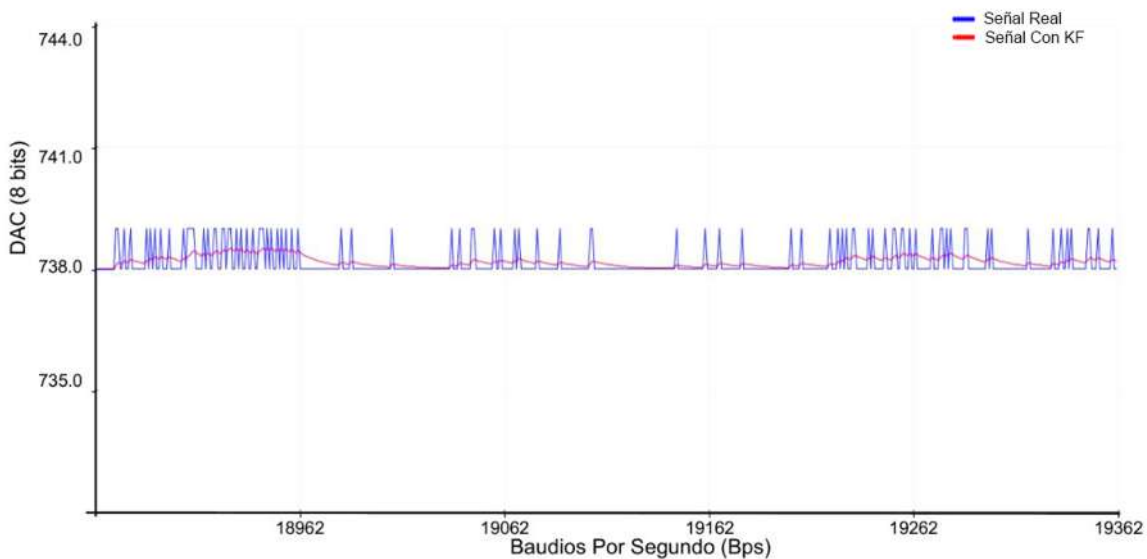


Figura 3.19: Implementación del filtro de Kalman con una masa estática

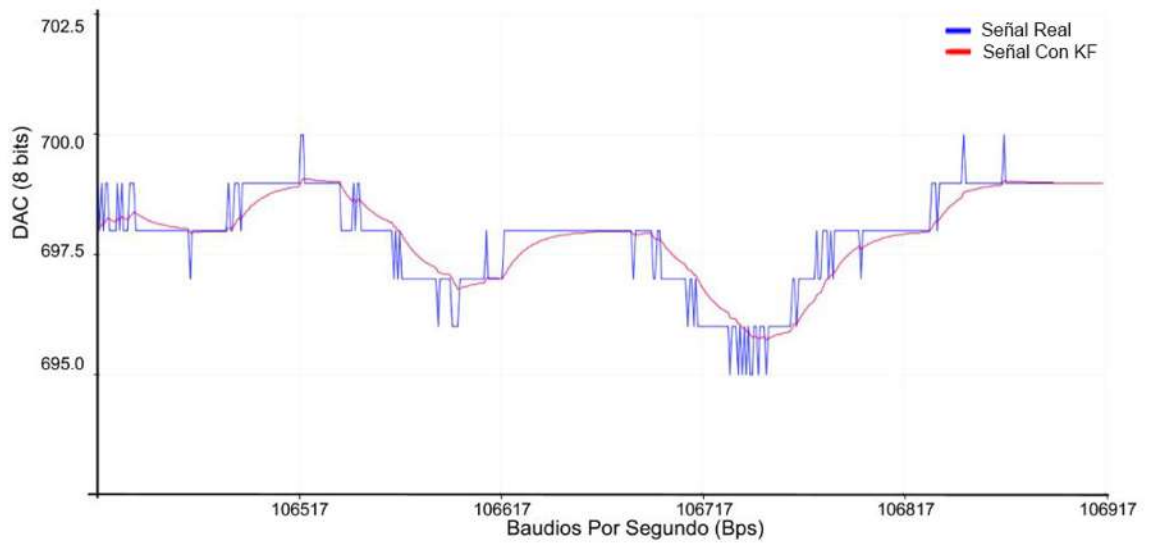


Figura 3.20: Implementación del filtro de Kalman con una masa en movimiento

En color azul se observa las fluctuaciones del sensor las cuales son las principales causas de error en la estimación así como posteriormente en la etapa de control, se observa en color rojo la estimación que hace el filtro de Kalman eliminando totalmente el ruido.

3.5.2. Obtención de los vectores adicionando filtro de Kalman

Como se muestra en la sección 3.4.3, se procede de la misma manera pero adicionando el filtro de Kalman a cada lectura proveniente de los sensores como se muestra en el diagrama de la Figura 3.21.

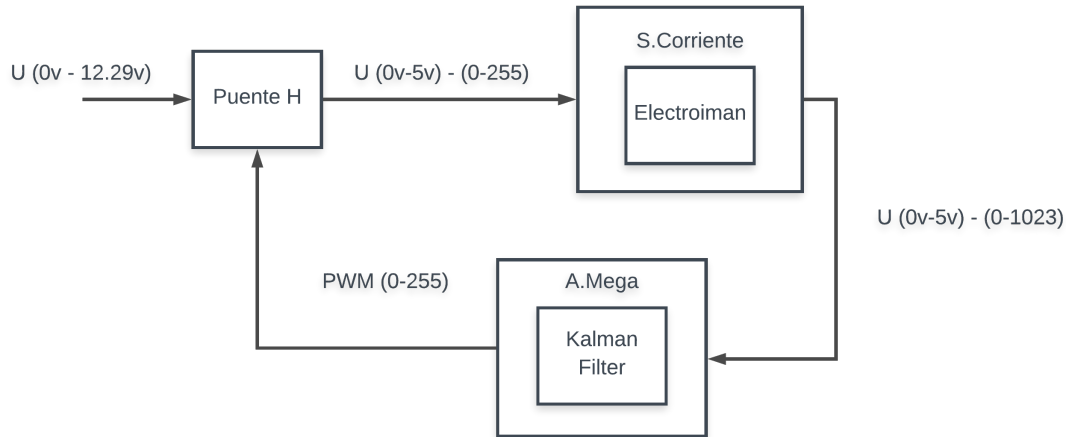


Figura 3.21: Identificación del sistema con el filtro de Kalman

En el Apéndice A se muestran los resultados muestra a muestra, siendo un total de 256 debido al convertor analógico-digital de 8 bits presente en el arduino.

3.5.3. Resultados de identificación utilizando el filtro de Kalman

De acuerdo a la Ecuación (3.5) se obtienen los parámetros de la Tabla 3.7

Tabla 3.7: Resultados de identificación con filtro de Kalman

Parámetro	Valore Real	Valor Estimado
R	15.6 Ohms	15.1770 Ohms
L	69.71 mH	66.7712 mH

Se observa una mejoría importante, ya que los valores estimados se aproximan más a los valores reales; en los cuales hay que tener en cuenta que el mismo tester que se usó puede tener variaciones, específicamente en el valor de la inductancia el cual es un parámetro difícil de medir ya que es no lineal y depende de la frecuencia a la cual sea medida. El valor de la resistencia se logra estimar en mayor medida gracias a que está al ser lineal, pero sus cambios en valor de resistencia son mucho menores y de mayor confiabilidad en la medición.

Finalmente con esto se logró una confirmación del modelo matemático, es decir

que las ecuaciones diferenciales planteadas realmente concuerdan con el fenómeno físico presente, de este modo; puede ser aplicado el control al sistema.

3.6. Control por modos deslizantes de primer orden

A lo largo de la historia en la teoría de control automático, la investigación de acciones de control sobre sistemas ha mantenido en un aspecto relevante, en el cual existen cantidad de técnicas tanto basadas en el modelo matemático y aquellas que no dependen de este, como se ha observado que la planta en el sistema de levitación magnética es demasiado rápida lo cual obliga a que la acción de control debe serlo también por lo tanto se utiliza un tipo de control llamado modos deslizantes. En el control por modos deslizantes la acción de control es equivalente a un interruptor (on/off) de alta frecuencia.

El estudio de este tipo de controladores abarca una amplia extensión de áreas, desde problemas matemáticos, hasta aspectos en aplicaciones de la vida cotidiana como en nuestro caso al tener una implementación directa.

El control con modos deslizantes, tiene la capacidad de ser una herramienta eficiente, para el control complejo de alto orden de plantas dinámicas no lineales que operan bajo condiciones inciertas, un problema común para varios sistemas actuales. Esto explica el alto nivel de la actividad en la investigación del área durante las últimas dos décadas. El control por modos deslizantes consiste en definir una ley de control que conmuta a alta frecuencia la cual consigue llevar el estado del sistema a una superficie denominada superficie de deslizamiento y una vez en ella mantenerlo ante posibles perturbaciones externas. La señal de control de los modos deslizantes de primer orden está definida por: (3.10).

$$u = k_1 \text{sign}(s) \quad (3.10)$$

donde:

- s = Superficie deslizante.
- k_1 = Ganancia
- $\text{sign}(s) = \begin{cases} 1 & \text{si } s > 0 \\ 0 & \text{si } s = 0 \\ -1 & \text{si } s < 0 \end{cases}$

Para este sistema, la superficie deslizante seleccionada es $s = e$ es decir (superficie deslizante = Posición de referencia - Posición medida) el error de la posición de la masa respecto al sensor de efecto hall.

La acción de control debe conmutar a una alta frecuencia entre valores positivos y negativos esto con el objeto de que el sistema converja a la superficie deslizante en cuestión. Debido a esto se produce un efecto indeseado llamado “chattering”, es decir que las trayectorias en lugar de deslizarse, oscilan alrededor de la superficie de deslizamiento. La presencia de chattering puede excitar dinámicas no modeladas, las cuales pueden reducir la exactitud del control e introducir grandes pérdidas por calor en drivers de potencia.

Por lo tanto es necesario “suavizar” la señal de control discontinua. Varios enfoques han sido propuestos para reducir el efecto del chattering, entre ellos el más conocido es la aplicación de la función saturación para el control de la ganancia cuando el sistema se encuentra cerca de la región de deslizamiento, con este enfoque se logra reducir el efecto del chattering pero existirá un error finito de régimen permanente, a pesar de esto se consigue que las dinámicas del sistema permanezcan en la superficie deslizante. Se debe mencionar que la superficie deslizante no depende de los parámetros de la planta ni perturbaciones, y como ventaja se puede considerar insensibilidad a variaciones paramétricas o perturbaciones, por lo que es considerado un esquema de control robusto.

3.6.1. Consideraciones de implementación

Para el sistema de levitación magnética la acción de control debe ser rápida, el tiempo de comunicación debe ser pequeño por lo cual obtener gráficas en tiempo real es bastante complicado. Por este motivo se realizó la implementación del algoritmo de control de dos formas distintas: control mediante Matlab ® como puente de enlace al microcontrolador, es decir en este caso se usa como una tarjeta de adquisición de datos (DAC) y Matlab ® Simulink realiza los cálculos para el control y Control embebido en el microcontrolador.

Para el primer caso se usó Matlab ® - Microcontrolador, se usó una comunicación de 115200 baudios por segundo, máxima que Matlab ® permite, existe un gran problema de implementación debido a que si bien es una velocidad bastante alta, Matlab ® tiene un protocolo interno de comunicación el cual afecta directamente al microcontrolador gastando demasiado tiempo de procesamiento lo cual hace bastante más difícil la aplicación, es decir ; que si bien la comunicación se realiza a 115200 se está limitado por el bus de comunicación interno de Matlab ®, aun así se logró la implementación aunque no de una manera muy eficiente.

Debido a esta gran problemática, sin importar la arquitectura de la computadora siempre se tendrá esta limitación interna de comunicación, lo cual lleva a la conclusión de que en la implementación de sistemas de alta velocidad como en este caso el uso de Matlab ® es deficiente. Es por eso que todos los controladores e identificación se programaron en código puro como se muestra en el Apéndice B.

Para el segundo caso directamente embebido se realizó la programación embebida a una velocidad de 115200 baudios por segundo en la comunicación serial. Logrando efectivamente la implementación.

En ambos casos se obtienen gráficas para la acción de control y el error, no se obtiene directamente una gráfica para la posición ya que en la gráfica del error se observa claramente su convergencia a cero.

3.6.2. Implementación modos deslizantes en Matlab - Arduino

Implementar sistemas de control en Simulink no es una tarea muy compleja ya que se pueden utilizar todas las librerías de Matlab ® para su implementación. Además este se encarga de realizar todos los cálculos y comunicarse con el microcontrolador por puerto serie, es decir el microcontrolador es un simple DAC o tarjeta de adquisición de datos. La Figura 3.22 muestra el diagrama a bloques utilizado en Matlab ® Simulink para realizar la implementación Matlab ®-Arduino.

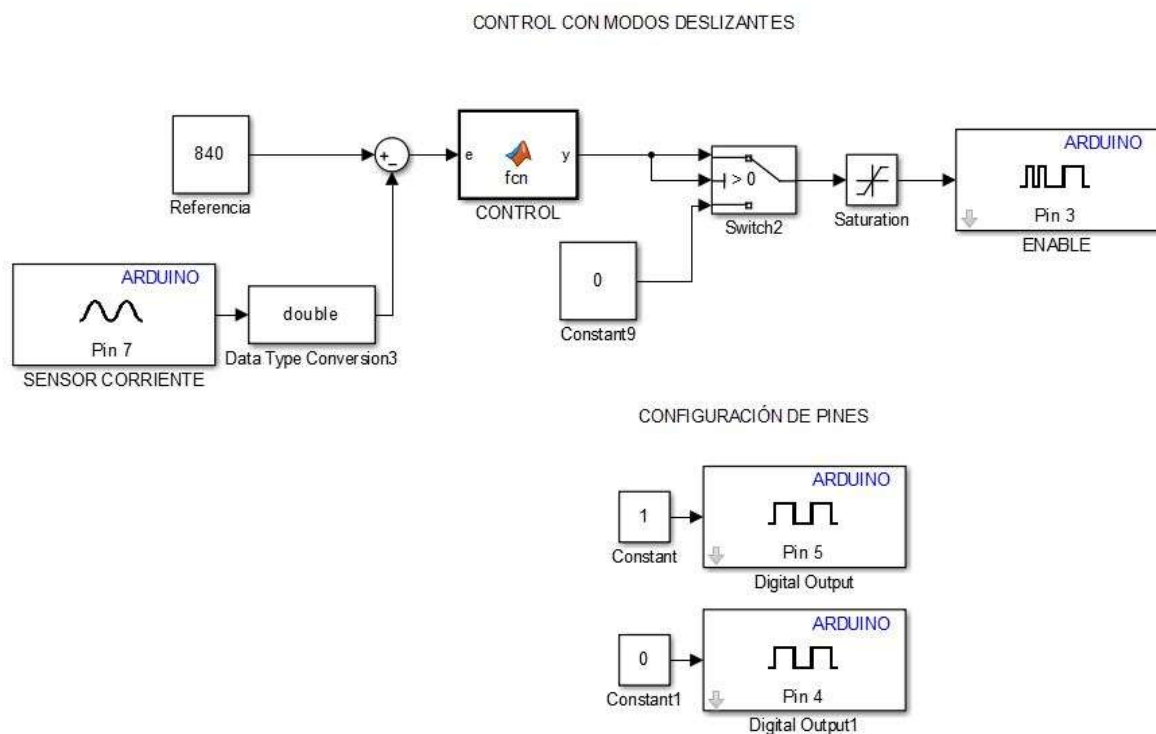


Figura 3.22: Esquema de simulación para el control con modos deslizantes del sistema de levitación magnética de un grado de libertad

Para el error en la Figura 3.23 se observa que oscila a no más de 10 valores, lo cual es un error pequeño pero considerable ya que como se explicó en una sección previa, se tienen 1023 valores, estas oscilaciones están presentes debido al tiempo de comunicación entre Matlab ® y Arduino.

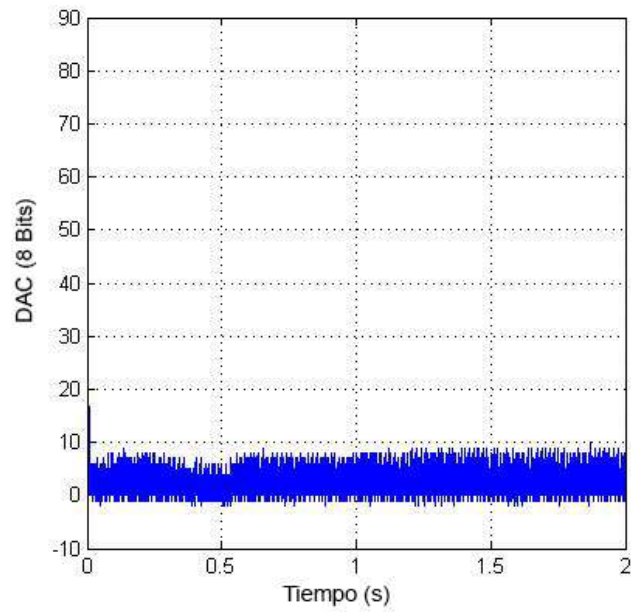


Figura 3.23: Respuestá del error para el control con modos deslizantes Matlab-Arduino

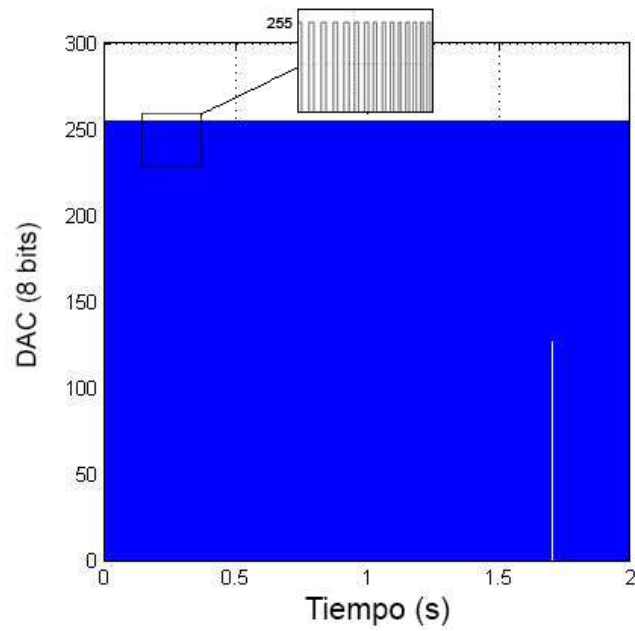


Figura 3.24: Respuestá de la acción de control para el control con modos deslizantes Matlab-Arduino

La Figura 3.24 muestra la acción de control aplicada en el actuador para esta implementación. Resulta bastante notorio la deficiencia de la implementación en Matlab®-Arduino.

3.6.3. Implementación modos deslizantes en Arduino embebido

La Figura 3.25 muestra los resultados de la acción de control aplicada al actuador (puente H), y del error. Se observa que la acción de control (color rojo) es una acción de control on/off de alta frecuencia, así como también que el error (color azul) oscila alrededor de 0. La ganancia para desarrollar esta implementación, ha sido seleccionada en base a la ganancia mayor que puede ser utilizada en la implementación de los sistemas reales (255 debido a que se emplean 8 bits), es decir $k = 255$,

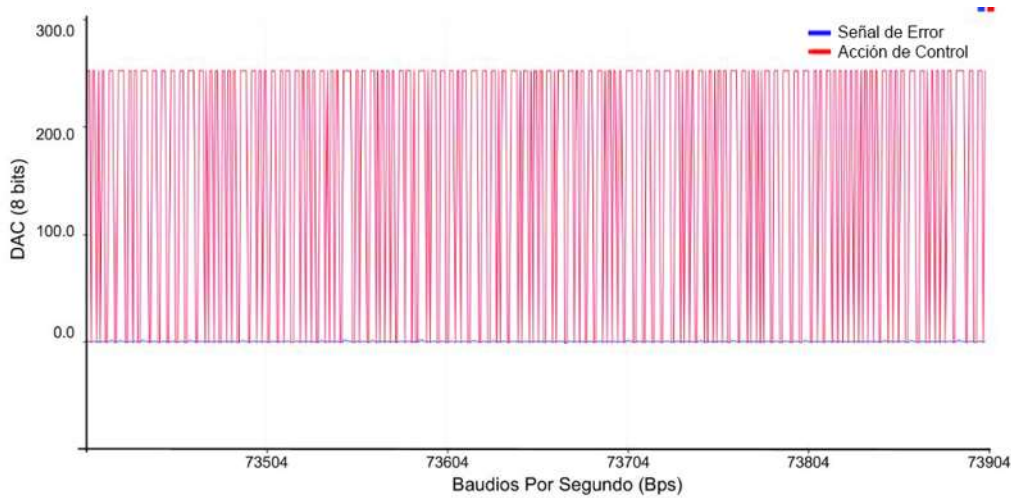


Figura 3.25: Señales de control y error para el control con modos deslizantes embebido.

Se observa una mejora tanto en el chattering como en la reducción del error respecto de lo mostrado en la Figura 3.23 y 3.24, respectivamente.

La señal del error se muestra más a detalle en la Figura 3.26.

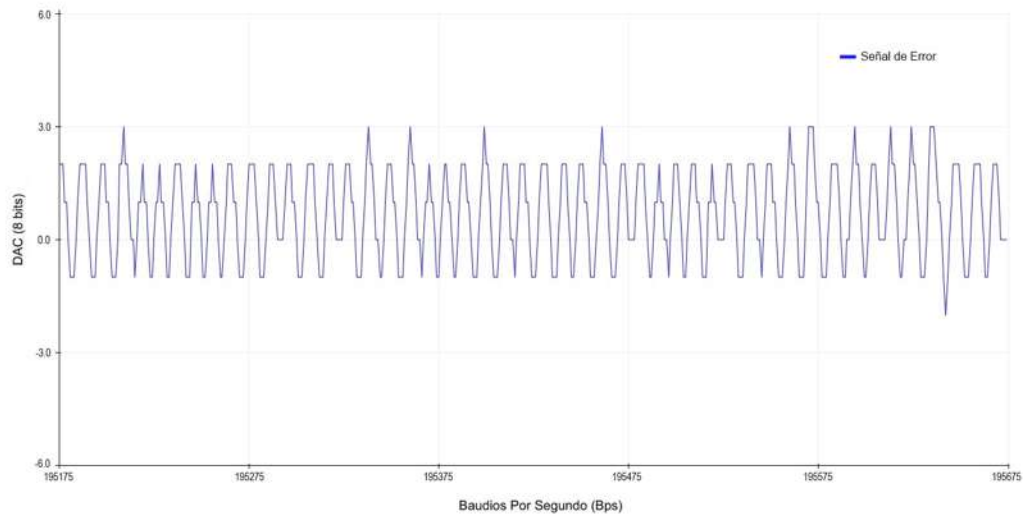


Figura 3.26: Señal de error para el control con modos deslizantes embebido

3.7. Control por modos deslizantes de orden superior

Como se observó en la sección anterior, el principal problema del control por modos deslizantes de primer orden, es el efecto del chattering. Para dar solución a este efecto indeseado, han surgido nuevas técnicas. La idea principal es cambiar la dinámica del control en una pequeña vecindad de la superficie deslizante, con el fin de evitar la discontinuidad real y al mismo tiempo preservar las propiedades principales de todo el sistema.

