



BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE PUEBLA

FACULTAD DE CIENCIAS DE LA ELECTRÓNICA

**MAESTRÍA EN CIENCIAS DE LA ELECTRÓNICA,
OPCIÓN EN AUTOMATIZACIÓN**

**ACTUALIZACIÓN DEL MODELADO, CONTROL Y
SISTEMA ELECTRÓNICO DE UN PÉNDULO
INVERTIDO SOBRE BASE MÓVIL**

T E S I S

**PRESENTADA PARA OBTENER EL TÍTULO DE:
MAESTRO EN CIENCIAS DE LA ELECTRONICA
OPCION EN AUTOMATIZACION**

PRESENTA

ING. LUIS FRANCISCO DE LA ROSA PINEDA

DIRECTORES DE TESIS

DR. SERGIO VERGARA LIMON (FCE-BUAP)

DR. DANIEL MARCELO GONZALEZ ARRIAGA (FCC-BUAP)

DIRECTOR EXTERNO

DR. JESUS LOPEZ GOMEZ (UJAT)

Puebla, Puebla, Agosto 2025.

Resumen

Este trabajo presenta el desarrollo del modelado, actualización de la electrónica y control de un péndulo invertido sobre base móvil, abordando los desafíos de integración interdisciplinaria entre mecánica, electrónica y software. La investigación parte de la necesidad de diseñar sistemas autónomos robustos, precisos y versátiles, aplicables en contextos industriales, educativos y de investigación. El diseño del robot se fundamenta en modelos dinámicos y cinemáticos que describen su comportamiento, considerando factores como la fricción y restricciones dinámicas. Estos modelos matemáticos no solo permiten predecir su movimiento, sino que también constituyen una base sólida para el diseño del sistema de control y optimización de su desempeño.

En el ámbito mecánico, se actualiza la estructura diseñada previamente, con la intención de asegurarnos que el mecanismo se encuentre en óptimas condiciones y garantizar su funcionamiento. Esto se complementa con un sistema electrónico que incluye módulos de potencia, como fuentes de alimentación y puentes H, y dispositivos de adquisición, como encoders y fines de carrera. Estos elementos garantizan un monitoreo continuo y detallado de las variables operativas del robot, asegurando su capacidad de respuesta en tiempo real.

El sistema de control embebido implementado en una FPGA DE0-CV, destaca por su capacidad para gestionar el lazo de control en tiempo real, integrando módulos de comunicación inalámbrica, generación de señales PWM, algoritmos de filtrado digital y compensación de errores. Estos elementos permiten una operación estable y eficiente del robot. El software complementa mediante una interfaz de usuario intuitiva y un lenguaje de programación específico, que facilita la interacción con el sistema y su configuración para distintas aplicaciones.

La validación experimental confirma la precisión del modelo dinámico obtenido y el acierto de las actualizaciones realizadas al robot, obteniendo un sistema funcional que cumple con los objetivos propuestos, estableciendo un marco metodológico replicable para futuros trabajos.

Agradecimientos

Agradezco a Dios por la vida que me ha dado, por todas las oportunidades que me ha brindado para seguir superándome y darme la fortaleza necesaria para culminar una meta más en mi vida.

A mi familia, el pilar más importante en mi vida. A mi padre, Agustín de la Rosa Gómez, gracias por tu ejemplo inquebrantable, por enseñarme a avanzar con humildad y esfuerzo, y por estar siempre ahí, sosteniéndome en silencio. A mi hermano, Enoc Agustín de la Rosa Pineda, gracias por ser mi compañero de vida, por tu apoyo incondicional, y por creer en mí incluso cuando yo mismo dudaba. Este logro también es de ustedes, porque en cada momento difícil encontré en ustedes la razón para seguir adelante. Los amo profundamente.

Agradezco profundamente al Dr. Sergio Vergara Limón, al Dr. Daniel Marcelo González Arriaga y al Dr. Jesús López Gómez por su acompañamiento, confianza y disposición a lo largo de este proceso. Su guía no solo fue clave para la realización de esta tesis, sino también una fuente de aprendizaje y motivación personal.

Agradezco a mis amigos por su apoyo constante, su compañía y su ánimo en los momentos difíciles durante este proceso. Su amistad ha sido un refugio y una fuente de motivación que me impulsó a seguir adelante. Gracias por estar siempre presentes y por compartir conmigo esta etapa tan importante de mi vida.

Agradezco sinceramente a todos los maestros que a lo largo de mi formación académica compartieron sus conocimientos, experiencia y dedicación. Su enseñanza y compromiso fueron fundamentales para construir las bases que me permitieron afrontar los retos de esta tesis y avanzar en mi desarrollo profesional y personal.

Agradezco sinceramente al Dr. José Eligio Moisés Gutiérrez, a la Dra. Olga Guadalupe Félix Beltrán y al Dr. Carlos Sánchez López por formar parte del jurado de esta tesis. Su tiempo, disposición y valiosas observaciones contribuyeron de manera significativa a enriquecer este trabajo y a mi formación académica.

Agradezco a la Benemérita Universidad Autónoma de Puebla (BUAP), a la Facultad de Ciencias de la Electrónica y a la Maestría en Ciencias de la Electrónica, Opción en Automatización por brindarme una formación integral, un entorno académico de calidad y las herramientas necesarias para desarrollar este trabajo de tesis.

Asimismo, expreso mi sincero agradecimiento a la Secretaría de Ciencia, Humanidades, Tecnología e Innovación (SECIHTI) por el apoyo otorgado a través de la beca que hizo posible la continuidad de mis estudios y el cumplimiento de esta etapa profesional.

Dedicatoria

*A la memoria de mi madre, Rosa Mónica Pineda Espinoza,
mi amor eterno y la raíz más profunda de quien soy.
Tu partida dejó un silencio imposible de llenar,
pero también una luz que aún me guía con fuerza en la oscuridad.
No hay logro en mi vida que no lleve tu nombre,
porque cada paso que he dado, cada obstáculo superado,
ha sido sostenido por todo lo que sembraste en mí.
Fuiste ejemplo de amor, valentía y entrega,
ser tu hijo es, y será siempre, mi mayor orgullo.
Esta tesis es para ti, mamá,
porque todo lo que soy, te lo debo a ti.*

Índice general

Resumen	I
Agradecimientos	II
Dedicatoria	III
Introducción	x
1. Descripción del robot	17
1.1. Diagrama a bloques	17
1.2. Sistema mecánico	18
1.3. Conclusión	24
2. Modelo dinámico	25
2.1. Metodología para la obtención del modelo dinámico	25
2.1.1. Análisis de parámetros agrupados	25
2.1.2. Método de Euler-Lagrange	27
2.1.3. Esfuerzos generalizados	28
2.2. Modelo cinemático	28
2.3. Modelo dinámico	29
2.4. Conclusión	45
3. Electrónica del robot	47
3.1. Electrónica de potencia	47
3.1.1. Caracterización del motor	48
3.1.2. Fuente de alimentación	53
3.1.3. Puente H	53
3.2. Electrónica de adquisición	56
3.2.1. Fines de carrera	56
3.2.2. Encoder	58
3.3. Integración de la electrónica	60
3.4. Conclusiones	62
4. Firmware del microcontrolador	63
4.1. Sistema embebido a implementar como lazo de control	63
4.2. FPGA DE0-CV	66
4.3. Módulo WI-FI	68

4.4. Decoder	69
4.