Para este fin (reducir el chattering) se desarrollaron los modos deslizantes de orden superior. Generalizando el principio de modo deslizante y actuando sobre las derivadas temporales de orden superior, se conservan las ventajas del enfoque original, y al mismo tiempo disminuye el efecto del chattering, proporcionando así una mayor precisión cuando el control es implementado.

El orden de los modos deslizantes caracteriza el grado de suavidad en la dinámica dentro de la vecindad de la superficie deslizante.

3.7.1. Supertwisting

La técnica de supertwisting es descrita a continuación, además se sabe que cuenta con estabilidad en tiempo finito.

$$u = -\lambda|s|^{1/2}\text{sign}(s) + u_1 \quad (3.11)$$

donde:

- $u_1 = \alpha \text{sign}(s)$
- s = Superficie deslizante.
- λ = Ganancia de control

3.7.2. Implementación SuperTwisting en Matlab - Arduino

Para esta implementación se procede exactamente de la misma manera como se realizó en la sección 3.6.2, donde el microcontrolador es un simple DAC o tarjeta de adquisición de datos.

La Figura 3.27 muestra el diagrama a bloques utilizado en Matlab [®] Simulink para realizar la implementación Matlab [®] - Arduino.

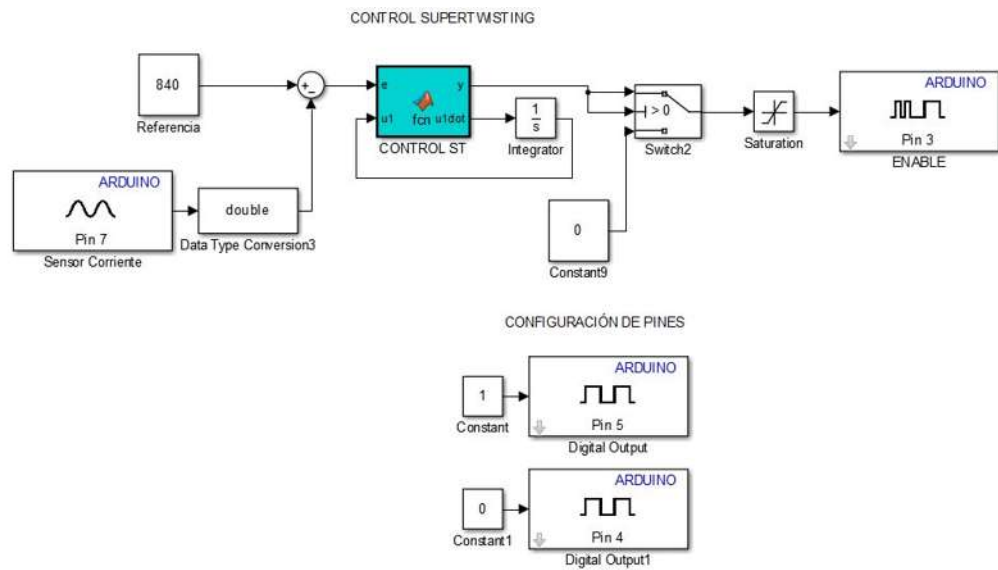


Figura 3.27: Esquema de simulación para el control SuperTwisting del sistema de levitación magnética de un grado de libertad

Para la señal del error, en la Figura 3.28 se observa que este disminuyó de manera considerable respecto al control con modos deslizantes de primer orden (Figura 3.23). La reducción del error es bastante notoria, así como en sus respectivas acciones de control, específicamente en las conmutaciones, se observa que en la Figura (3.29) son mucho menores que en la Figura (3.24).

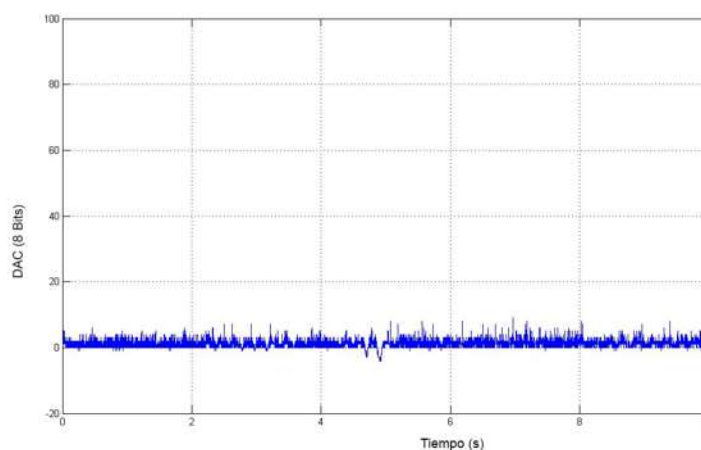


Figura 3.28: Respuesta del error para el control SuperTwisting Matlab-Arduino

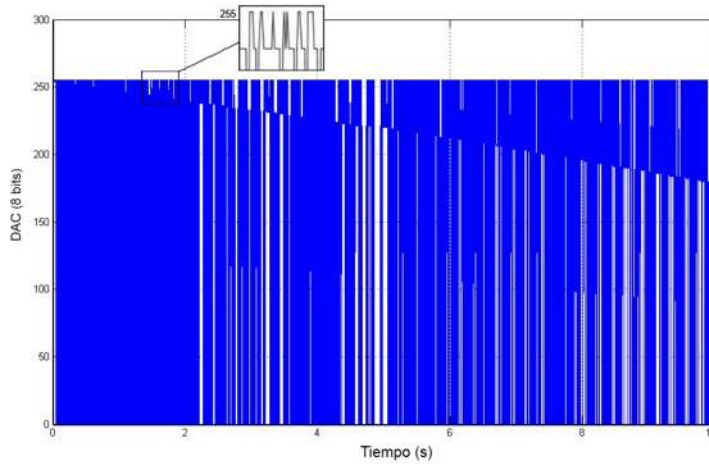


Figura 3.29: Respuestá de la acción de control para el control SuperTwisting Matlab-Arduino

La Figura 3.29 muestra la acción de control aplicada en el actuador para está implementación. Nuevamente se observa la diferencia clara de la implementación usando Matlab ® ya que al realizar mayor cantidad de operaciones la computadora, mayor es el tiempo de procesamiento y envío de información por puerto serial al arduino, lo que hace mermar la acción de control para un sistema rápido como este.

Sin embargo aún hay errores aunque son muy mínimos, se realiza una implementación final añadiendo el filtro de Kalman tanto al control con modos deslizantes de primer orden como al SuperTwisting, cabe resaltar que está solo se pudo realizar en Arduino ya que en Matlab ® el tiempo de comunicación imposibilitó dicha acción.

3.7.3. Implementación SuperTwisting en Arduino embebido

La Figura 3.30 muestra los resultados de la acción de control aplicada al actuador (puente H), y del error. Se observa que la acción de control (color rojo) es una acción de control on/off de alta frecuencia, así como también que el error (color azul) oscila alrededor de 0. La ganancia para desarrollar está implementación, ha sido seleccionada en base a la ganancia mayor que puede ser utilizada en la implementación de los sistemas reales (255 debido a que se emplean 8 bits), es decir $\alpha = 255$ y $\lambda = 255$,

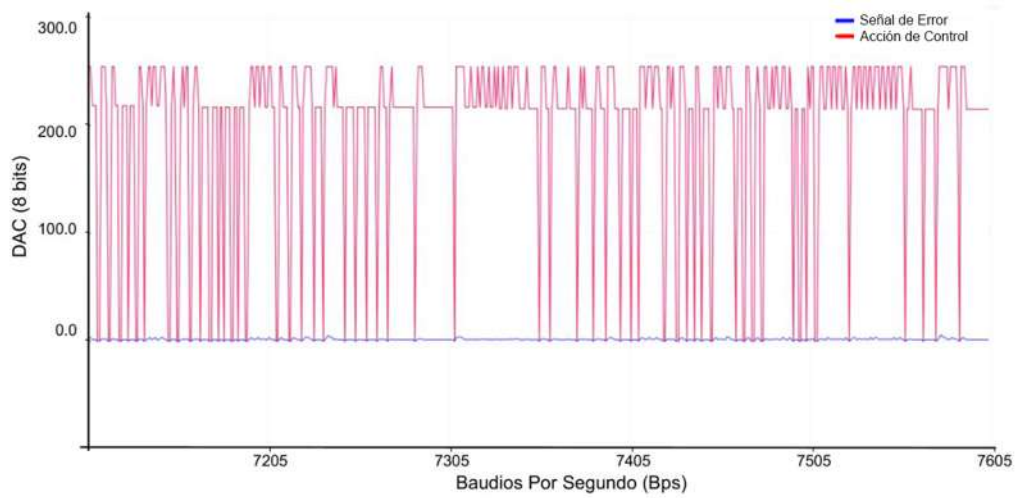


Figura 3.30: Señales de control y error para el control SuperTwisting embebido.

Se observa claramente como la acción de control mejora respecto a la acción de control implementada en Matlab [®] mostrada la Figura 3.29, lo que reduce considerablemente el chattering, esto se debe a la acción integral presente en el controlador. Esta acción también hace aún más robusto este esquema de control no solo ante perturbaciones, debido a su alta frecuencia de conmutación, sino ahora incluso tendiendo a un error cero cada vez más con el paso del tiempo, en comparativa de la señal de error mostrada en la Figura 3.28.

La señal del error se muestra más a detalle en la Figura 3.31.

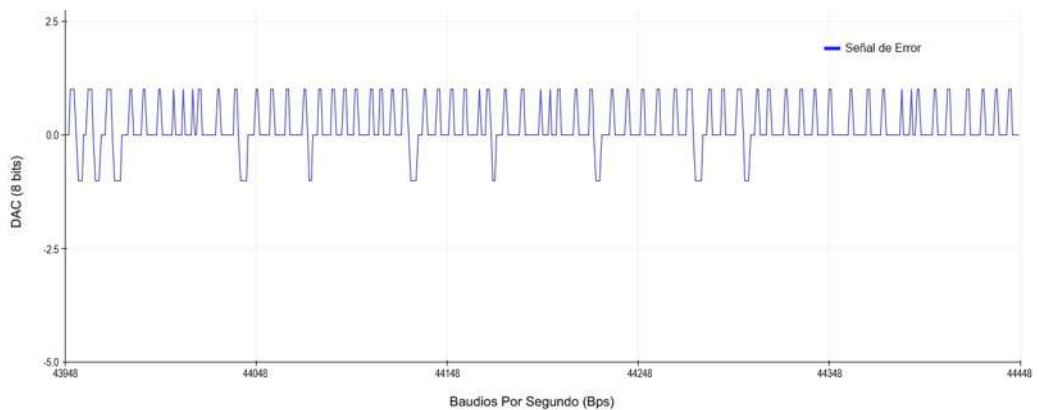


Figura 3.31: Señal de error para el control SuperTwisting embebido.

3.8. Implementación añadiendo filtro de Kalman

Para esta implementación donde se agrega el KF con el fin de reducir el ruido en las muestras, se observa que ahora Matlab ® ya es incapaz de poder controlar al sistema. Como se ha mencionado a lo largo de esta sección esto es debido al protocolo interno de comunicación entre el ambiente mismo de Matlab ® y el microcontrolador, pasando a realizar todo de manera embebida. Los códigos de implementación son mostrados en el Apéndice A.

3.8.1. Implementación control con modos deslizantes y filtro de Kalman en Arduino embebido

La Figura 3.32 muestra los resultados de la acción de control aplicada al actuador (puente h), y del error. Se observa que la acción de control (color rojo) es una acción de control on/off de alta frecuencia, así como también que el error (color azul) oscila alrededor de 0. La ganancia para desarrollar esta implementación es $k = 255$,

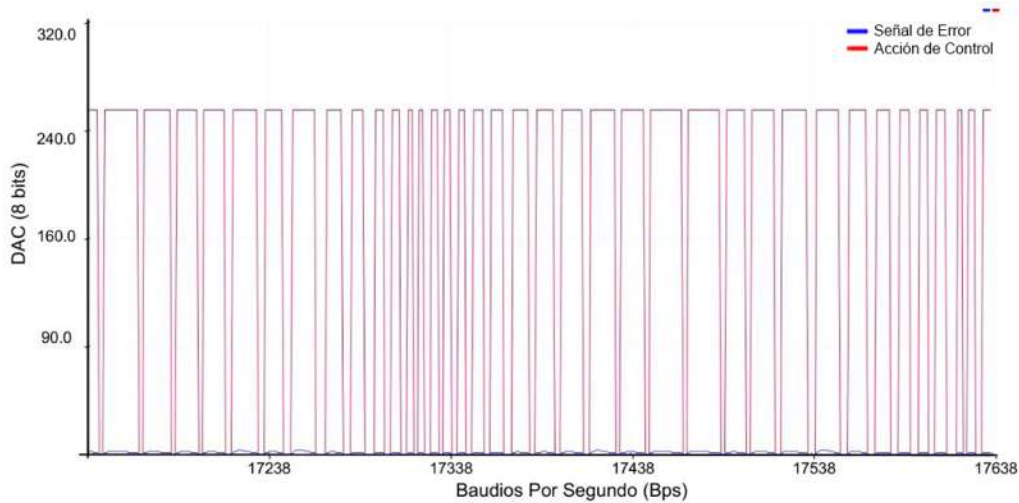


Figura 3.32: Control con modos deslizantes y filtro de Kalman - Arduino

Se obtienen mejoras en la acción de control, es decir una reducción en el chattering debido a la disminución del ruido en el sensor, comparando las Figuras 3.25 y 3.32 respectivamente, así mismo se observa mejoras en la reducción del error. Se observa que la

capacidad del microcontrolador se ve afectada en la implementación, pasando de utilizar una velocidad de baudios por segundo de 115200 a 230400. Como se observa comparando las Figuras (3.25) y (3.32) respectivamente.

La señal del error se muestra a detalle en la Figura 3.33.

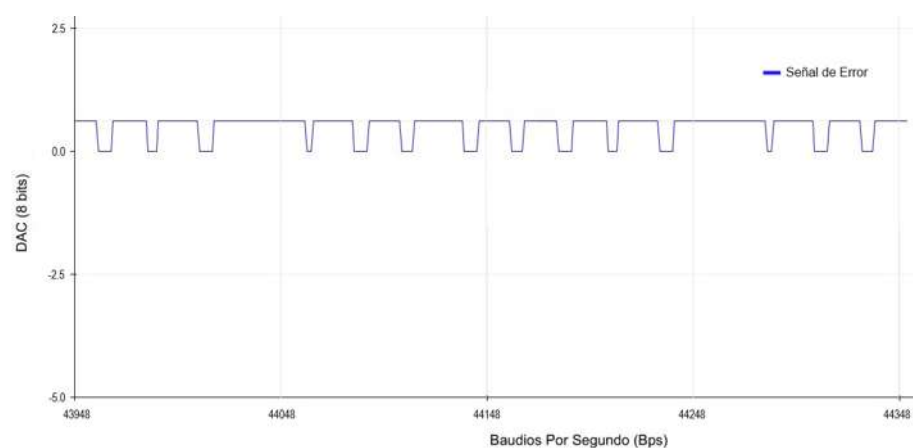


Figura 3.33: Señal de error para el control con modos deslizantes y filtro de Kalman - Arduino

3.8.2. Implementación control SuperTwisting y filtro de Kalman en Arduino embebido

La Figura 3.34 muestra los resultados de la acción de control aplicada al actuador (puente H), y del error. Se observa que la acción de control (color rojo) es una acción de control on/off de alta frecuencia, así como también que el error (color azul) oscila alrededor de 0. Las ganancias para desarrollar esta implementación $\alpha = 255$ y $\lambda = 255$,

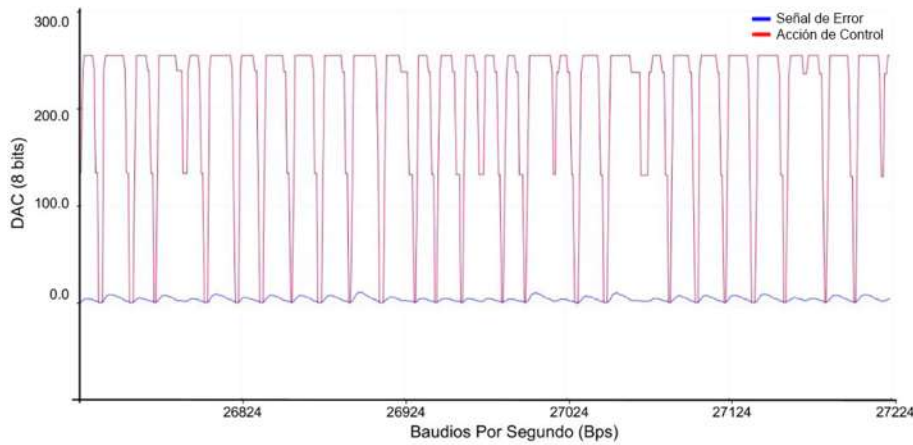


Figura 3.34: Control SuperTwisting y filtro de Kalman - Arduino

Se observa claramente como la acción de control mejora considerablemente reduciendo el chattering pero se observa que baja el desempeño del microcontrolador aun usando 230400 bps, lo cual afecta al error el cual aumenta un poco esto debido al esfuerzo computacional que debe realizar el microcontrolador al usar el KF y el control supertwisting simultáneamente. Sin embargo si no se usara una comunicación serie, es decir; solamente el microcontrolador realizando la implementación sin enviar datos a la computadora el desempeño de esta combinación es altamente efectiva ya que se anulan los tiempos de comunicación.

La señal del error se muestra a detalle en la Figura 3.35.

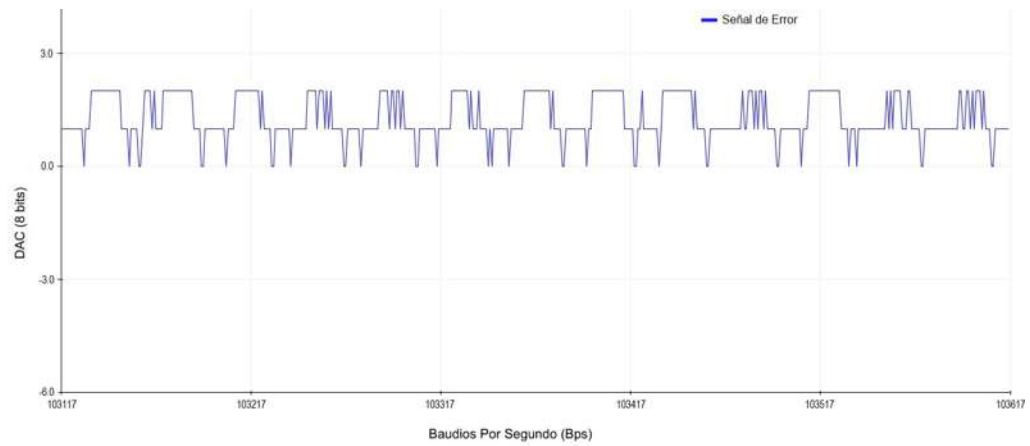


Figura 3.35: Señal del error para el control con modos deslizantes y filtro de Kalman - Arduino

3.9. Estimación del parámetro de la fuerza electromagnética

Como se observa en la Sección 3.3 para poder obtener el parámetro k se debe buscar el punto de equilibrio del sistema de acuerdo a la Ecuación (3.3), y como indica la expresión que el parámetro k varía de acuerdo a el punto de levitación (punto de equilibrio). Lo que significa que el parámetro no es constante, incluso si es considerado constante se obtiene una buena aproximación como se mostró anteriormente. No obstante este modelo donde se considere el parámetro k constante no es un modelo general para la situación donde la levitación de la bola ferromagnética es deseada a varias distancias, como en el caso del seguimiento de una señal sinusoidal. Tomando en cuenta lo descrito anteriormente, una solución podría ser desarrollar un modelo matemático para el parámetro electromagnético k , lo cual es sumamente complicado ya que fenómenos como la histéresis o magnetización son difíciles de modelar. Para determinar la posición de equilibrio y la corriente de equilibrio así como el valor del parámetro k se usa una aproximación experimental, donde la corriente del circuito y posición de la bola son medidas. La bola ferromagnética es localizada en un estante no magnético directamente debajo del electroimán. La posición exacta es registrada por el sensor de posición, luego una rampa suave de voltaje es aplicada al circuito magnético y la posición resultante y la corriente consumida por el electroimán son medidas. La Tabla

3.8 presenta una serie de datos experimentales obtenidos para la posición de equilibrio y la corriente eléctrica de equilibrio.

Tabla 3.8: Dependencia entre posición de equilibrio y la corriente de equilibrio

Posición de equilibrio (m).	Corriente de equilibrio (Amp)	Masa (Kg)	Gravedad (m/s ²)	Parámetro K $kg * m^5 * s^{-2} * A^{-1}$
0.018	0.3651	0.0587	9.81	1.65571E-07
0.02	0.360261624	0.0587	9.81	2.55746E-07
0.022	0.355456276	0.0587	9.81	3.795E-07
0.023	0.35068392	0.0587	9.81	4.59518E-07
0.024	0.34594452	0.0587	9.81	5.52263E-07
0.025	0.341238039	0.0587	9.81	6.59189E-07

Los valores experimentales presentados en la Tabla 3.8 indican que el valor del parámetro k en el modelo matemático del levitador magnético no puede ser considerado constante ya que para cada punto caracterizado por el par posición-corriente. El parámetro electromagnético k tiene una dependencia no lineal para lo cual, encontrar una expresión analítica para este parámetro es complejo, la dificultad se centra en modelar una serie de fenómenos no lineales tales como: la magnetización de la bola ferromagnética, histéresis, campos magnéticos etc.

3.9.1. Estimación con redes neuronales

Una red neuronal es un procesador paralelo distribuido compuesto de unidades simples de procesamiento, los cuales son propensos por naturaleza a guardar conocimiento experimental y hacerlo disponible para su uso.

Con la intención de capturar el comportamiento real del parámetro electromagnético k en el modelo del sistema de levitación magnética, se introduce una red neuronal capaz de proporcionar una aproximación a este parámetro el cual será entrenado usando datos experimentales obtenidos. La red neuronal empleada para la aproximación del parámetro electromagnético k es una red del tipo feed-forward con 2 neuronas de entrada (posición y corriente), una capa oculta y una neurona de salida (parámetro k) como lo muestra la Figura 3.36.

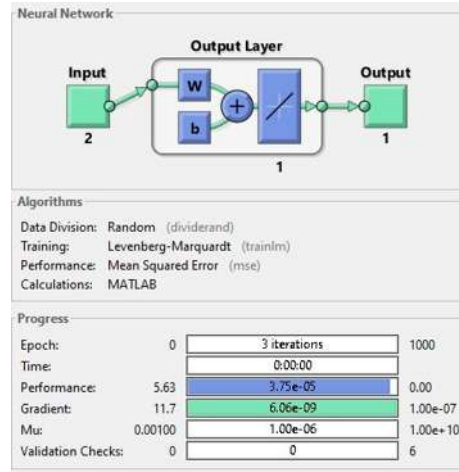


Figura 3.36: Red neuronal para aproximar el parámetro electromagnetico.

Para la función de activación de las neuronas en la capa oculta y para la neurona de salida de la red neuronal es usada una función lineal (Figura 3.37), la cual es comunmente utilizada para funciones de ajuste de acuerdo a la Ecuación 3.12.

$$y = x \quad (3.12)$$

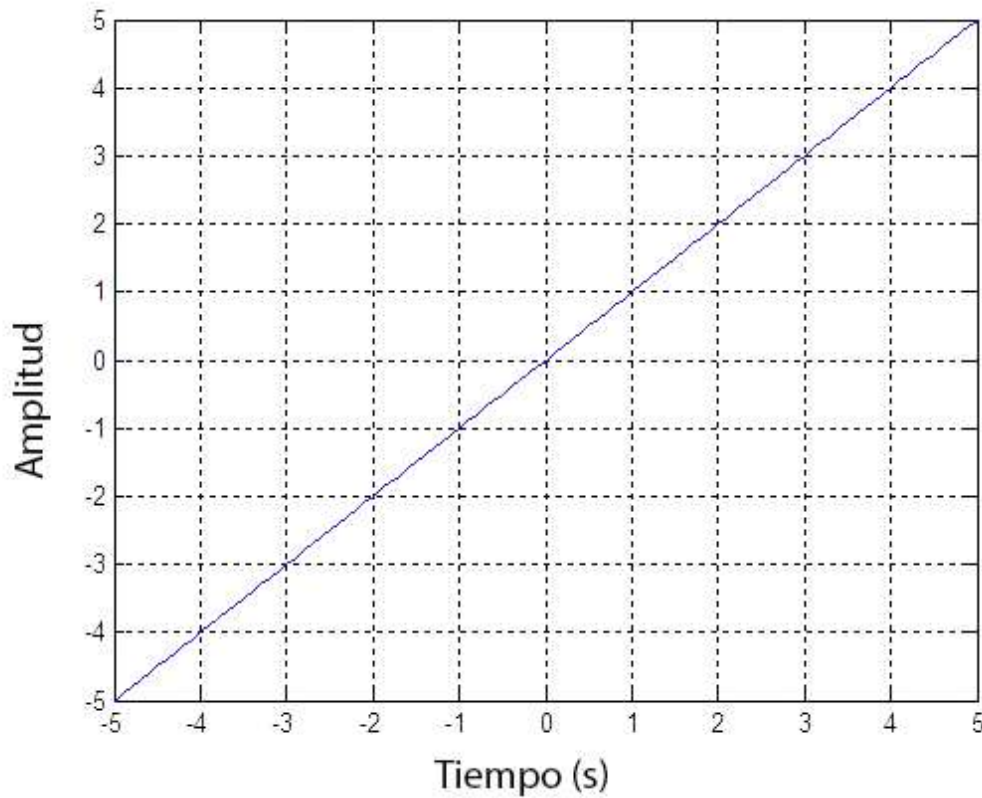


Figura 3.37: Función de activación lineal

También podemos hacer uso de la función de activación Sigmoide (Figura 3.38), la cual es la más común como función de activación usada en la construcción de redes neuronales artificiales, la cual define una función estrictamente creciente que exhibe un gran balance entre un comportamiento lineal y no lineal.

$$P(t) = \frac{1}{1 + e^{-t}} \quad (3.13)$$

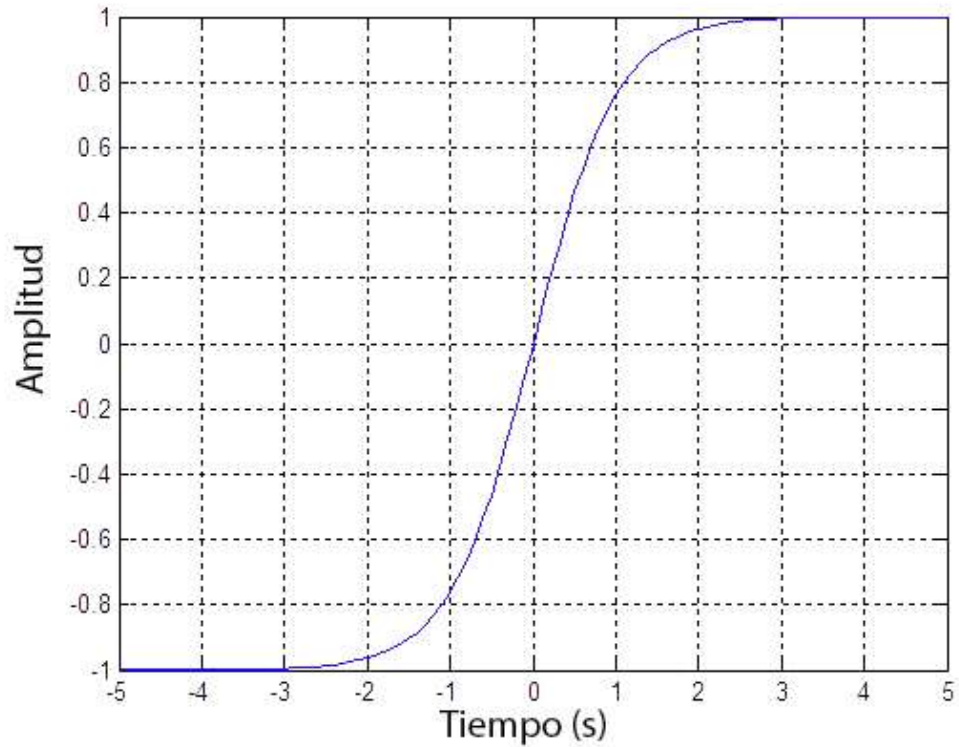


Figura 3.38: Función de activación sigmoide

El entrenamiento de la red se hace de acuerdo a los datos experimentales provenientes de la Tabla 3.8, los cuales a su vez vienen de los datos mostrados en el Apéndice B. De este modo se obtiene la validación de la red neuronal de acuerdo a la Tabla 3.9

Tabla 3.9: Validación de Red Neuronal

Posición de equilibrio (m).	Corriente de equilibrio (Amp)	Parámetro K (Objetivo) 10^{-7}	Parámetro K (Red Neuronal) 10^{-7}
0.018	0.3651	1.65571	1.6557
0.02	0.360261624	2.55746	2.6717
0.022	0.355456276	3.795	3.6819
0.023	0.35068392	4.59518	4.6001
0.024	0.34594452	5.52263	5.5126
0.025	0.341238039	6.59189	6.4194

Matlab ® permite por medio del comando "*gensim*" crear la red neuronal en el ambiente de simulink, como se muestra en la Figura 3.39, la cual valida los datos de la Tabla 3.9

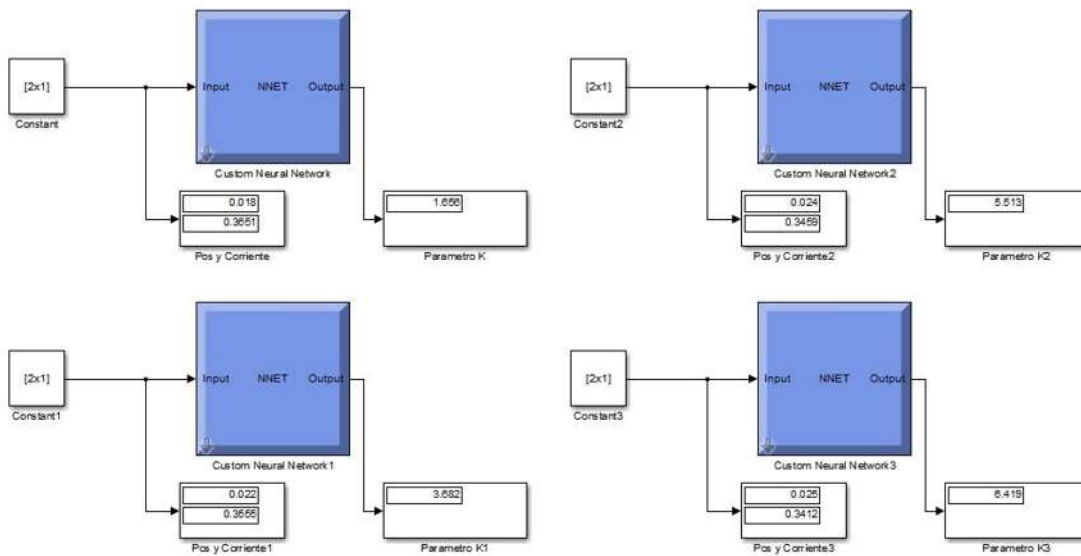


Figura 3.39: Red neuronal implementada en Matlab Simulink

La Figura 3.40, muestra la validación y como decrece el error cuadrático medio de la red neuronal.

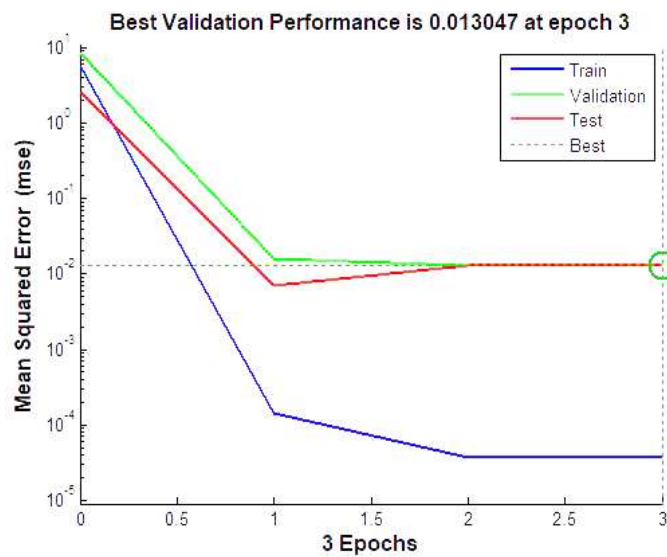


Figura 3.40: Error cuadrático medio de la red neuronal

A lo largo de este capítulo hemos venido analizando cómo es que en aplicaciones

donde el tiempo es un factor importante, simulink no es la forma más óptima y por mucho menos para la implementación. Por eso codificar la red neuronal embebida en el microcontrolador es la forma más óptima. A continuación presentamos como recrear la red neuronal a partir de la red creada en Matlab ® esto obteniendo los bias y pesos para su reconstrucción en el microcontrolador eliminando los tiempos de reloj implícitos, así como los tiempos muertos debido al protocolo de comunicación entre Matlab ® y el elemento controlador en este caso un microcontrolador.

3.9.2. Red neuronal embebida

De acuerdo a la Sección 3.9.1 se procede de igual manera, la red neuronal empleada para la aproximación del parámetro electromagnético k es una red del tipo Backpropagation, usando funciones de activación sigmoideas y basada en el algoritmo del gradiente descendiente.

Se establece un sistema de matrices que almacenan los pesos y los datos que se alimentan a través de la red. Los datos se transmiten secuencialmente y luego los errores se propagan a través de la red ajustando las ponderaciones.

El código trabaja de la siguiente manera (Apéndice A):

- Configurar las matrices y asignar pesos aleatorios.
- Iniciar un ciclo que recorra cada elemento de datos de entrenamiento.
- Aleatorizar el orden en que se ejecutan los datos de entrenamiento a través de cada iteración para garantizar que no se produzca la convergencia en los mínimos locales.
- Alimentar los datos a través de la red calculando la activación de los nodos de la capa oculta, los nodos de la capa de salida y los errores.
- Volver propagar los errores a la capa oculta.
- Actualizar los pesos asociados.
- Comparar el error con el umbral y decidir si ejecutar otro ciclo o si el entrenamiento se ha completado.

- Enviar una muestra de los datos de entrenamiento al monitor en serie cada mil ciclos.

```

Training Pattern: 0
Input 0.0179999995 0.0199999990 0.0220000004 0.0229999995 0.0240000009 0.0250000000
Training Pattern: 1
Input 0.365000022 0.3602616310 0.3554562568 0.3506339275 0.3459445238 0.3412380456
Target 1.6556999683 2.6717000007 3.6819000244 4.6001000404 5.5125959450 6.419402151
Output 0.51011 0.44150 0.45482 0.45430 0.44810 0.43714
TrainingCycle: 1 Error = 48.51107

Training Pattern: 0
Input 0.0179999995 0.0199999990 0.0220000004 0.0229999995 0.0240000009 0.0250000000
Training Pattern: 1
Input 0.365000022 0.3602616310 0.3554562568 0.3506339275 0.3459445238 0.3412380456
Target 1.6556999683 2.6717000007 3.6819000244 4.6001000404 5.5125959450 6.419402151
Output 0.51130 0.47076 0.50716 0.51704 0.53973 0.54661
TrainingCycle: 1000 Error = 38.05875

Training Pattern: 0
Input 0.0179999995 0.0199999990 0.0220000004 0.0229999995 0.0240000009 0.0250000000
Training Pattern: 1
Input 0.365000022 0.3602616310 0.3554562568 0.3506339275 0.3459445238 0.3412380456
Target 1.6556999683 2.6717000007 3.6819000244 4.6001000404 5.5125959450 6.419402151
Output 0.95914 0.95961 0.95954 0.95956 1.00000 1.00000
TrainingCycle: 2000 Error = 37.55812

Training Pattern: 0
Input 0.0179999995 0.0199999990 0.0220000004 0.0229999995 0.0240000009 0.0250000000
Training Pattern: 1
Input 0.365000022 0.3602616310 0.3554562568 0.3506339275 0.3459445238 0.3412380456
Target 1.6556999683 2.6717000007 3.6819000244 4.6001000404 5.5125959450 6.419402151
Output 0.95961 0.95954 0.95957 0.95976 0.95959 1.00000
TrainingCycle: 3000 Error = 37.06091

...

Training Pattern: 0
Input 0.0179999995 0.0199999990 0.0220000004 0.0229999995 0.0240000009 0.0250000000
Training Pattern: 1
Input 0.365000022 0.3602616310 0.3554562568 0.3506339275 0.3459445238 0.3412380456
Target 1.6556999683 2.6717000007 3.6819000244 4.6001000404 5.5125959450 6.419402151
Output 1.67159 2.56891 3.67135 4.43020 5.52344 6.81213
TrainingCycle: 320000
Training Set Solved!
-----

```

Figura 3.41: Red neuronal embebida

En la Figura 3.41 se muestra la evolución de la red neuronal, siendo el desempeño deficiente en tiempo de convergencia debido a la capacidad computacional del microcontrolador, para eliminar este problema lo abordaremos en la siguiente sección.

3.9.3. Red neuronal reconstruida

Para la reconstrucción de la red la cual es de la forma mostrada en la Figura 3.36 necesitamos conocer los pesos y bias los cuales se obtienen con el comando *formwb* y *separatewb* para la red se tiene:

$$w1 = 0.1268 \quad (3.14)$$

$$w2 = -0.8743 \quad (3.15)$$

$$bias = 0.0011 \quad (3.16)$$

La función de activación es lineal, esto se obtiene con el comando *layers* para lo cual la salida de la red está dada por:

$$z = w1 * Entrada1 + w2 * Entrada2 + bias \quad (3.17)$$

Si bien la salida de la red está determinada por la Ecuación (3.17), s debe aplicar conversiones de entrada y salida acotadas (de -1 a 1), esto se obtiene con el comando *processedRange* Como las entradas deben ser pasadas por una función que ajuste estás al rango necesario, esto se hace por medio de la siguiente Ecuación.

$$Scala = \frac{max - min}{xmax - xmin}(x - xmin) + min$$

Donde :

- max = Es el valor de conversión mayor de Matlab ®.
- min = Es el valor de conversión menor de Matlab ®
- xmax = Es el valor de entrada mayor
- xmin = Es el valor de entrada menor
- x = Entrada

Para lo cual la reconstrucción de la red se muestra en el diagrama de la Figura 3.42

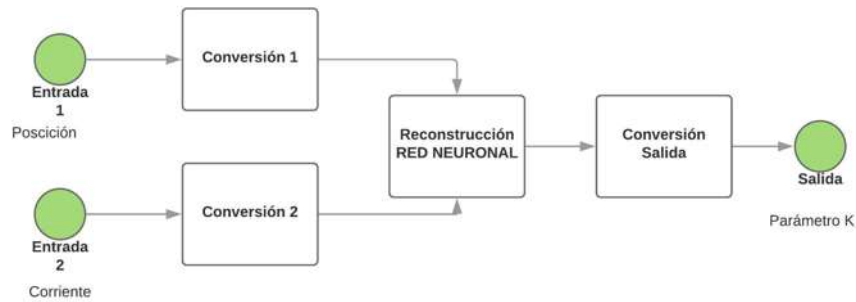


Figura 3.42: Reconstrucción de la red neuronal

Obteniendo los siguientes resultados:

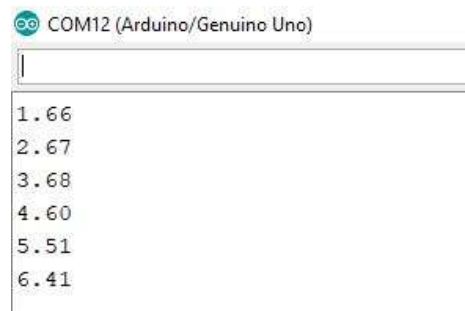


Figura 3.43: Salida - reconstrucción red neuronal

Se observa como se pudo reconstruir totalmente la red y eliminando totalmente el tiempo muerto debido al protocolo de comunicación entre Matlab ® y el elemento controlador, en este caso un microcontrolador. Así mismo, el tiempo de cálculos se redujo en el algoritmo a unas simples cantidad de líneas y operaciones. (El código de implementación se muestra en el Apéndice B).

3.10. Conclusiones

En este capítulo se presentó la implementación del sistema de levitación magnética de un grado de libertad, desde la creación del electroimán, la caracterización de los sensores, la eliminación del ruido haciendo uso del filtro de Kalman, estimación paramétrica del

sistema y el control por medio de 2 técnicas de control no convencional (modos deslizantes de primer orden y modos deslizantes de orden superior “superTwisting”). Se comparó el desempeño de cada técnica, de igual manera se comparó la implementación embebida en el microcontrolador contra la implementación en la computadora en un software como Matlab® y su comunicación con el microcontrolador por puerto serie y sus desventajas del porqué otras técnicas fallarían en esta implementación, debido a la carga computacional considerable a implementar en el microcontrolador. Finalmente se realizó la estimación del parámetro de la fuerza electromagnética usando redes neuronales a nivel simulación e implementación embebida en el microcontrolador.

Capítulo 4

Simulación, estimación paramétrica y control del modelo del sistema de levitación magnética de 2 GDL

4.1. Introducción

Este capítulo muestra los resultados de las simulaciones del sistema de levitación magnética de dos grados de libertad. Para la estimación paramétrica se utilizan 3 técnicas distintas: Mínimos cuadrados, mínimos cuadrados recursivos y filtro de Kalman-bucy; siendo la primera técnica fuera de línea y las otras dos en línea respectivamente. El control es aplicado por 2 técnicas de control (Modos deslizantes de primer orden y Modos deslizantes de orden superior "SuperTwisting").

4.2. Estimación paramétrica

Para realizar la estimación paramétrica se realiza un proceso similar al analizado en las secciones 3.4, 3.4.1 y 3.4.2 por medio del método de mínimos cuadrados. Se procede de la misma manera en esta sección; así mismo añadiendo dos técnicas más: Mínimos cuadrados recursivos y Kalman Bucy.

4.2.1. Mínimos cuadrados

Partiendo de la Ecuación diferencial (2.19), la cual es exactamente la misma que (2.6) para el sistema de un grado de libertad mostrado anteriormente.

Planteando en la forma de mínimos cuadrados:

$$\dot{i} = -b_1(i) + b_2(u) \quad (4.0)$$

Donde:

- i = corriente del electroimán
- u = voltaje del electroimán
- $b_1 = \frac{R}{L}$
- $b_2 = \frac{1}{L}$

El modelo anterior puede ser escrito como:

$$y(i) = \varphi(i)^T \theta$$

donde

$$\varphi(i)^T = [i \quad u] \quad (4.1)$$

y

$$\theta^T = [b_1 \quad b_2] \quad (4.2)$$

dónde: b_1, b_2 , son los parámetros a estimar. Así, los parámetros estimados están dados por:

$$\hat{\theta} = [\Phi^T \Phi]^{-1} \Phi^T Y \quad (4.2)$$

Para este caso de estudio:

$$\hat{\theta} = [\varphi_1^T \varphi_1]^{-1} \varphi_1^T (\dot{i}) \quad (4.2)$$

4.2.2. Obtención de los vectores

Para obtener los vectores como se observa en la Ecuación (4.2.1), se inyectó una señal senoidal así como un ruido gaussiano que contamine al sistema, como se muestra en la Figura 4.1.

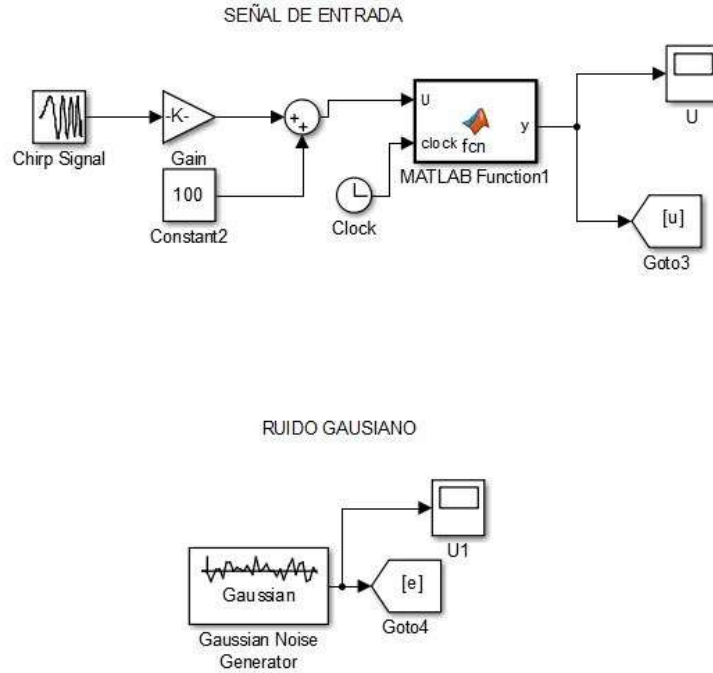


Figura 4.1: Entradas al sistema de levitación magnética de dos grados de libertad.

La diferencia entre el sistema de levitación magnética de dos grados radica en agregar 3 ecuaciones diferenciales más para el grado extra como se muestra en la Figura 4.2, donde se inyectó la señal para la obtención de los vectores propiamente dichos anteriormente.

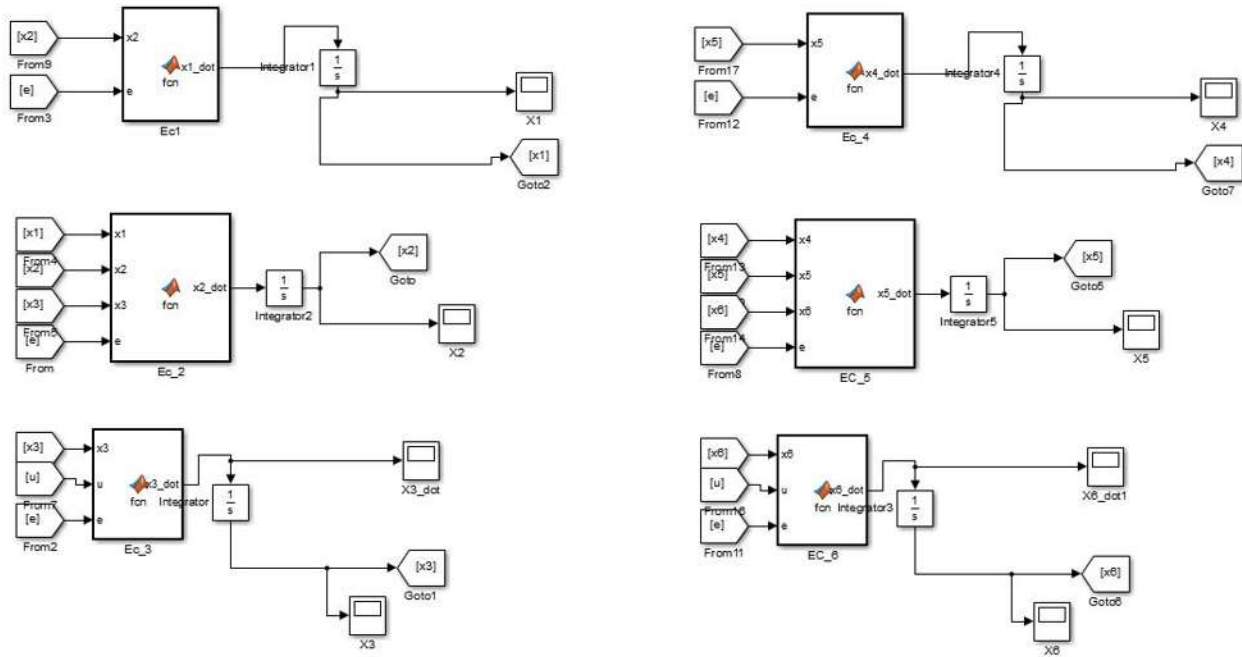
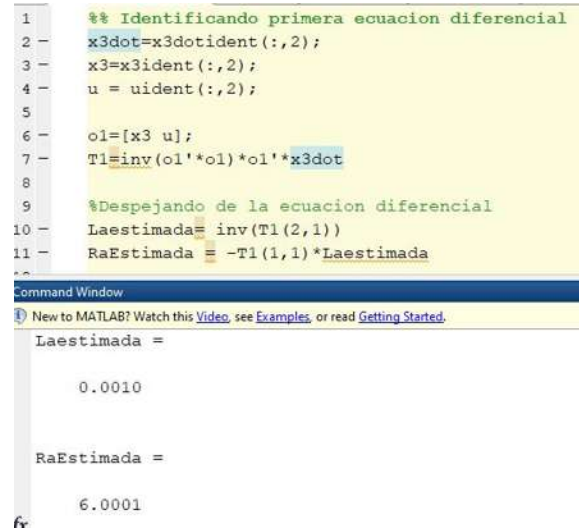


Figura 4.2: Sistema de levitación magnética de dos grados de libertad.

Se exportan los valores de las señales por medio de los scope con el uso del script mostrado en la Figura 4.3, también se observa los resultados de la identificación de los parámetros de resistencia e inductancia del sistema de levitación magnética de dos grados de libertad.



```

1 %% Identificando primera ecuacion diferencial
2 x3dot=x3dotident(:,2);
3 x3=x3ident(:,2);
4 u = uident(:,2);
5
6 o1=[x3 u];
7 T1=inv(o1'*o1)*o1'*x3dot
8
9 %Despejando de la ecuacion diferencial
10 Laestimada= inv(T1(2,1))
11 RaEstimada = -T1(1,1)*Laestimada

```

Command Window

New to MATLAB? Watch this [Video](#), see [Examples](#), or read [Getting Started](#).

Laestimada =

0.0010

RaEstimada =

6.0001

Figura 4.3: Resultados de identificación

4.2.3. Mínimos cuadrados recursivos

Considerando nuevamente el modelo desconocido (2.19), se utiliza el algoritmo de mínimos cuadrados recursivos para la estimación paramétrica.

Considerando que la matriz $\phi(t)$ es de rango pleno, esto es, $\phi^T(t)\phi(t)$ es no singular $\forall t \geq t_0$

Dado $\hat{\theta}^0 y P(t_0) = [\phi^T(t)\phi(t)]^{-1}$ los $\hat{\theta}$ estimados por mínimos cuadrados están dados por:

$$\hat{\theta}(t) = \hat{\theta}(t-1) + K(t)[y(t) - \varphi^T(t)\hat{\theta}(t-1)] \quad (4.2)$$

$$K(t) = P(t-1)\varphi(t)[I + \varphi^T(t)P(t-1)\varphi(t)]^{-1} \quad (4.2)$$

$$P(t) = [I - K(t)\varphi^T(t)]P(t-1) \quad (4.2)$$

usando un modelo de segundo orden para aproximar la variable de salida $y(i)$ como:

$$y(i) = -b_1x_3 + b_2u + e \quad (4.2)$$

donde:

- x_3 = corriente del electroimán
- u = voltaje del electroimán
- $b_1 = \frac{R}{L}$
- $b_2 = \frac{1}{L}$
- e = Ruido Gaussiano

Se procede de la misma forma, teniendo especial cuidado en las condiciones iniciales de $P(t)$ las cuales tienen que ser de acuerdo al orden del sistema, es decir; las condiciones iniciales deben de ser matricial de " $n \times n$ ".

En la Figura 4.4 se muestra el diagrama de simulación, así como en la Tabla 4.1 los resultados de la estimación de los parámetros, mencionando que se hace uso de la mismas señales que en la sección de minimos cuadrados.

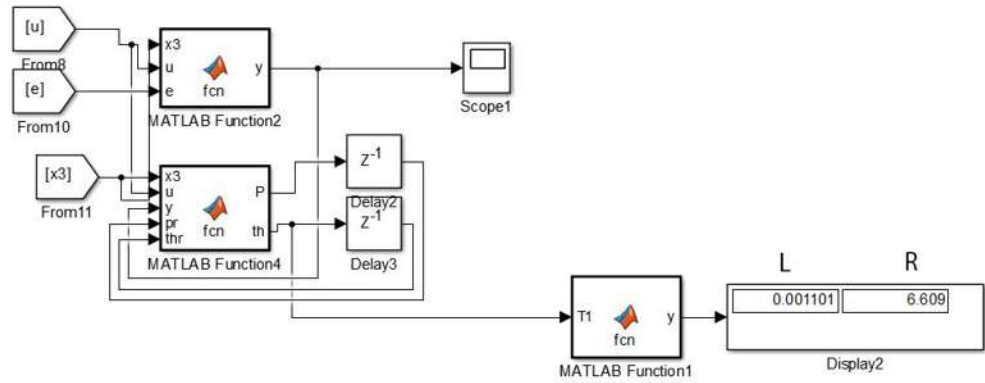


Figura 4.4: Diagrama de identificación usando el método de mínimos cuadrados recursivos

Tabla 4.1: Resultados de identificación usando el método de mínimos cuadrados recursivos

Tiempo	Inductancia	Resistencia
0	0	0
0.1	5.76E-07	0.00345324
0.2	0.00143007	6.85804431
0.3	0.00130074	6.44314301
0.4	0.00110143	6.60858044
0.5	0.00110143	6.60858044

Como muestra la Tabla 4.1, la mejor estimación de ambos parámetros se logra en 0.4 segundos.

4.2.4. Filtro Kalman-Bucy

El filtro de Kalman Bucy (KBF) es un algoritmo en línea recursivo, utilizado como estimador de estado, donde los parámetros se convierten en los estados a estimar. El objetivo principal del KBF es determinar el óptimo para el vector de parámetros θ de modo que el error de identificación tienda a cero.

El algoritmo es descrito a continuación.

$$\dot{\theta} = -g\Phi w \varepsilon \quad (4.2)$$

$$\dot{\Phi} = \Psi - g\Phi w w^T \Phi, \quad \Psi, g > 0 \quad (4.2)$$

Donde:

- θ es el valor optimo θ^*
- w es el vector regresor, (contiene los vectores base).
- Ψ, g son matrices nxn de valores constantes que ayudan a determinar la convergencia del error en el algoritmo, para está aplicación el valor es de 1.
- Φ Es la matriz de nxn de covarianza entre más grande en valores numéricos más rápida será la convergencia del estimador.
- ε Es la variable de error es el regresor, el cual contiene los vectores base. Cabe señal entonces que la salida y de nuestro sistema esta dada por $w^T \theta$

Considerando nuevamente la Ecuación diferencial (2.19)

$$w = \begin{pmatrix} x_3 \\ u \end{pmatrix}$$

$$\Phi = \begin{pmatrix} 1000 & 1000 \\ 1000 & 1000 \end{pmatrix}$$

Asi mismo valores más grades apartir de este.

$$\Psi = g = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$$

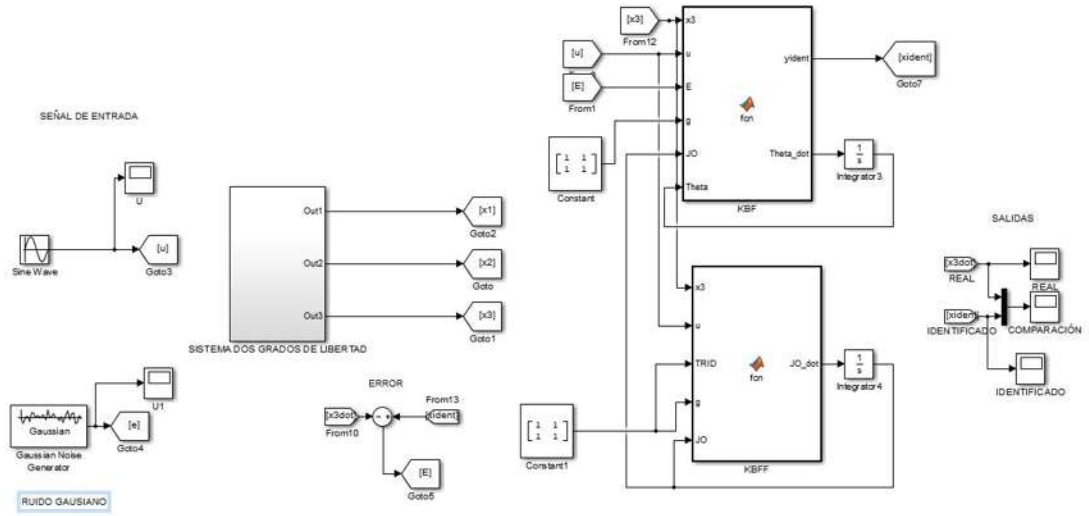


Figura 4.5: Filtro Kalman-Bucy

En la Figura 4.5, se presenta el diagrama de simulación del KBF en el cual se inyecta una señal senoidal para demostrar que el sistema es capaz de adaptarse al sistema real, incluso en presencia de contaminación de ruido gaussiano.

4.2.5. Resultados de simulación KBF

Se realizan dos simulaciones, la primera a poco tiempo Figura 4.6 en la cual se observa exactamente el momento en el cual se logra la total adaptación del estimador con el sistema real, todo esto bajo las consideraciones explicadas anteriormente. Una segunda simulación a mayor tiempo Figura 4.7 en la cual se observa la total convergencia y adaptación del KBF para tiempos posteriores.

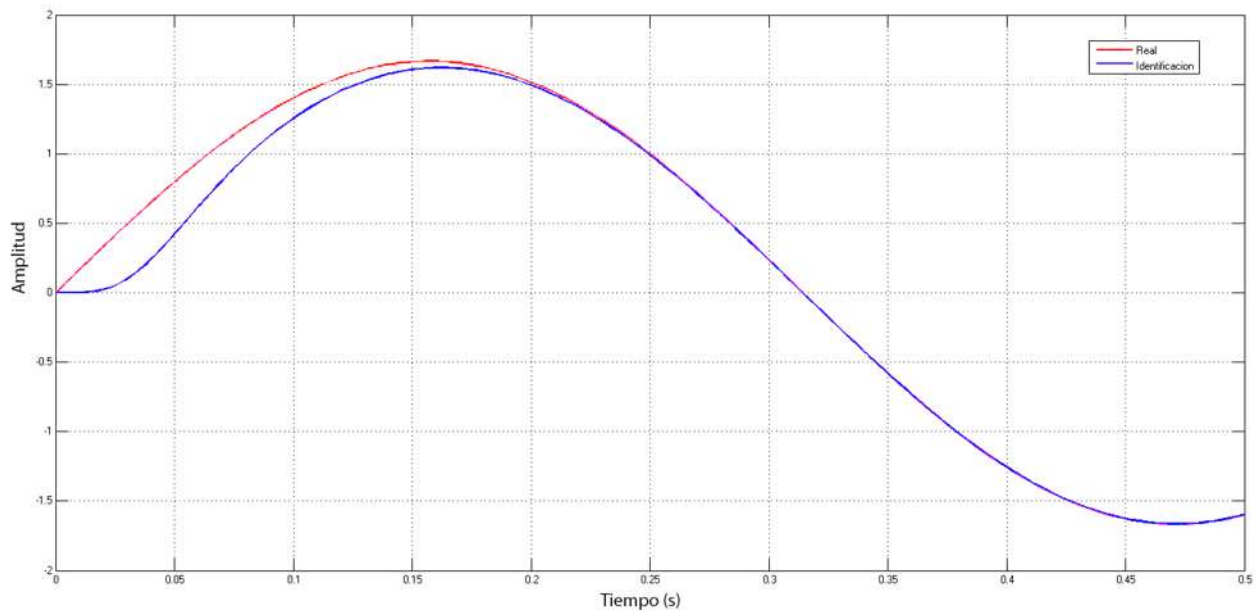


Figura 4.6: Kalman Bucy simulación un ciclo

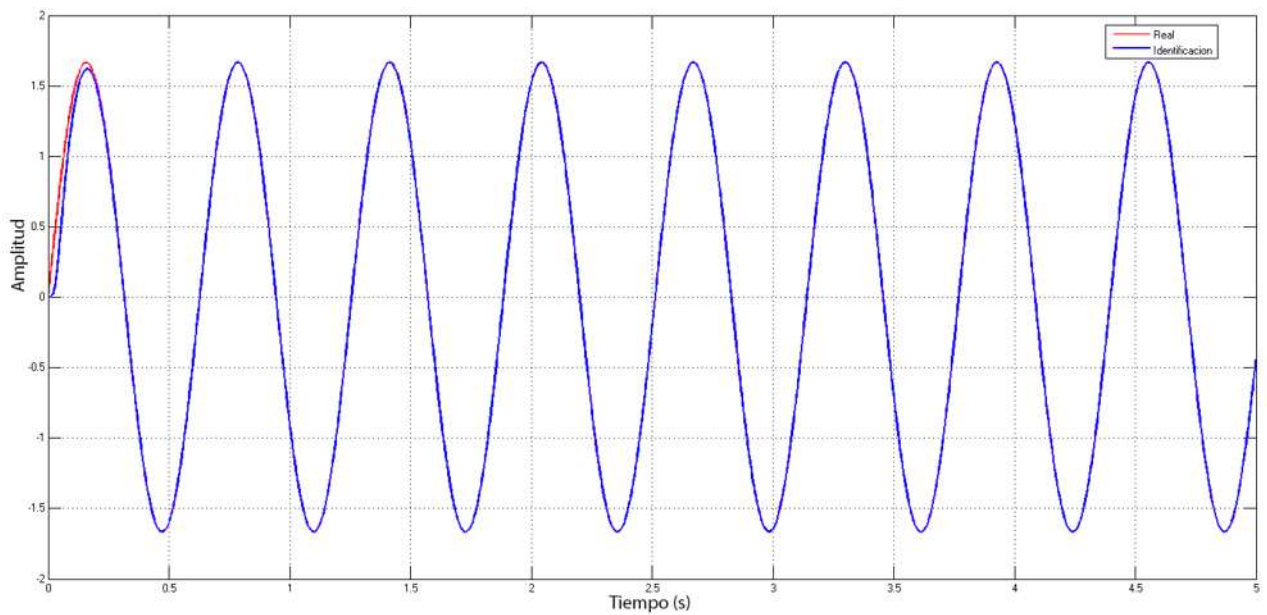


Figura 4.7: Kalman Bucy simulación tiempo extendido

4.3. Simulación control por modos deslizantes de primer orden

De acuerdo a la Sección 3.6, se procede de la misma manera, el objetivo de control es poder realizar seguimiento de trayectoria para ambos ejes (X e Y). Para demostrar esto se introducen señales de referencia distintas a cada grado de libertad (Figura 4.9 y Figura 4.10), el comportamiento de introducir ambas señales simultáneas es mostrado en la Figura 4.8 al introducir una señal de control por modos deslizantes de primer orden a cada grado de libertad.

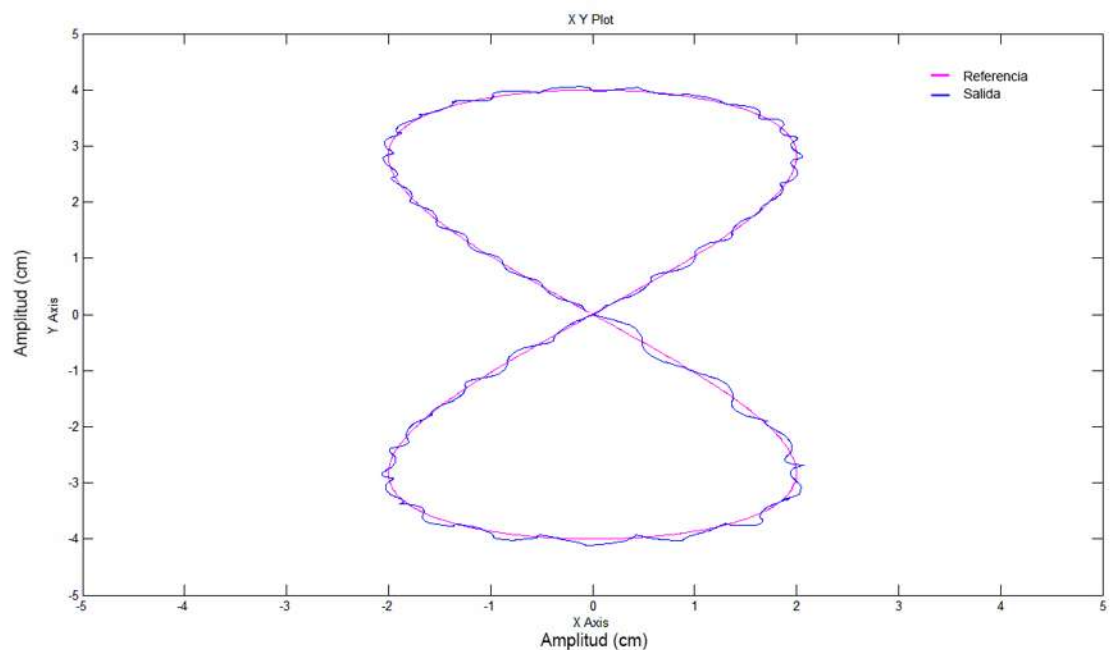


Figura 4.8: Salida del sistema en ambos ejes x,y

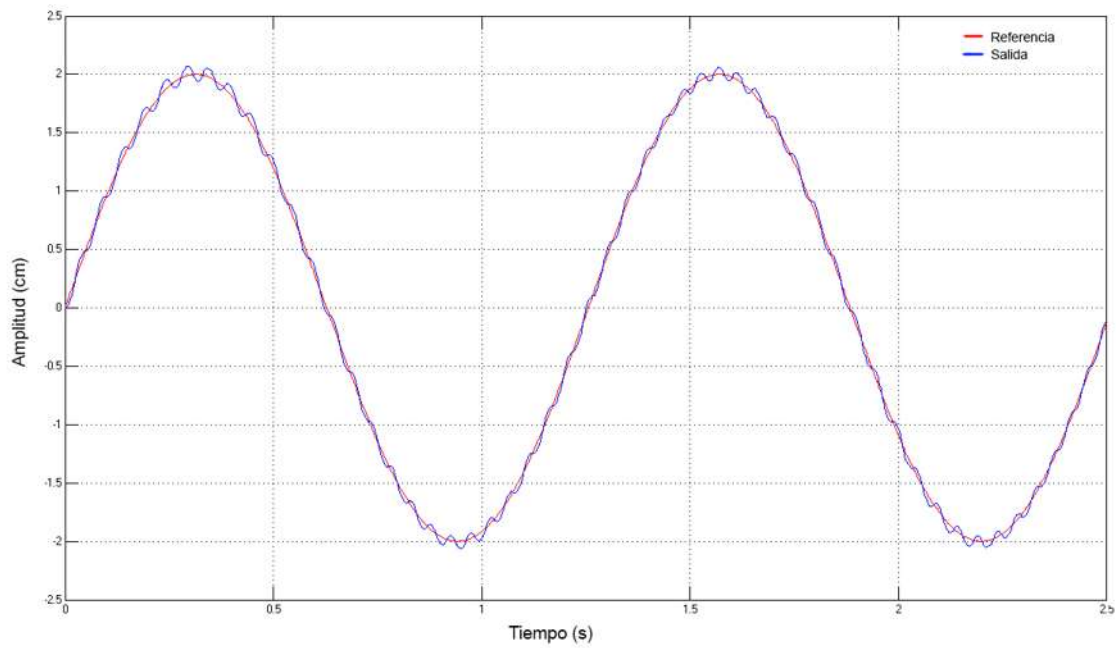


Figura 4.9: Referencia y salida del sistema para el eje x

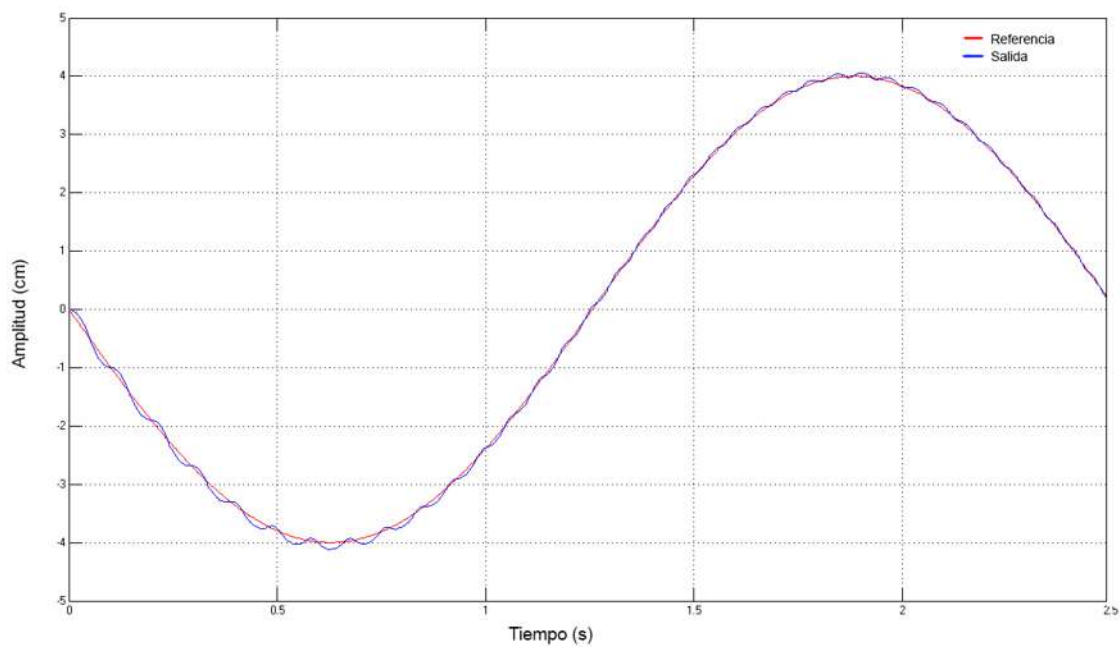


Figura 4.10: Referencia y salida del sistema para el eje y

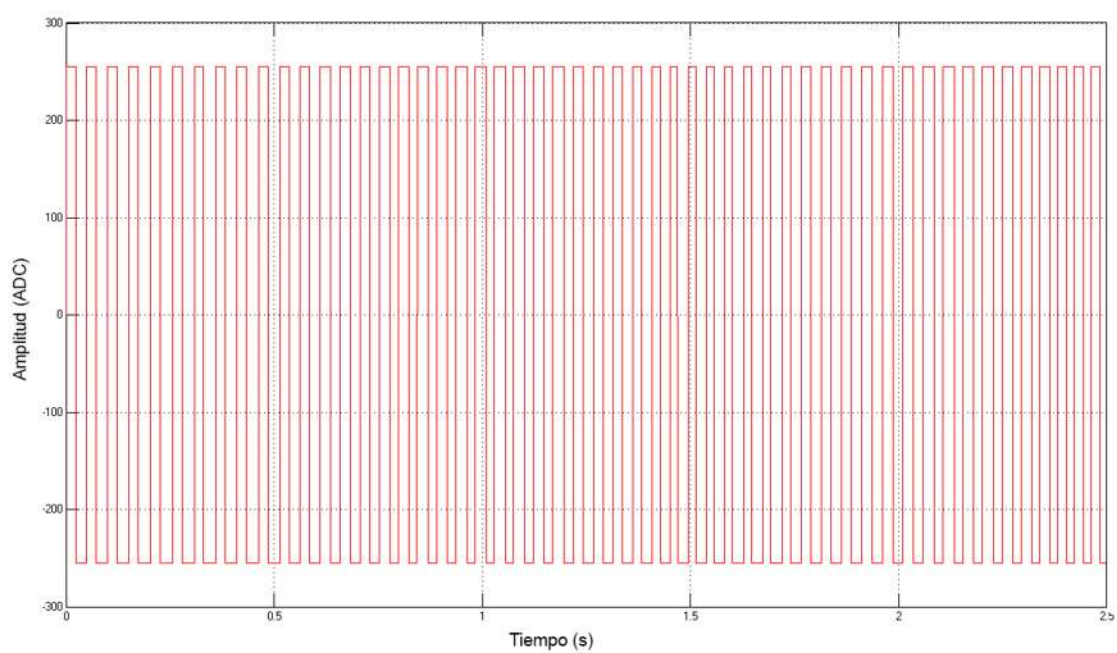


Figura 4.11: Señal de control para el eje x

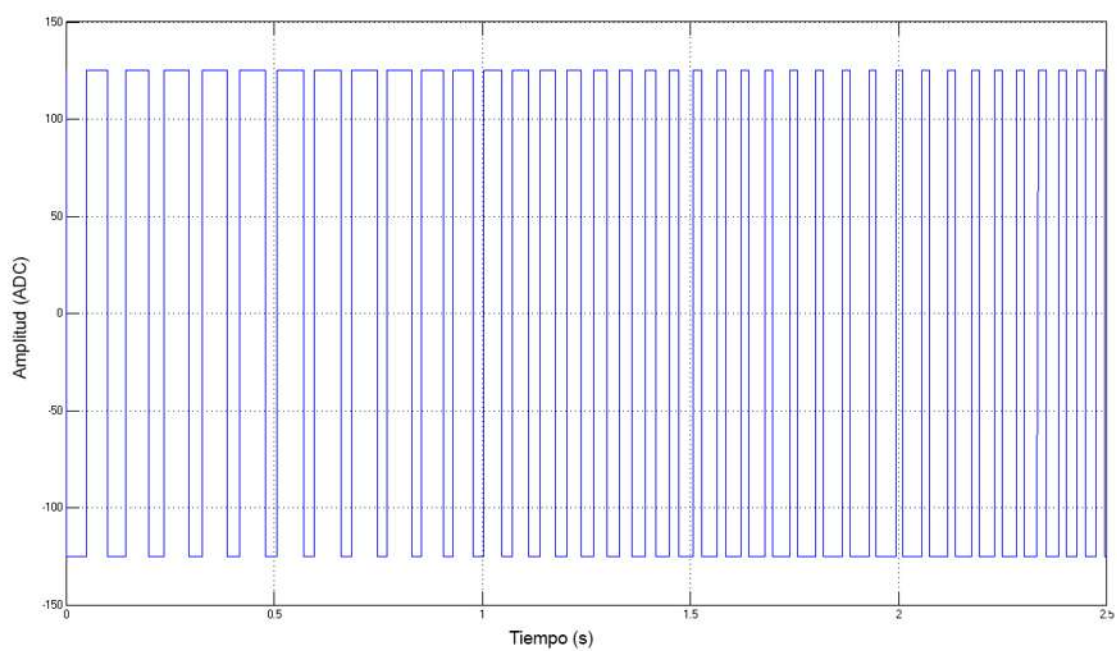


Figura 4.12: Señal de control para el eje y

El objetivo de introducir distintas señales a cada grado de libertad es hacer mover al sistema en un rango tanto positivo y negativo de ambos ejes o grados de libertad, así como a diferente frecuencia para demostrar la adaptabilidad de la acción de control.

La acción de control para cada grado de libertad es mostrada en la Figura 4.11 y Figura 4.12 respectivamente. Se observa la conmutación a alta frecuencia causante del chattering.

4.4. Simulación control por modos deslizantes de orden superior

De acuerdo a la Sección 3.7, se procede de la misma manera, el objetivo de control es poder realizar seguimiento de trayectoria para ambos ejes (X e Y). Para demostrar esto se introducen señales de referencia distintas a cada grado de libertad (Figura 4.14 y Figura 4.15), el comportamiento de introducir ambas señales simultáneas es mostrado en la Figura 4.13 al introducir una señal de control por modos deslizantes de primer orden a cada grado de libertad.

A diferencia de la sección anterior la simulación se alarga a 10 segundos con el objetivo de mostrar la reducción del chattering y del error conforme pasa el tiempo debido a la acción integral del control SuperTwisting, para ello se observa la figura principal remarcada debido al recorrido de la trayectoria de la masa.

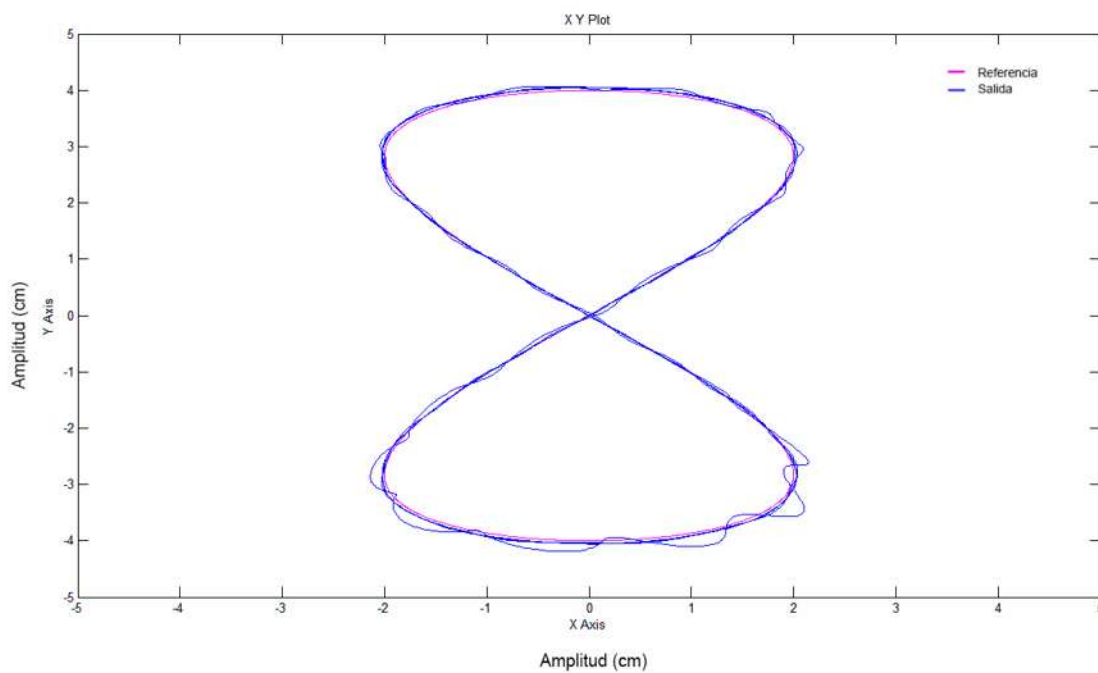


Figura 4.13: Salida del sistema en ambos ejes x,y con una acción de control SuperTwisting

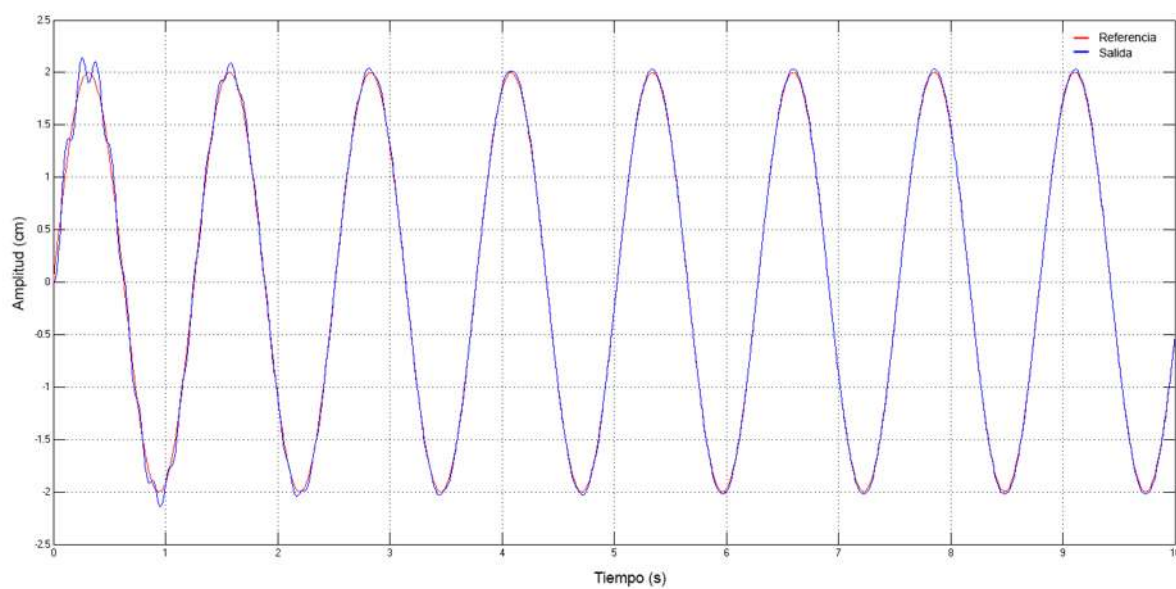


Figura 4.14: Referencia y salida del sistema para el eje x con una acción de control SuperTwisting

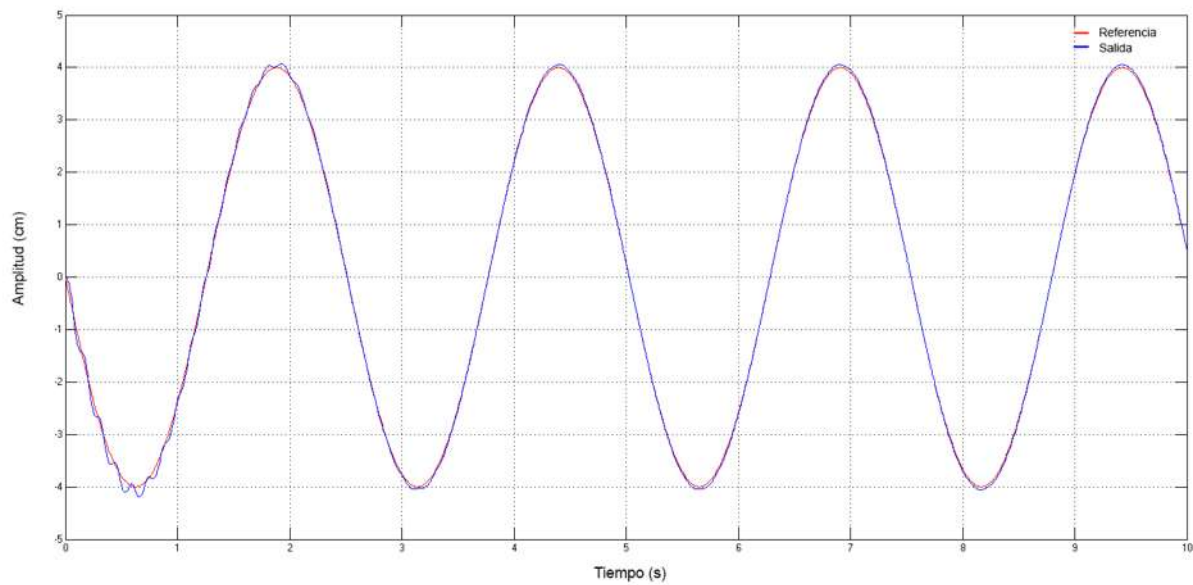


Figura 4.15: Referencia y salida del sistema para el eje y con una acción de control SuperTwisting

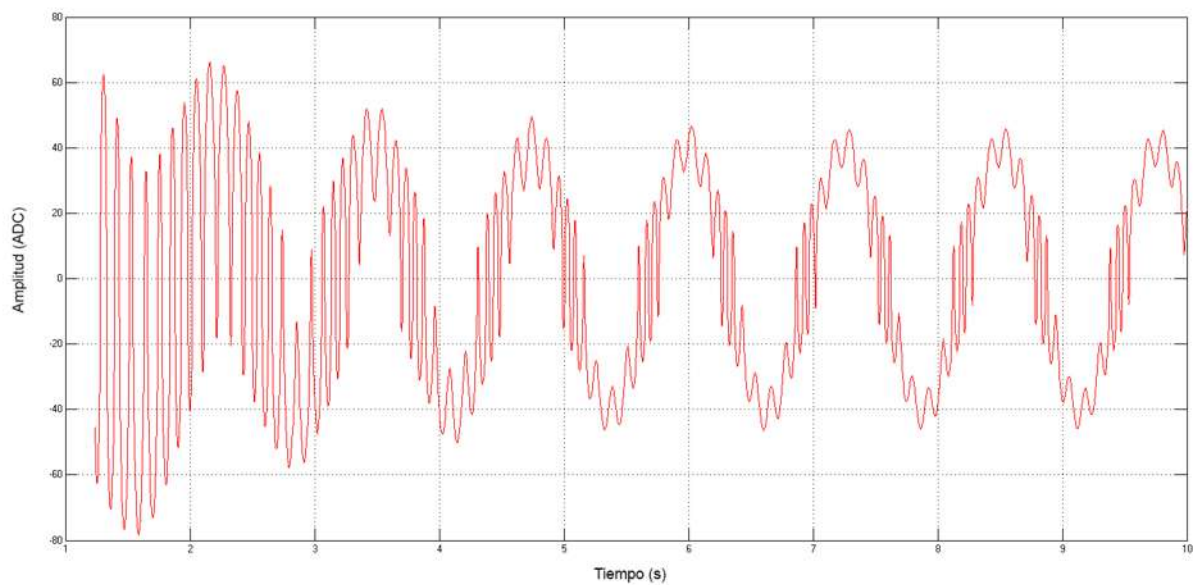


Figura 4.16: Señal de control SuperTwisting para el eje x

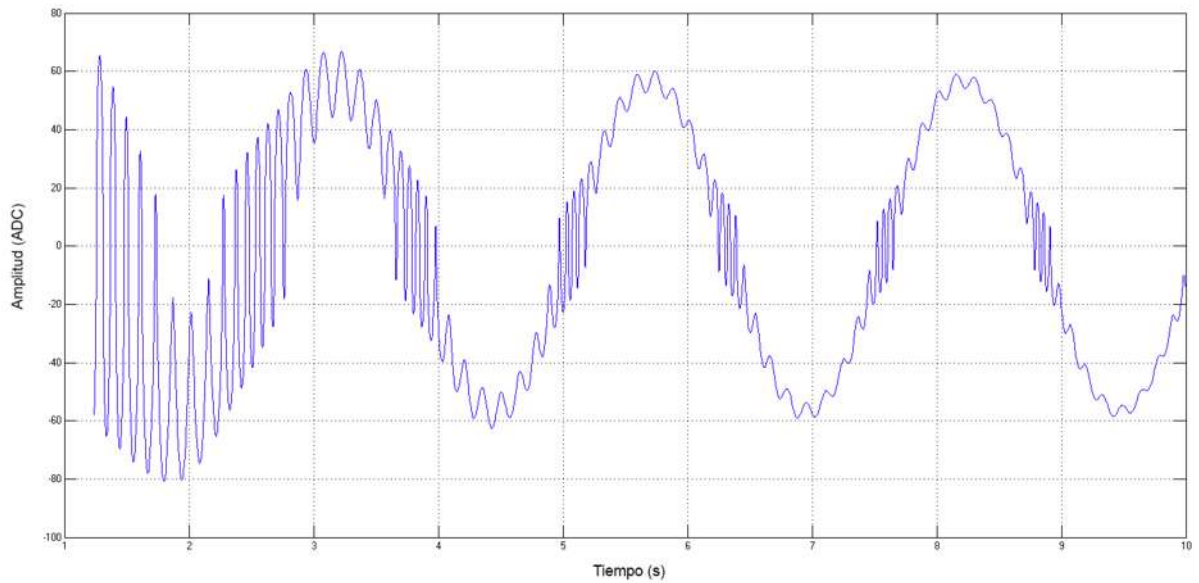


Figura 4.17: Señal de control SuperTwisting para el eje y

El objetivo de introducir distintas señales a cada grado de libertad es hacer mover al sistema en un rango tanto positivo y negativo de ambos ejes o grados de libertad, así como a diferente frecuencia para demostrar la adaptabilidad de la acción de control.

La acción de control para cada grado de libertad es mostrada en la Figura 4.16 y Figura 4.17 respectivamente. Se observa en la Figura 4.13 la disminución continua del chattering conforme aumenta el paso del tiempo respecto a la Figura 4.8. Esto se observa más claramente en la Figura 4.16 y la Figura 4.17 donde se observa que la acción de control es menos agresiva.

4.4.1. Conclusiones

De acuerdo a los 3 métodos analizados para la estimación paramétrica, el método de mínimos cuadrados es una implementación realizada fuera de línea, como el sistema cambia conforme transcurre el tiempo, es necesario aplicar un método como lo son los mínimos cuadrados recurrentes que es por mucho mejor, ya que logra estimar con un error mínimo ya que es un algoritmo adaptativo en línea. El principal problema con este algoritmo es que está en tiempo discreto, si bien la gran mayoría de dispositivos como microcontroladores

operan bajo tiempo discreto, las velocidades de relojes son altas alrededor de 20 MHz y tener un paso fijo sería mermar la capacidad del dispositivo. Finalmente se muestra la capacidad del algoritmo KBF, el cual es un algoritmo recursivo en línea adaptivo y en tiempo continuo, lo cual lo hace Excel[®] ante para la implementación en un dispositivo. Como observamos los 3 algoritmos cumplieron su función la cual fue estimar los parámetros del sistema con un error mínimo aun en presencia de ruido gaussiano. En la parte del control se observa que ambas técnicas cumplen con el objetivo de controlar la planta y realizar el seguimiento de trayectoria, como se esperaba la técnica del “SuperTwisting” tiene un mejor desempeño ya que la acción de control es menos agresiva, si bien esto se tiene que tener en consideración ya que afecta directamente en los actuadores en una implementación como se analizó en el capítulo anterior.

Capítulo 5

Conclusiones

5.1. Conclusiones generales

En esta tesis se presentó el modelado matemático de un sistema de levitación magnética de un grado de libertad tanto para masas con propiedades magnéticas como aquellas que no tienen esa propiedad, esto con el objetivo de aplicar técnicas de control.

Se realizó la implementación física lo cual considero la creación del electroimán, la instrumentación y caracterización de los sensores, los elementos actuadores y el microcontrolador. Se procedió a realizar la eliminación del ruido en los sensores haciendo uso del Filtro de Kalman.

Se realizó la estimación paramétrica en simulink y embebida del sistema usando de mínimos cuadrados y Filtro de Kalman para la validación del modelado matemático, posteriormente se realizó la estimación del parámetro de la fuerza electromagnética usando redes neuronales a nivel simulación e implementación en el microcontrolador.

El control del sistema se hace bajo dos técnicas de control no convencional- modos deslizantes de primer orden y modos deslizantes de orden superior “supertwisting”, se realizó una comparativa entre Embebido vs Matlab ® se determino porqué estos controles mencionados son los únicos que funcionan en este tipo de plantas donde la acción de control debe de ser rápida.

Se extendió el estudio a un sistema de levitación magnética de dos grados de libertad donde se realizó la estimación paramétrica del sistema mediante tres técnicas (Mínimos cuadrados, mínimos cuadrados recursivos, y Filtro de Kalman-Bucy). Finalmente se hizo el diseño de controladores robustos para el seguimiento de trayectoria.

Para la implementación del sistema de dos grados de libertad se realizaron varias pruebas, las cuales determinaron que la acción de control debe ser de forma paralela a ambos electroimanes (eje x - eje y) debido a que si no están alineados en frecuencia la acción de control no es efectiva, teniendo el mayor problema en esta sección ya que actualmente debido a la necesidad de ejecución en paralelo, para lo cual se debe implementar en dos distintos elementos controladores sincronizados, lo cual nos lleva que Matlab ® no es una opción considerable como se ha visto a lo largo de todo este trabajo.

5.2. Trabajos futuros

1. Realizar la implementación del sistema de dos grados de libertad.
2. Implementar en un microcontrolador con arquitectura ARM de 32bits para aumentar el desempeño y mantener costos bajos.
3. Extrapolar el estudio del caso a un aerogenerador con fines de reducción de fricción debido a levitación magnética.
4. Realizar pruebas de consumo energético vs generación de energía, haciendo uso de levitación magnética para la eliminación de la fricción.

Apéndice A

Codigos para implementación

A.1. Filtro de Kalman

```
#define HALL A7

class Khaman{
public:

float VarSensor = 1.5e-5;//0.3371;// 1e-5;//varianza del sensor a determinar
float VarProcess = 1e-7;/* Varianza del proceso pequeña en caso de que no varie dejarlo bajo,
para respuesta mas rapidas al cambio en el valor del sensor se incrementa esta variaicon mayores al cambio de */
float P;
float Pc;
float G;
float Xp;
float Zp;
float Xe;

float sensorValue;
float Estimate;

Khaman(float _P, float _Pc, float _G,float _Xp,float _Zp,float _Xe){
    P = _P;
    Pc = _Pc;
    G = _G;
    Xp = _Xp;
    Zp = _Zp;
    Xe = _Xe;
}

void addNewSample(double _Sample){
    sensorValue = _Sample;
}

double process(){

Pc = P + VarProcess;
G = Pc/(Pc + VarSensor);    // kalman gain
P = (1-G)*Pc;
Xp = Xe;
```

```
Zp = Xp;
Xe = G*(sensorValue-Zp)+Xp;
Estimate = Xe;
return Estimate;
}
};

// 0.1 1 y 10
Khalman KhalmanHall(1,0,0,0,0,0);

void setup(){
  Serial.begin(115200);
}

void loop() {

  float LectReal = analogRead(HALL);
  KhalmanHall.addNewSample(LectReal);
  float KhalmanRes = KhalmanHall.process();

  Serial.print(LectReal);
  Serial.print(" ");
  Serial.println(KhalmanRes);
}
```

A.2. Control con modos deslizantes

```
class TWISTING{
public:

    double error;
    double sample;
    double K;
    double U;

    double setPoint;
    long lastProcess;

    TWISTING(double _k){
        K = _k;
    }

    void addNewSample(double _sample){
        sample = _sample;
    }

    void setSetPoint(double _setPoint){
        setPoint = _setPoint;
    }

    double process(){

        error = setPoint - sample;

        int signo = 0;

        if(error >0){
            signo = 1;
        }

        if(error ==0){
            signo = 0;
        }

        if(error <0){
            signo = -1;
        }

        U = K*signo;

        return U;
    }
};

int IN3 = 5;
int IN4 = 4;
int ENB = 3;

// 0.1 1 y 10
TWISTING STLev(255);

void setup(){
```

```
Serial.begin(230400);
pinMode (ENB, OUTPUT);
pinMode (IN3, OUTPUT);
pinMode (IN4, OUTPUT);

}

void loop() {

int controlePwm;
STLev.setSetPoint(852);

int sensorValue = analogRead(A7);
// Añadimos la muestra
STLev.addNewSample(sensorValue);

// Mandamos el valor al PWM
controlePwm = STLev.process();
if (controlePwm < 0) controlePwm = 0;
if (controlePwm > 255) controlePwm = 255;

ActivaElectro(controlePwm);
Serial.print(STLev.error);
Serial.print(" ");
Serial.println(controlePwm);

}

void ActivaElectro(int PWM){
digitalWrite (IN3, HIGH);
digitalWrite(IN4, LOW);
analogWrite (ENB, PWM);
}
```

A.3. Control Supertwisting

```

class STWISTING{
public:

    double error;
    double sample;
    double lastSample;
    double ALPHA, LAMDA;
    double U, U1;

    double setPoint;
    long lastProcess;

    STWISTING(double _ALPHA, double _LAMDA){
        ALPHA = _ALPHA;
        LAMDA = _LAMDA;
    }

    void addNewSample(double _sample){
        sample = _sample;
    }

    void setSetPoint(double _setPoint){
        setPoint = _setPoint;
    }

    double process(){

        error = setPoint - sample;
        float deltaTime = (millis() - lastProcess) / 1000.0;
        lastProcess = millis();

        int signo = 0;
        //P
        if(error >0){
            signo = 1;
        }

        if(error ==0){
            signo = 0;
        }

        if(error <0){
            signo = -1;
        }

        U1= U1 -ALPHA*signo*deltaTime;
        U = LAMDA*sqrt(abs(error))*signo + U1 ;

        //I

        return U;
    }
};

int IN3 = 5;
int IN4 = 4;
int ENB = 3;

```

```
// 0.1 1 y 10
STWISTING STLev(10, 255);

void setup(){
  Serial.begin(230400);
  pinMode (ENB, OUTPUT);
  pinMode (IN3, OUTPUT);
  pinMode (IN4, OUTPUT);
}

void loop() {

  int controlePwm;
  STLev.setSetPoint(845);

  int sensorValue = analogRead(A7);
  // Añadimos la muestra
  STLev.addNewSample(sensorValue);

  // Hacemos el proceso del PID
  // Mandamos el valor al PWM
  controlePwm = STLev.process();
  if (controlePwm < 0) controlePwm = 0;
  if (controlePwm > 255) controlePwm = 255;
  ActivaElectro(controlePwm);

  Serial.print(STLev.error);
  Serial.print(",");
  Serial.print("\t");

  Serial.print(controlePwm);
  Serial.print(",");
  Serial.print("\t");

  Serial.print(STLev.lastProcess);
  Serial.println(";");

}

void ActivaElectro(int PWM){
  digitalWrite (IN3, HIGH);
  digitalWrite(IN4, LOW);
  analogWrite (ENB, PWM);
}
```

A.4. Control con modos deslizantes y filtro de Kalman

```

#define HALL A7
#define IN3 5
#define IN4 4
#define ENB 3

class Khalman{
public:

    float VarSensor = 1.5e-5;//0.3371;// 1e-5;//varianza del sensor a determinar
    float VarProcess = 1e-7;/* Varianza del proceso pequeña en caso de que no varie dejarlo bajo,
    para respuesta mas rapidas al cambio en el valor del sensor se incrementa esta variaicon mayores al cambio d
    float P;
    float Pc;
    float G;
    float Xp;
    float Zp;
    float Xe;

    float sensorValue;
    float Estimate;

    Khalman(float _P, float _Pc, float _G,float _Xp,float _Zp,float _Xe){
        P = _P;
        Pc = _Pc;
        G = _G;
        Xp = _Xp;
        Zp = _Zp;
        Xe = _Xe;
    }

    void addNewSample(double _Sample){
        sensorValue = _Sample;
    }

    double process(){

        Pc = P + VarProcess;
        G = Pc/(Pc + VarSensor);    // kalman gain
        P = (1-G)*Pc;
        Xp = Xe;
        Zp = Xp;
        Xe = G*(sensorValue-Zp)+Xp;
        Estimate = Xe;
        return Estimate;
    }
};

class TWISTING{
public:

    double error;
    double sample;
    double lastSample;
    double K;
    double U;

    double setPoint;
    long lastProcess;

```

```
    TWISTING(double _k){
        K = _k;
    }

    void addNewSample(double _sample){
        sample = _sample;
    }

    void setSetPoint(double _setPoint){
        setPoint = _setPoint;
    }

    double process(){

        error = setPoint - sample;

        int signo = 0;

        if(error >0){
            signo = 1;
        }

        if(error ==0){
            signo = 0;
        }

        if(error <0){
            signo = -1;
        }

        U = K*signo;

        return U;
    }
};

// 0.1 1 y 10
Khalman KhalmanHall(1,0,0,0,0,0);
TWISTING STLev(255);

void setup(){
    //
    Serial.begin(230400);
    pinMode (ENB, OUTPUT);
    pinMode (IN3, OUTPUT);
    pinMode (IN4, OUTPUT);
}

void loop() {

    float LectReal = analogRead(HALL);
    KhalmanHall.addNewSample(LectReal);
    float KhalmanRes = KhalmanHall.process();
```

```
int controlePwm;
STLev.setSetPoint(831);

int sensorValue = KhalmanRes;
// Añadimos la muestra
STLev.addNewSample(sensorValue);

// Mandamos el valor al PWM
controlePwm = STLev.process();
if (controlePwm < 0) controlePwm = 0;
if (controlePwm > 255) controlePwm = 255;
ActivaElectro(controlePwm);

Serial.print(STLev.error);
Serial.print(" ");
Serial.println(controlePwm);
}

void ActivaElectro(int PWM){
    digitalWrite (IN3, HIGH);
    digitalWrite(IN4, LOW);
    analogWrite (ENB, PWM);
}
```

A.5. Control SuperTwisting y filtro de Kalman

```

#define HALL A7
#define IN3 5
#define IN4 4
#define ENB 3

class Khaman{
public:

    float VarSensor = 1.5e-5;//0.3371;// 1e-5;//varianza del sensor a determinar
    float VarProcess = 1e-7;/* Varianza del proceso pequeña en caso de que no varie dejarlo bajo,
    para respuesta mas rapidas al cambio en el valor del sensor se incrementa esta variaicon mayores al cambio de */
    float P;
    float Pc;
    float G;
    float Xp;
    float Zp;
    float Xe;

    float sensorValue;
    float Estimate;

    Khaman(float _P, float _Pc, float _G,float _Xp,float _Zp,float _Xe){
        P = _P;
        Pc = _Pc;
        G = _G;
        Xp = _Xp;
        Zp = _Zp;
        Xe = _Xe;
    }

    void addNewSample(double _Sample){
        sensorValue = _Sample;
    }

    double process(){

        Pc = P + VarProcess;
        G = Pc/(Pc + VarSensor);    // kalman gain
        P = (1-G)*Pc;
        Xp = Xe;
        Zp = Xp;
        Xe = G*(sensorValue-Zp)+Xp;
        Estimate = Xe;
        return Estimate;
    }
};

class STWISTING{
public:

    double error;
    double sample;
    double lastSample;
    double ALPHA, LAMDA;
    double U, U1;

    double setPoint;
    long lastProcess;

```

```

STWISTING(double _ALPHA, double _LAMDA){
    ALPHA = _ALPHA;
    LAMDA = _LAMDA;
}

void addNewSample(double _sample){
    sample = _sample;
}

void setSetPoint(double _setPoint){
    setPoint = _setPoint;
}

double process(){

    error = setPoint - sample;
    float deltaTime = (millis() - lastProcess) / 1000.0;
    lastProcess = millis();

    int signo = 0;
    //P
    if(error >0){
        signo = 1;
    }

    if(error ==0){
        signo = 0;
    }

    if(error <0){
        signo = -1;
    }

    U1= U1 -ALPHA*signo*deltaTime;
    U = LAMDA*sqrt(abs(error))*signo + U1 ;

    //I

    return U;
}
};

// 0.1 1 y 10
Khalman KhalmanHall(1,0,0,0,0,0);
STWISTING STLev(10, 255);

void setup(){
    //230400
    Serial.begin(230400);
    pinMode (ENB, OUTPUT);
    pinMode (IN3, OUTPUT);
    pinMode (IN4, OUTPUT);
}

void loop() {

```

```
float LectReal = analogRead(HALL);
KhalmanHall.addNewSample(LectReal);
float KhalmanRes = KhalmanHall.process();

int controlePwm;
STLev.setSetPoint(840);

int sensorValue = KhalmanRes;
// Añadimos la muestra
STLev.addNewSample(sensorValue);

// Mandamos el valor al PWM
controlePwm = STLev.process();
if (controlePwm < 0) controlePwm = 0;
if (controlePwm > 255) controlePwm = 255;
ActivaElectro(controlePwm);

Serial.print(STLev.error);
Serial.print(" ");
Serial.println(controlePwm);
}

void ActivaElectro(int PWM){
    digitalWrite (IN3, HIGH);
    digitalWrite (IN4, LOW);
    analogWrite (ENB, PWM);
}
```

A.6. Red neuronal embebida

```

#include <math.h>

const int PatternCount = 2;
const int InputNodes = 6;
const int HiddenNodes = 1;
const int OutputNodes = 6;
const float LearningRate = 0.3;
const float Momentum = 0.9;
const float InitialWeightMax = 0.5;
const float Success = 0.0004;

const float Input[PatternCount][InputNodes] = {
  { 0.018, 0.02, 0.022, 0.023, 0.024, 0.025 }, // 0
  { 0.3651 , 0.360261624, 0.355456276, 0.35068392, 0.34594452, 0.341238039 }, // 1
};

const float Target[PatternCount][OutputNodes] = {
  { 1.6557, 2.6717, 3.6819, 4.6001, 5.5126, 6.4194 },
};

/*****
 * End Network Configuration
 *****/

int i, j, p, q, r;
int ReportEvery1000;
int RandomizedIndex[PatternCount];
long TrainingCycle;
float Rando;
float Error;
float Accum;

float Hidden[HiddenNodes];
float Output[OutputNodes];
float HiddenWeights[InputNodes+1][HiddenNodes];
float OutputWeights[HiddenNodes+1][OutputNodes];
float HiddenDelta[HiddenNodes];
float OutputDelta[OutputNodes];
float ChangeHiddenWeights[InputNodes+1][HiddenNodes];
float ChangeOutputWeights[HiddenNodes+1][OutputNodes];

void setup(){
  Serial.begin(9600);
  randomSeed(analogRead(3));
  ReportEvery1000 = 1;
  for( p = 0 ; p < PatternCount ; p++ ) {
    RandomizedIndex[p] = p ;
  }
}

void loop (){

/*****
 * Initialize HiddenWeights and ChangeHiddenWeights
 *****/

```

```

for( i = 0 ; i < HiddenNodes ; i++ ) {
    for( j = 0 ; j <= InputNodes ; j++ ) {
        ChangeHiddenWeights[j][i] = 0.0 ;
        Rando = float(random(100))/100;
        HiddenWeights[j][i] = 2.0 * ( Rando - 0.5 ) * InitialWeightMax ;
    }
}
/*****
* Initialize OutputWeights and ChangeOutputWeights
*****/

for( i = 0 ; i < OutputNodes ; i ++ ) {
    for( j = 0 ; j <= HiddenNodes ; j++ ) {
        ChangeOutputWeights[j][i] = 0.0 ;
        Rando = float(random(100))/100;
        OutputWeights[j][i] = 2.0 * ( Rando - 0.5 ) * InitialWeightMax ;
    }
}
Serial.println("Initial/Untrained Outputs: ");
toTerminal();
/*****
* Begin training
*****/

for( TrainingCycle = 1 ; TrainingCycle < 2147483647 ; TrainingCycle++ ) {

/*****
* Randomize order of training patterns
*****/

    for( p = 0 ; p < PatternCount ; p++ ) {
        q = random(PatternCount);
        r = RandomizedIndex[p] ;
        RandomizedIndex[p] = RandomizedIndex[q] ;
        RandomizedIndex[q] = r ;
    }
    Error = 0.0 ;
/*****
* Cycle through each training pattern in the randomized order
*****/
    for( q = 0 ; q < PatternCount ; q++ ) {
        p = RandomizedIndex[q];

/*****
* Compute hidden layer activations
*****/

        for( i = 0 ; i < HiddenNodes ; i++ ) {
            Accum = HiddenWeights[InputNodes][i] ;
            for( j = 0 ; j < InputNodes ; j++ ) {
                Accum += Input[p][j] * HiddenWeights[j][i] ;
            }
            Hidden[i] = 1.0/(1.0 + exp(-Accum)) ;
        }

/*****
* Compute output layer activations and calculate errors
*****/

        for( i = 0 ; i < OutputNodes ; i++ ) {
            Accum = OutputWeights[HiddenNodes][i] ;
            for( j = 0 ; j < HiddenNodes ; j++ ) {
                Accum += Hidden[j] * OutputWeights[j][i] ;
            }

```

```

    }
    Output[i] = 1.0/(1.0 + exp(-Accum)) ;
    OutputDelta[i] = (Target[p][i] - Output[i]) * Output[i] * (1.0 - Output[i]) ;
    Error += 0.5 * (Target[p][i] - Output[i]) * (Target[p][i] - Output[i]) ;
}

/*****
* Backpropagate errors to hidden layer
*****/

for( i = 0 ; i < HiddenNodes ; i++ ) {
    Accum = 0.0 ;
    for( j = 0 ; j < OutputNodes ; j++ ) {
        Accum += OutputWeights[i][j] * OutputDelta[j] ;
    }
    HiddenDelta[i] = Accum * Hidden[i] * (1.0 - Hidden[i]) ;
}

/*****
* Update Inner-->Hidden Weights
*****/

for( i = 0 ; i < HiddenNodes ; i++ ) {
    ChangeHiddenWeights[InputNodes][i] = LearningRate * HiddenDelta[i] + Momentum * ChangeHiddenWeights[InputNodes][i] ;
    HiddenWeights[InputNodes][i] += ChangeHiddenWeights[InputNodes][i] ;
    for( j = 0 ; j < InputNodes ; j++ ) {
        ChangeHiddenWeights[j][i] = LearningRate * Input[p][j] * HiddenDelta[i] + Momentum * ChangeHiddenWeights[j][i] ;
        HiddenWeights[j][i] += ChangeHiddenWeights[j][i] ;
    }
}

/*****
* Update Hidden-->Output Weights
*****/

for( i = 0 ; i < OutputNodes ; i++ ) {
    ChangeOutputWeights[HiddenNodes][i] = LearningRate * OutputDelta[i] + Momentum * ChangeOutputWeights[HiddenNodes][i] ;
    OutputWeights[HiddenNodes][i] += ChangeOutputWeights[HiddenNodes][i] ;
    for( j = 0 ; j < HiddenNodes ; j++ ) {
        ChangeOutputWeights[j][i] = LearningRate * Hidden[j] * OutputDelta[i] + Momentum * ChangeOutputWeights[j][i] ;
        OutputWeights[j][i] += ChangeOutputWeights[j][i] ;
    }
}

/*****
* Every 1000 cycles send data to terminal for display
*****/
ReportEvery1000 = ReportEvery1000 - 1;
if (ReportEvery1000 == 0)
{
    Serial.println();
    Serial.println();
    Serial.print ("TrainingCycle: ");
    Serial.print (TrainingCycle);
    Serial.print (" Error = ");
    Serial.println (Error, 5);

    toTerminal();

    if (TrainingCycle==1)

```

```

    {
        ReportEvery1000 = 999;
    }
    else
    {
        ReportEvery1000 = 1000;
    }
}

/*****
* If error rate is less than pre-determined threshold then end
*****/

    if( Error < Success ) break ;
}
Serial.println ();
Serial.println ();
Serial.print ("TrainingCycle: ");
Serial.print (TrainingCycle);
Serial.print (" Error = ");
Serial.println (Error, 5);

toTerminal();

Serial.println ();
Serial.println ();
Serial.println ("Training Set Solved! ");
Serial.println ("-----");
Serial.println ();
Serial.println ();
ReportEvery1000 = 1;
}

void toTerminal()
{

    for( p = 0 ; p < PatternCount ; p++ ) {
        Serial.println();
        Serial.print (" Training Pattern: ");
        Serial.println (p);
        Serial.print (" Input ");
        for( i = 0 ; i < InputNodes ; i++ ) {
            Serial.print (Input[p][i], DEC);
            Serial.print (" ");
        }
        Serial.print (" Target ");
        for( i = 0 ; i < OutputNodes ; i++ ) {
            Serial.print (Target[p][i], DEC);
            Serial.print (" ");
        }
    }

/*****
* Compute hidden layer activations
*****/

    for( i = 0 ; i < HiddenNodes ; i++ ) {
        Accum = HiddenWeights[InputNodes][i] ;
        for( j = 0 ; j < InputNodes ; j++ ) {
            Accum += Input[p][j] * HiddenWeights[j][i] ;
        }
        Hidden[i] = 1.0/(1.0 + exp(-Accum)) ;
    }
}

```

```

/*****
* Compute output layer activations and calculate errors
*****/

    for( i = 0 ; i < OutputNodes ; i++ ) {
        Accum = OutputWeights[HiddenNodes][i] ;
        for( j = 0 ; j < HiddenNodes ; j++ ) {
            Accum += Hidden[j] * OutputWeights[j][i] ;
        }
        Output[i] = 1.0/(1.0 + exp(-Accum)) ;
    }
    Serial.print ( "  Output " );
    for( i = 0 ; i < OutputNodes ; i++ ) {
        Serial.print (Output[i], 5);
        Serial.print ( " ");
    }
}

```

A.7. Reconstrucción red neuronal

```

double w1 = 0.1268;
double w2 = -0.8743;
double bias = 0.0011;
double Pos[]={0.018, 0.020, 0.022, 0.023, 0.024, 0.025};
double Corriente[]={0.365100000, 0.360261624, 0.355456276, 0.350683920, 0.34594452, 0.341238039};

int i = 0;
void setup(){
  Serial.begin(9600);
}

void loop(){

  double RedNeuronal = RED(w1,w2,bias,Pos[i],Corriente[i]);
  Serial.println(RedNeuronal);
  i++;

  if(i >=6){
    i = 0;
  }
  delay(1000);
}

double ConvEntrada1(double Entrada1){
  double maxx = 1;
  double minn = -1;

  double xmaxx = 0.0250;
  double xminn = 0.0180;

  double Scala = (maxx - minn)/(xmaxx - xminn) *(Entrada1 - xminn) + minn;
  return  Scala;
}

double ConvEntrada2(double Entrada2){
  double maxx = 1;
  double minn = -1;

  double xmaxx = 0.3651;
  double  xminn = 0.3412;

  double Scala = (maxx - minn)/(xmaxx - xminn) *(Entrada2 - xminn) + minn;
  return  Scala;
}

double ConvSalida(double z){

  double maxx = 6.4143;
  double minn = 1.6557;

  double xmaxx = 1;
  double xminn = -1;

```

```
double Scala = (maxx - minn)/(xmaxx - xminn) *(z - xminn) + minn;
return  Scala;

}

double RED (double w1, double w2, double bias, double Ent1, double Ent2){
    double Entrada1 = ConvEntrada1(Ent1);
    double Entrada2 = ConvEntrada2(Ent2);
    double Z = w1*Entrada1 + w2*Entrada2 + bias;
    double Salida = ConvSalida(Z);

    return Salida;
}
```


Apéndice B

Vectores de identificación

B.1. Vectores de identificación del sistema con un conversor ADC de 8 bits

Muestra	U(Voltaje)	i(Corriente)	idot(Derivada de laCorriente)
1	0	0	0
2	0.0481960784313725	0.0103755923438195	-0.0566892733564014
3	0.0963921568627451	0.00927690330265132	-0.0553767012687428
4	0.144588235294118	0.00820396906167311	-0.0540622837370242
5	0.192784313725490	0.00715682580606252	-0.0527460207612457
6	0.240980392156863	0.00613550972099720	-0.0514279123414072
7	0.289176470588235	0.00514005699165479	-0.0501079584775087
8	0.337372549019608	0.00417050380321294	-0.0487861591695502
9	0.385568627450980	0.00322688634084930	-0.0474625144175317
10	0.433764705882353	0.00230924078974150	-0.0461370242214533
11	0.481960784313725	0.00141760333506720	-0.0448096885813149
12	0.530156862745098	0.000552010162004055	-0.0434805074971165
13	0.578352941176471	-0.000287502544270303	-0.0421494809688581
14	0.626549019607843	-0.00110089859857822	-0.0408166089965398
15	0.674745098039216	-0.00188814181574206	-0.0394818915801615
16	0.722941176470588	-0.00264919601058417	-0.0381453287197232
17	0.771137254901961	-0.00338402499792689	-0.0368069204152249
18	0.819333333333333	-0.00409259259259259	-0.0354666666666667
19	0.867529411764706	-0.00477486260940362	-0.0341245674740485
20	0.915725490196078	-0.00543079886318234	-0.0327806228373702
21	0.963921568627451	-0.00606036516875108	-0.0314348327566321
22	1.01211764705882	-0.00666352534093222	-0.0300871972318339
23	1.06031372549020	-0.00724024319454810	-0.0287377162629758
24	1.10850980392157	-0.00779048254442108	-0.0273863898500577
25	1.15670588235294	-0.00831420720537350	-0.0260332179930796
26	1.20490196078431	-0.00881138099222773	-0.0246782006920415
27	1.25309803921569	-0.00928196771980611	-0.0233213379469435
28	1.30129411764706	-0.00972593120293100	-0.0219626297577855
29	1.34949019607843	-0.0101432352564248	-0.0206020761245675
30	1.39768627450980	-0.0105338436951097	-0.0192396770472895

Tabla B.1: Valores identificación 1

Muestra	U(Voltaje)	i(Corriente)	idot(Derivada de laCorriente)
31	1.44588235294118	-0.0108977203338083	-0.0178754325259516
32	1.49407843137255	-0.0112348289873427	-0.0165093425605536
33	1.54227450980392	-0.0115451334705355	-0.0151414071510957
34	1.59047058823529	-0.0118285975982088	-0.0137716262975779
35	1.63866666666667	-0.0120851851851852	-0.0124000000000000
36	1.68686274509804	-0.0123148600462869	-0.0110265282583622
37	1.73505882352941	-0.0125175859963363	-0.00965121107266437
38	1.78325490196078	-0.0126933268501557	-0.00827404844290659
39	1.83145098039216	-0.0128420464225675	-0.00689504036908881
40	1.87964705882353	-0.0129637085283941	-0.00551418685121108
41	1.92784313725490	-0.0130582769824577	-0.00413148788927337
42	1.97603921568627	-0.0131257155995809	-0.00274694348327566
43	2.02423529411765	-0.0131659881945858	-0.00136055363321800
44	2.07243137254902	-0.0131790585822949	2.76816608996561e-05
45	2.12062745098039	-0.0131648905775305	0.00141776239907727
46	2.16882352941176	-0.0131234479951150	0.00280968858131488
47	2.21701960784314	-0.0130546946498707	0.00420346020761245
48	2.26521568627451	-0.0129585943566200	0.00559907727797000
49	2.31341176470588	-0.0128351109301852	0.00699653979238753
50	2.36160784313726	-0.0126842081853887	0.00839584775086506
51	2.40980392156863	-0.0125058499370529	0.00979700115340253
52	2.45800000000000	-0.0123000000000000	0.0112000000000000
53	2.50619607843137	-0.0120666221890525	0.0126048442906574
54	2.55439215686275	-0.0118056803190327	0.0140115340253749
55	2.60258823529412	-0.0115171382047629	0.0154200692041523
56	2.65078431372549	-0.0112009596610655	0.0168304498269896
57	2.69898039215686	-0.0108571085027629	0.0182426758938870
58	2.74717647058824	-0.0104855485446774	0.0196567474048443
59	2.79537254901961	-0.0100862436016313	0.0210726643598616
60	2.84356862745098	-0.00965915748844714	0.0224904267589389

Tabla B.2: Valores identificación 2

Muestra	U(Voltaje)	i(Corriente)	idot(Derivada de laCorriente)
61	2.89176470588235	-0.00920425401994708	0.0239100346020761
62	2.93996078431373	-0.00872149701095356	0.0253314878892733
63	2.98815686274510	-0.00821085027628891	0.0267547866205306
64	3.03635294117647	-0.00767227763077550	0.0281799307958477
65	3.08454901960784	-0.00710574288923567	0.0296069204152249
66	3.13274509803922	-0.00651120986649178	0.0310357554786620
67	3.18094117647059	-0.00588864237736618	0.0324664359861592
68	3.22913725490196	-0.00523800423668122	0.0338989619377163
69	3.27733333333333	-0.00455925925925927	0.0353333333333333
70	3.32552941176471	-0.00385237125992266	0.0367695501730104
71	3.37372549019608	-0.00311730405349378	0.0382076124567474
72	3.42192156862745	-0.00235402145479492	0.0396475201845444
73	3.47011764705882	-0.00156248727864849	0.0410892733564014
74	3.51831372549020	-0.000742665339876835	0.0425328719723183
75	3.56650980392157	0.000105480546697709	0.0439783160322953
76	3.61470588235294	0.000981986566252775	0.0454256055363322
77	3.66290196078431	0.00188688890396605	0.0468747404844291
78	3.71109803921569	0.00282022374501511	0.0483257208765859
79	3.75929411764706	0.00378202727457765	0.0497785467128028
80	3.80749019607843	0.00477233567783130	0.0512332179930796
81	3.85568627450980	0.00579118513995370	0.0526897347174164
82	3.90388235294118	0.00683861184612253	0.0541480968858131
83	3.95207843137255	0.00791465198151540	0.0556083044982699
84	4.00027450980392	0.00901934173130996	0.0570703575547866
85	4.04847058823529	0.0101527172806839	0.0585342560553633
86	4.09666666666667	0.0113148148148148	0.0600000000000000
87	4.14486274509804	0.0125056705188804	0.0614675893886967
88	4.19305882352941	0.0137253205780582	0.0629370242214533
89	4.24125490196078	0.0149738011775260	0.0644083044982699
90	4.28945098039216	0.0162511485024613	0.0658814302191465

Tabla B.3: Valores identificación 3

Muestra	U(Voltaje)	i(Corriente)	idot(Derivada de laCorriente)
91	4.33764705882353	0.0175573987380419	0.0673564013840831
92	4.38584313725490	0.0188925880694454	0.0688332179930796
93	4.43403921568627	0.0202567526818494	0.0703118800461361
94	4.48223529411765	0.0216499287604315	0.0717923875432526
95	4.53043137254902	0.0230721524903694	0.0732747404844291
96	4.57862745098039	0.0245234600568409	0.0747589388696655
97	4.62682352941176	0.0260038876450234	0.0762449826989619
98	4.67501960784314	0.0275134714400947	0.0777328719723183
99	4.72321568627451	0.0290522476272324	0.0792226066897347
100	4.77141176470588	0.0306202523916141	0.0807141868512111
101	4.81960784313726	0.0322175219184175	0.0822076124567474
102	4.86780392156863	0.0338440923928203	0.0837028835063437
103	4.91600000000000	0.0355000000000000	0.0852000000000000
104	4.96419607843137	0.0371852809251344	0.0866989619377163
105	5.01239215686274	0.0388999713534010	0.0881997693194925
106	5.06058823529412	0.0406441074699776	0.0897024221453287
107	5.10878431372549	0.0424177254600418	0.0912069204152249
108	5.15698039215686	0.0442208615087711	0.0927132641291811
109	5.20517647058824	0.0460535518013434	0.0942214532871973
110	5.25337254901961	0.0479158325229361	0.0957314878892734
111	5.30156862745098	0.0498077398587270	0.0972433679354094
112	5.34976470588235	0.0517293099938937	0.0987570934256055
113	5.39796078431373	0.0536805791136139	0.100272664359862
114	5.44615686274510	0.0556615834030652	0.101790080738178
115	5.49435294117647	0.0576723590474252	0.103309342560554
116	5.54254901960784	0.0597129422318716	0.104830449826990
117	5.59074509803922	0.0617833691415820	0.106353402537486
118	5.63894117647059	0.0638836759617342	0.107878200692042
119	5.68713725490196	0.0660138988775056	0.109404844290657
120	5.73533333333333	0.0681740740740741	0.110933333333333

Tabla B.4: Valores identificación 4

Muestra	U(Voltaje)	i(Corriente)	idot(Derivada de laCorriente)
121	5.78352941176471	0.0703642377366171	0.112463667820069
122	5.83172549019608	0.0725844260503125	0.113995847750865
123	5.87992156862745	0.0748346752003377	0.115529873125721
124	5.92811764705882	0.0771150213718705	0.117065743944637
125	5.97631372549020	0.0794255007500885	0.118603460207612
126	6.02450980392157	0.0817661495201695	0.120143021914648
127	6.07270588235294	0.0841370038672909	0.121684429065744
128	6.12090196078431	0.0865380999766304	0.123227681660900
129	6.16909803921569	0.0889694740333658	0.124772779700115
130	6.21729411764706	0.0914311622226745	0.126319723183391
131	6.26549019607843	0.0939232007297344	0.127868512110727
132	6.31368627450980	0.0964456257397230	0.129419146482122
133	6.36188235294118	0.0989984734378180	0.130971626297578
134	6.41007843137255	0.101581780009197	0.132525951557093
135	6.45827450980392	0.104195581639038	0.134082122260669
136	6.50647058823529	0.106839914512518	0.135640138408304
137	6.55466666666667	0.109514814814815	0.137200000000000
138	6.60286274509804	0.112220318731106	0.138761707035755
139	6.65105882352941	0.114956462446570	0.140325259515571
140	6.69925490196078	0.117723282146384	0.141890657439446
141	6.74745098039216	0.120520814015725	0.143457900807382
142	6.79564705882353	0.123349094239772	0.145026989619377
143	6.84384313725490	0.126208159003701	0.146597923875433
144	6.89203921568627	0.129098044492691	0.148170703575548
145	6.94023529411765	0.132018786891919	0.149745328719723
146	6.98843137254902	0.134970422386563	0.151321799307959
147	7.03662745098039	0.137952987161801	0.152900115340254
148	7.08482352941176	0.140966517402809	0.154480276816609
149	7.13301960784314	0.144011049294766	0.156062283737024
150	7.18121568627451	0.147086619022849	0.157646136101499

Tabla B.5: Valores identificación 5

Muestra	U(Voltaje)	i(Corriente)	idot(Derivada de laCorriente)
151	7.22941176470588	0.150193262772237	0.159231833910035
152	7.27760784313725	0.153331016728106	0.160819377162630
153	7.32580392156863	0.156499917075635	0.162408765859285
154	7.37400000000000	0.159700000000000	0.164000000000000
155	7.42219607843137	0.162931301686380	0.165593079584775
156	7.47039215686274	0.166193858319952	0.167188004613610
157	7.51858823529412	0.169487706085895	0.168784775086505
158	7.56678431372549	0.172812881169384	0.170383391003460
159	7.61498039215686	0.176169419755599	0.171983852364475
160	7.66317647058824	0.179557358029717	0.173586159169550
161	7.71137254901961	0.182976732176915	0.175190311418685
162	7.75956862745098	0.186427578382372	0.176796309111880
163	7.80776470588235	0.189909932831264	0.178404152249135
164	7.85596078431373	0.193423831708770	0.180013840830450
165	7.90415686274510	0.196969311200066	0.181625374855825
166	7.95235294117647	0.200546407490332	0.183238754325260
167	8.00054901960784	0.204155156764744	0.184853979238754
168	8.04874509803921	0.207795595208479	0.186471049596309
169	8.09694117647059	0.211467759006717	0.188089965397924
170	8.14513725490196	0.215171684344634	0.189710726643599
171	8.19333333333333	0.218907407407407	0.191333333333333
172	8.24152941176471	0.222674964380216	0.192957785467128
173	8.28972549019608	0.226474391448236	0.194584083044983
174	8.33792156862745	0.230305724796647	0.196212226066897
175	8.38611764705882	0.234169000610625	0.197842214532872
176	8.43431372549020	0.238064255075348	0.199474048442907
177	8.48250980392157	0.241991524375994	0.201107727797001
178	8.53070588235294	0.245950844697741	0.202743252595156
179	8.57890196078431	0.249942252225765	0.204380622837370
180	8.62709803921569	0.253965783145246	0.206019838523645

Tabla B.6: Valores identificación 6

Muestra	U(Voltaje)	i(Corriente)	idot(Derivada de laCorriente)
181	8.67529411764706	0.258021473641360	0.207660899653979
182	8.72349019607843	0.262109359899285	0.209303806228374
183	8.77168627450980	0.266229478104198	0.210948558246828
184	8.81988235294118	0.270381864441278	0.212595155709343
185	8.86807843137255	0.274566555095702	0.214243598615917
186	8.91627450980392	0.278783586252648	0.215893886966551
187	8.96447058823529	0.283032994097293	0.217546020761246
188	9.01266666666667	0.287314814814815	0.219200000000000
189	9.06086274509804	0.291629084590391	0.220855824682814
190	9.10905882352941	0.295975839609200	0.222513494809689
191	9.15725490196078	0.300355116056419	0.224173010380623
192	9.20545098039216	0.304766950117225	0.225834371395617
193	9.25364705882353	0.309211377976796	0.227497577854671
194	9.30184313725490	0.313688435820310	0.229162629757785
195	9.35003921568627	0.318198159832945	0.230829527104960
196	9.39823529411765	0.322740586199878	0.232498269896194
197	9.44643137254902	0.327315751106286	0.234168858131488
198	9.49462745098039	0.331923690737348	0.235841291810842
199	9.54282352941176	0.336564441278241	0.237515570934256
200	9.59101960784314	0.341238038914143	0.239191695501730
201	9.63921568627451	0.345944519830231	0.240869665513264
202	9.68741176470588	0.350683920211683	0.242549480968858
203	9.73560784313725	0.355456276243677	0.244231141868512
204	9.78380392156863	0.360261624111390	0.245914648212226
205	9.83200000000000	0.365100000000000	0.247600000000000
206	9.88019607843137	0.369971440094685	0.249287197231834
207	9.92839215686274	0.374875980580621	0.250976239907728
208	9.97658823529412	0.379813657642988	0.252667128027682
209	10.0247843137255	0.384784507466962	0.254359861591696
210	10.0729803921569	0.389788566237721	0.256054440599769

Tabla B.7: Valores identificación 7

Muestra	U(Voltaje)	i(Corriente)	idot(Derivada de laCorriente)
211	10.1211764705882	0.394825870140444	0.257750865051903
212	10.1693725490196	0.399896455360306	0.259449134948097
213	10.2175686274510	0.405000358082487	0.261149250288351
214	10.2657647058824	0.410137614492164	0.262851211072664
215	10.3139607843137	0.415308260774514	0.264555017301038
216	10.3621568627451	0.420512333114715	0.266260668973472
217	10.4103529411765	0.425749867697944	0.267968166089965
218	10.4585490196078	0.431020900709380	0.269677508650519
219	10.5067450980392	0.436325468334200	0.271388696655133
220	10.5549411764706	0.441663606757582	0.273101730103806
221	10.6031372549020	0.447035352164703	0.274816608996540
222	10.6513333333333	0.452440740740741	0.276533333333333
223	10.6995294117647	0.457879808670873	0.278251903114187
224	10.7477254901961	0.463352592140278	0.279972318339100
225	10.7959215686275	0.468859127334133	0.281694579008074
226	10.8441176470588	0.474399450437615	0.283418685121107
227	10.8923137254902	0.479973597635902	0.285144636678201
228	10.9405098039216	0.485581605114172	0.286872433679354
229	10.9887058823529	0.491223509057602	0.288602076124568
230	11.0369019607843	0.496899345651371	0.290333564013841
231	11.0850980392157	0.502609151080655	0.292066897347174
232	11.1332941176471	0.508352961530633	0.293802076124567
233	11.1814901960784	0.514130813186482	0.295539100346021
234	11.2296862745098	0.519942742233379	0.297277970011534
235	11.2778823529412	0.525788784856503	0.299018685121107
236	11.3260784313725	0.531668977241031	0.300761245674740
237	11.3742745098039	0.537583355572140	0.302505651672434
238	11.4224705882353	0.543531956035009	0.304251903114187
239	11.4706666666667	0.549514814814815	0.306000000000000
240	11.5188627450980	0.555531968096735	0.307749942329873

Tabla B.8: Valores identificación 8

Muestra	U(Voltaje)	i(Corriente)	idot(Derivada de laCorriente)
241	11.5670588235294	0.561583452065947	0.309501730103806
242	11.6152549019608	0.567669302907630	0.311255363321799
243	11.6634509803922	0.573789556806960	0.313010841983852
244	11.7116470588235	0.579944249949114	0.314768166089965
245	11.7598431372549	0.586133418519272	0.316527335640138
246	11.8080392156863	0.592357098702610	0.318288350634371
247	11.8562352941176	0.598615326684307	0.320051211072664
248	11.9044313725490	0.604908138649539	0.321815916955017
249	11.9526274509804	0.611235570783484	0.323582468281430
250	12.0008235294118	0.617597659271321	0.325350865051903
251	12.0490196078431	0.623994440298226	0.327121107266436
252	12.0972156862745	0.630425950049378	0.328893194925029
253	12.1454117647059	0.636892224709953	0.330667128027682
254	12.1936078431373	0.643393300465130	0.332442906574395
255	12.2418039215686	0.649929213500087	0.334220530565167
256	12.2900000000000	0.656500000000000	0.336000000000000

Tabla B.9: Valores identificación 9

B.2. Vectores de identificación del sistema con filtro de Kalman

Muestra	U(Voltaje)	i(Corriente)	idot(Derivada de laCorriente)
1	0	0	0
2	0.0481960784313725	0.0135231449442522	-0.0694392156862745
3	0.0963921568627451	0.0121768935024990	-0.0678784313725490
4	0.144588235294118	0.0108612456747405	-0.0663176470588235
5	0.192784313725490	0.00957620146097655	-0.0647568627450980
6	0.240980392156863	0.00832176086120723	-0.0631960784313725
7	0.289176470588235	0.00709792387543253	-0.0616352941176471
8	0.337372549019608	0.00590469050365244	-0.0600745098039216
9	0.385568627450980	0.00474206074586698	-0.0585137254901961
10	0.433764705882353	0.00361003460207613	-0.0569529411764706
11	0.481960784313725	0.00250861207227989	-0.0553921568627451
12	0.530156862745098	0.00143779315647828	-0.0538313725490196
13	0.578352941176471	0.000397577854671283	-0.0522705882352941
14	0.626549019607843	-0.000612033833141096	-0.0507098039215686
15	0.674745098039216	-0.00159104190695886	-0.0491490196078431
16	0.722941176470588	-0.00253944636678201	-0.0475882352941176
17	0.771137254901961	-0.00345724721261053	-0.0460274509803922
18	0.819333333333333	-0.00434444444444444	-0.0444666666666667
19	0.867529411764706	-0.00520103806228374	-0.0429058823529412
20	0.915725490196078	-0.00602702806612841	-0.0413450980392157
21	0.963921568627451	-0.00682241445597847	-0.0397843137254902
22	1.01211764705882	-0.00758719723183391	-0.0382235294117647
23	1.06031372549020	-0.00832137639369473	-0.0366627450980392
24	1.10850980392157	-0.00902495194156093	-0.0351019607843137
25	1.15670588235294	-0.00969792387543252	-0.0335411764705882
26	1.20490196078431	-0.0103402921953095	-0.0319803921568627
27	1.25309803921569	-0.0109520569011918	-0.0304196078431373
28	1.30129411764706	-0.0115332179930796	-0.0288588235294118
29	1.34949019607843	-0.0120837754709727	-0.0272980392156863
30	1.39768627450980	-0.0126037293348712	-0.0257372549019608

Muestra	U(Voltaje)	i(Corriente)	idot(Derivada de laCorriente)
31	1.44588235294118	-0.0130930795847751	-0.0241764705882353
32	1.49407843137255	-0.0135518262206843	-0.0226156862745098
33	1.54227450980392	-0.0139799692425990	-0.0210549019607843
34	1.59047058823529	-0.0143775086505190	-0.0194941176470588
35	1.63866666666667	-0.0147444444444444	-0.0179333333333333
36	1.68686274509804	-0.0150807766243752	-0.0163725490196078
37	1.73505882352941	-0.0153865051903114	-0.0148117647058824
38	1.78325490196078	-0.0156616301422530	-0.0132509803921569
39	1.83145098039216	-0.0159061514801999	-0.0116901960784314
40	1.87964705882353	-0.0161200692041522	-0.0101294117647059
41	1.92784313725490	-0.0163033833141100	-0.00856862745098038
42	1.97603921568627	-0.0164560938100730	-0.00700784313725489
43	2.02423529411765	-0.0165782006920415	-0.00544705882352940
44	2.07243137254902	-0.0166697039600154	-0.00388627450980392
45	2.12062745098039	-0.0167306036139946	-0.00232549019607842
46	2.16882352941176	-0.0167608996539792	-0.000764705882352931
47	2.21701960784314	-0.0167605920799692	0.000796078431372554
48	2.26521568627451	-0.0167296808919646	0.00235686274509804
49	2.31341176470588	-0.0166681660899654	0.00391764705882354
50	2.36160784313726	-0.0165760476739715	0.00547843137254903
51	2.40980392156863	-0.0164533256439831	0.00703921568627451
52	2.45800000000000	-0.0163000000000000	0.00860000000000001
53	2.50619607843137	-0.0161160707420223	0.0101607843137255
54	2.55439215686275	-0.0159015378700500	0.0117215686274510
55	2.60258823529412	-0.0156564013840830	0.0132823529411765
56	2.65078431372549	-0.0153806612841215	0.0148431372549020
57	2.69898039215686	-0.0150743175701653	0.0164039215686275
58	2.74717647058824	-0.0147373702422145	0.0179647058823530
59	2.79537254901961	-0.0143698193002691	0.0195254901960784
60	2.84356862745098	-0.0139716647443291	0.0210862745098039

Tabla B.11: Valores identificación KF 2

Muestra	U(Voltaje)	i(Corriente)	idot(Derivada de laCorriente)
61	2.89176470588235	-0.0135429065743945	0.0226470588235294
62	2.93996078431373	-0.0130835447904652	0.0242078431372549
63	2.98815686274510	-0.0125935793925413	0.0257686274509804
64	3.03635294117647	-0.0120730103806228	0.0273294117647059
65	3.08454901960784	-0.0115218377547097	0.0288901960784314
66	3.13274509803922	-0.0109400615148020	0.0304509803921569
67	3.18094117647059	-0.0103276816608996	0.0320117647058824
68	3.22913725490196	-0.00968469819300269	0.0335725490196079
69	3.27733333333333	-0.00901111111111109	0.0351333333333333
70	3.32552941176471	-0.00830692041522490	0.0366941176470588
71	3.37372549019608	-0.00757212610534409	0.0382549019607843
72	3.42192156862745	-0.00680672818146865	0.0398156862745098
73	3.47011764705882	-0.00601072664359861	0.0413764705882353
74	3.51831372549020	-0.00518412149173394	0.0429372549019608
75	3.56650980392157	-0.00432691272587465	0.0444980392156863
76	3.61470588235294	-0.00343910034602075	0.0460588235294118
77	3.66290196078431	-0.00252068435217223	0.0476196078431373
78	3.71109803921569	-0.00157166474432910	0.0491803921568628
79	3.75929411764706	-0.000592041522491326	0.0507411764705882
80	3.80749019607843	0.000418185313341039	0.0523019607843138
81	3.85568627450980	0.00145901576316803	0.0538627450980392
82	3.90388235294118	0.00253044982698964	0.0554235294117647
83	3.95207843137255	0.00363248750480586	0.0569843137254902
84	4.00027450980392	0.00476512879661670	0.0585450980392157
85	4.04847058823529	0.00592837370242217	0.0601058823529412
86	4.09666666666667	0.00712222222222224	0.0616666666666667
87	4.14486274509804	0.00834667435601693	0.0632274509803922
88	4.19305882352941	0.00960173010380625	0.0647882352941177
89	4.24125490196078	0.0108873894655902	0.0663490196078432
90	4.28945098039216	0.0122036524413687	0.0679098039215686

Tabla B.12: Valores identificación KF 3

Muestra	U(Voltaje)	i(Corriente)	idot(Derivada de laCorriente)
91	4.33764705882353	0.0135505190311419	0.0694705882352941
92	4.38584313725490	0.0149279892349097	0.0710313725490196
93	4.43403921568627	0.0163360630526721	0.0725921568627451
94	4.48223529411765	0.0177747404844291	0.0741529411764706
95	4.53043137254902	0.0192440215301807	0.0757137254901961
96	4.57862745098039	0.0207439061899270	0.0772745098039216
97	4.62682352941176	0.0222743944636679	0.0788352941176471
98	4.67501960784314	0.0238354863514033	0.0803960784313725
99	4.72321568627451	0.0254271818531334	0.0819568627450980
100	4.77141176470588	0.0270494809688581	0.0835176470588235
101	4.81960784313726	0.0287023836985775	0.0850784313725490
102	4.86780392156863	0.0303858900422915	0.0866392156862745
103	4.91600000000000	0.0321000000000000	0.0882000000000000
104	4.96419607843137	0.0338447135717032	0.0897607843137255
105	5.01239215686274	0.0356200307574010	0.0913215686274510
106	5.06058823529412	0.0374259515570934	0.0928823529411765
107	5.10878431372549	0.0392624759707805	0.0944431372549020
108	5.15698039215686	0.0411296039984622	0.0960039215686275
109	5.20517647058824	0.0430273356401384	0.0975647058823530
110	5.25337254901961	0.0449556708958093	0.0991254901960785
111	5.30156862745098	0.0469146097654748	0.100686274509804
112	5.34976470588235	0.0489041522491350	0.102247058823529
113	5.39796078431373	0.0509242983467897	0.103807843137255
114	5.44615686274510	0.0529750480584391	0.105368627450980
115	5.49435294117647	0.0550564013840831	0.106929411764706
116	5.54254901960784	0.0571683583237217	0.108490196078431
117	5.59074509803922	0.0593109188773549	0.110050980392157
118	5.63894117647059	0.0614840830449827	0.111611764705882
119	5.68713725490196	0.0636878508266052	0.113172549019608
120	5.73533333333333	0.0659222222222223	0.114733333333333

Tabla B.13: Valores identificación KF 4

Muestra	U(Voltaje)	i(Corriente)	idot(Derivada de laCorriente)
121	5.78352941176471	0.0681871972318340	0.116294117647059
122	5.83172549019608	0.0704827758554402	0.117854901960784
123	5.87992156862745	0.0728089580930412	0.119415686274510
124	5.92811764705882	0.0751657439446367	0.120976470588235
125	5.97631372549020	0.0775531334102268	0.122537254901961
126	6.02450980392157	0.0799711264898117	0.124098039215686
127	6.07270588235294	0.0824197231833910	0.125658823529412
128	6.12090196078431	0.0848989234909651	0.127219607843137
129	6.16909803921569	0.0874087274125337	0.128780392156863
130	6.21729411764706	0.0899491349480969	0.130341176470588
131	6.26549019607843	0.0925201460976548	0.131901960784314
132	6.31368627450980	0.0951217608612072	0.133462745098039
133	6.36188235294118	0.0977539792387544	0.135023529411765
134	6.41007843137255	0.100416801230296	0.136584313725490
135	6.45827450980392	0.103110226835832	0.138145098039216
136	6.50647058823529	0.105834256055363	0.139705882352941
137	6.55466666666667	0.108588888888889	0.141266666666667
138	6.60286274509804	0.111374125336409	0.142827450980392
139	6.65105882352941	0.114189965397924	0.144388235294118
140	6.69925490196078	0.117036409073433	0.145949019607843
141	6.74745098039216	0.119913456362937	0.147509803921569
142	6.79564705882353	0.122821107266436	0.149070588235294
143	6.84384313725490	0.125759361783929	0.150631372549020
144	6.89203921568627	0.128728219915417	0.152192156862745
145	6.94023529411765	0.131727681660900	0.153752941176471
146	6.98843137254902	0.134757747020377	0.155313725490196
147	7.03662745098039	0.137818415993849	0.156874509803922
148	7.08482352941176	0.140909688581315	0.158435294117647
149	7.13301960784314	0.144031564782776	0.159996078431373
150	7.18121568627451	0.147184044598231	0.161556862745098

Tabla B.14: Valores identificación KF 5

Muestra	U(Voltaje)	i(Corriente)	idot(Derivada de laCorriente)
151	7.22941176470588	0.150367128027682	0.163117647058824
152	7.27760784313725	0.153580815071127	0.164678431372549
153	7.32580392156863	0.156825105728566	0.166239215686275
154	7.37400000000000	0.160100000000000	0.167800000000000
155	7.42219607843137	0.163405497885429	0.169360784313725
156	7.47039215686274	0.166741599384852	0.170921568627451
157	7.51858823529412	0.170108304498270	0.172482352941176
158	7.56678431372549	0.173505613225682	0.174043137254902
159	7.61498039215686	0.176933525567090	0.175603921568627
160	7.66317647058824	0.180392041522491	0.177164705882353
161	7.71137254901961	0.183881161091888	0.178725490196078
162	7.75956862745098	0.187400884275279	0.180286274509804
163	7.80776470588235	0.190951211072664	0.181847058823529
164	7.85596078431373	0.194532141484045	0.183407843137255
165	7.90415686274510	0.198143675509420	0.184968627450980
166	7.95235294117647	0.201785813148789	0.186529411764706
167	8.00054901960784	0.205458554402153	0.188090196078431
168	8.04874509803921	0.209161899269512	0.189650980392157
169	8.09694117647059	0.212895847750865	0.191211764705882
170	8.14513725490196	0.216660399846213	0.192772549019608
171	8.19333333333333	0.220455555555556	0.194333333333333
172	8.24152941176471	0.224281314878893	0.195894117647059
173	8.28972549019608	0.228137677816225	0.197454901960784
174	8.33792156862745	0.232024644367551	0.199015686274510
175	8.38611764705882	0.235942214532872	0.200576470588235
176	8.43431372549020	0.239890388312188	0.202137254901961
177	8.48250980392157	0.243869165705498	0.203698039215686
178	8.53070588235294	0.247878546712803	0.205258823529412
179	8.57890196078431	0.251918531334102	0.206819607843137
180	8.62709803921569	0.255989119569396	0.208380392156863

Tabla B.15: Valores identificación KF 6

Muestra	U(Voltaje)	i(Corriente)	idot(Derivada de laCorriente)
181	8.67529411764706	0.260090311418685	0.209941176470588
182	8.72349019607843	0.264222106881969	0.211501960784314
183	8.77168627450980	0.268384505959246	0.213062745098039
184	8.81988235294118	0.272577508650519	0.214623529411765
185	8.86807843137255	0.276801114955786	0.216184313725490
186	8.91627450980392	0.281055324875048	0.217745098039216
187	8.96447058823529	0.285340138408305	0.219305882352941
188	9.01266666666667	0.289655555555556	0.220866666666667
189	9.06086274509804	0.294001576316801	0.222427450980392
190	9.10905882352941	0.298378200692042	0.223988235294118
191	9.15725490196078	0.302785428681277	0.225549019607843
192	9.20545098039216	0.307223260284506	0.227109803921569
193	9.25364705882353	0.311691695501730	0.228670588235294
194	9.30184313725490	0.316190734332949	0.230231372549020
195	9.35003921568627	0.320720376778162	0.231792156862745
196	9.39823529411765	0.325280622837370	0.233352941176471
197	9.44643137254902	0.329871472510573	0.234913725490196
198	9.49462745098039	0.334492925797770	0.236474509803922
199	9.54282352941176	0.339144982698962	0.238035294117647
200	9.59101960784314	0.343827643214148	0.239596078431373
201	9.63921568627451	0.348540907343330	0.241156862745098
202	9.68741176470588	0.353284775086505	0.242717647058824
203	9.73560784313725	0.358059246443676	0.244278431372549
204	9.78380392156863	0.362864321414841	0.245839215686275
205	9.83200000000000	0.367700000000000	0.247400000000000
206	9.88019607843137	0.372566282199154	0.248960784313725
207	9.92839215686274	0.377463168012303	0.250521568627451
208	9.97658823529412	0.382390657439446	0.252082352941177
209	10.0247843137255	0.387348750480584	0.253643137254902
210	10.0729803921569	0.392337447135717	0.255203921568627

Tabla B.16: Valores identificación KF 7

Muestra	U(Voltaje)	i(Corriente)	idot(Derivada de laCorriente)
211	10.1211764705882	0.397356747404844	0.256764705882353
212	10.1693725490196	0.402406651287966	0.258325490196078
213	10.2175686274510	0.407487158785083	0.259886274509804
214	10.2657647058824	0.412598269896194	0.261447058823529
215	10.3139607843137	0.417739984621300	0.263007843137255
216	10.3621568627451	0.422912302960400	0.264568627450980
217	10.4103529411765	0.428115224913495	0.266129411764706
218	10.4585490196078	0.433348750480584	0.267690196078431
219	10.5067450980392	0.438612879661669	0.269250980392157
220	10.5549411764706	0.443907612456747	0.270811764705882
221	10.6031372549020	0.449232948865821	0.272372549019608
222	10.6513333333333	0.454588888888889	0.273933333333333
223	10.6995294117647	0.459975432525952	0.275494117647059
224	10.7477254901961	0.465392579777009	0.277054901960784
225	10.7959215686275	0.470840330642061	0.278615686274510
226	10.8441176470588	0.476318685121108	0.280176470588235
227	10.8923137254902	0.481827643214149	0.281737254901961
228	10.9405098039216	0.487367204921184	0.283298039215686
229	10.9887058823529	0.492937370242215	0.284858823529412
230	11.0369019607843	0.498538139177240	0.286419607843137
231	11.0850980392157	0.504169511726259	0.287980392156863
232	11.1332941176471	0.509831487889273	0.289541176470588
233	11.1814901960784	0.515524067666282	0.291101960784314
234	11.2296862745098	0.521247251057286	0.292662745098039
235	11.2778823529412	0.527001038062284	0.294223529411765
236	11.3260784313725	0.532785428681276	0.295784313725490
237	11.3742745098039	0.538600422914264	0.297345098039216
238	11.4224705882353	0.544446020761246	0.298905882352941
239	11.4706666666667	0.550322222222223	0.300466666666667
240	11.5188627450980	0.556229027297194	0.302027450980392

Tabla B.17: Valores identificación KF 8

Muestra	U(Voltaje)	i(Corriente)	idot(Derivada de laCorriente)
241	11.5670588235294	0.562166435986159	0.303588235294118
242	11.6152549019608	0.568134448289120	0.305149019607843
243	11.6634509803922	0.574133064206075	0.306709803921569
244	11.7116470588235	0.580162283737024	0.308270588235294
245	11.7598431372549	0.586222106881969	0.309831372549020
246	11.8080392156863	0.592312533640907	0.311392156862745
247	11.8562352941176	0.598433564013841	0.312952941176471
248	11.9044313725490	0.604585198000769	0.314513725490196
249	11.9526274509804	0.610767435601692	0.316074509803922
250	12.0008235294118	0.616980276816609	0.317635294117647
251	12.0490196078431	0.623223721645521	0.319196078431373
252	12.0972156862745	0.629497770088428	0.320756862745098
253	12.1454117647059	0.635802422145329	0.322317647058824
254	12.1936078431373	0.642137677816225	0.323878431372549
255	12.2418039215686	0.648503537101115	0.325439215686275
256	12.2900000000000	0.654900000000000	0.327000000000000

Tabla B.18: Valores identificación KF 9

Referencias

- [1] Abusleme A. and Cipriano A. Experiencia de levitacion magnetica de bajo costo para laboratorio de control automatico. Technical report, Pontificia Universidad Catolica de Chile, 2001.
- [2] B. Bhushan. Introduction to tribology . New York, Wiley, 2002
- [3] D.L. Trumper, S.M. Olson, and P.K. Subrahmanyam. Linearizing control of magnetic-suspension systems. IEEE Trans. Contr. Syst. Technol. , 5(4):427–438, 1997
- [4] M. Dussaux. The industrial applications of the active magnetic bearings technology. Proc. 2nd Int. Symp. Magn. Bearings , pages 33–38, 1990.
- [5] C. Samiappan, N. Mirnategui, B.E. Paden, and J.F. Antaki. Maglev apparatus for power minimization and control of artificial hearts. IEEE Trans. Contr. Syst. Technol. ,16(1):13–18, 2008
- [6] J. F. Whidborne. Ems control system design for a maglev vehicle: A critical system. Automatica , 29(5):1345–1349, 1993.
- [7] Peijnenburg A., Vermeulen J., and Eijk J., van. Magnetic levitation systems compared to conventional bearing systems. Microelectronic Engineering ELSEVIER, 83:1372–1375, 2006.
- [8] Weilert M. A., Whitaker D. L., Maris H. J., and Seidel G. M. Magnetic levitation of helium liquid. Low temperature physics, 106:50–58, 1997.

-
- [9] Sun L., Zhao K., and Kou B. An electromagnetic launcher with magnetic levitation realized based on vector control. *IEEE transactions on magnetics*, 45: 467–470, 2009.
 - [10] Samiappan C., Mirnateghi N., and Paden B. Maglev apparatus for power minimization and control of artificial hearts. *IEEE transaction on control system technology*, 16:13–18, 2008.
 - [11] Li-mei W., Fu-ying J., and Chun-fang L. Sliding mode control of cnc electromagnetic levitation system based on feedback linearization. In *Chinese control and decision conference*, 2009
 - [12] Berkelman P. and Dzadovsky M. Extending the motion ranges of magnetic levitation for haptic interaction. In *Third joint eurohaptics conference and symposium on haptic interfaces*, 2009.
 - [13] SHIAO Ying-shing. Design and implementation of a controller for a magnetic levitation system. *Proc. Natl. Sci. Conc. ROC(D)*, 11:88–94, 2001.
 - [14] Lilienkamnp katie and Lundberg K. Low-cost magnetic levitation project kits for teaching feedback system design. In *American control conference*, 2004.
 - [15] G. Michel A., Lempel and Carbajales R. Levitador magnetico. Technical report, Universidad de buenos aires, 2004.
 - [16] QIU H., CAO G., Pan J., and LIN L. The development of magnetic levitation ball control system based on tms320f2812. In *3rd International conference on power electronics systems and applications*, 2009
 - [17] Huachun W. and Yefa H. Study on fuzzy control algorithm for magnetic levitated platform. In *International conference on measuring technology and mechatronics automation*, 2009.
 - [18] Shafayet Hossain. Design of a robust controller for a magnetic levitation system. Wichita State University, 2007.

-
- [19] G. Michel A., Lempel and Carbajales R. Levitador magnetico. Technical report, Universidad de buenos aires, 2004.
 - [20] Guillermo E., Larriva J., and Trelles J. Control de un levitador magnetico. Technical report, Ecuador, 2003
 - [21] Espinosa U., Aceves A., Vega A., and Carbajal C. Sistema de levitacion magnetica controlado con logica difusa. Technical report, ITESM Campus Estado de Mexico, 2003.
 - [22] Panuncio Cruz Francisco. Control de un sistema de levitacion magnetica con compensacion de redes neuronales. Master's thesis, Centro de investigacion y de estudios avanzados del instituto politecnico nacional, 2009.
 - [23] Hurley W., G. Electromagnetic design of a magnetic suspension system. IEEE Transactions on education, 40:30–35, 1997.
 - [24] Palmieri D. and M. Pucci. Contruccion de un sistema de levitacion magnetica para laboratorio controlado por pc en tiempo real. Technical report, Universidad Nacional de Quilmes, 2004.
 - [25] Peijnenburg A., Vermeulen J., and Eijk J., van. Magnetic levitation systems compared to conventional bearing systems. Microelectronic Engineering ELSEVIER, 83:1372–1375, 2006.
 - [26] Clark T., Kondoleon A., Thompson M., and Thornton R. Shaftless, magnetically levitated flywheel energy storage system. Technical report, Aerospace flywheel workshop, 1998.
 - [27] Weilert M. A., Whitaker D. L., Maris H. J., and Seidel G. M. Magnetic levitation of helium liquid. Low temperature physics, 106:50–58, 1997.
 - [28] Khamesee B., Kato N., and Nakamura T. Performance improvement of a magnetically levitated microrobot using an adaptive control. Proceeding of the international conference on MEMS, NANO and Smart systems, 23:46–52, 2003.

-
- [29] Kang D. and Chung W. Integrated monitoring scheme for a maglev guideway using multiplexed fbg sensor arrays. *NDTE International ELSEVIER*, 42: 260–266, 2009.
- [30] Perren G. Estudio de las aplicaciones practicas de la levitación magnetica. Technical report, n,n, 2003. 2003, Tech. Report.
- [31] E. Bejar and A. Moran, “Deep reinforcement learning based neuro-control for a two-dimensional magnetic positioning system,” *Proceedings of the 4th International Conference on Control, Automation and Robotics (ICCAR)*, pp. 268–273, April 2018.
- [32] S. Xiao, Y. Li, and J. Liu, “A model reference adaptive pid control for electromagnetic actuated micro-positioning state,” *Proceedings of the 8th IEEE International Conference on Automation Science and Engineering*, August 2012.
- [33] Y. Li and Q. Xu, “Design and analysis of a totally decoupled flexure-based xy parallel micromanipulator,” *IEEE Transactions on Robotics*, vol. 25, no. 3, pp. 645–657, June 2009.
- [34] Hajjaji, A.E. y Guladsine,M. Modeling and non linear control of magnetic levitation system. *IEEE transaction on industrial electronic* Vo.48 No.4 2001
- [35] Poznyak A. MODELADO MATEMÁTICO de los Sistemas Mecánicos, Electricos y Electromecánicos
- [36] Isaac g. walter s. Sistema de levitación magnética 2014
- [37] Dorf, R. C. y Bishop, R. H. (2005). *Sistemas de Control Moderno* (10a ed.), Madrid: Pearson Educación, p. 43-45.
- [38] Li-mei W., Fu-ying J., and Chun-fang L. Sliding mode control of cnc electromagnetic levitation system based on feedback linearization. In *Chinese control and decision conference*, 2009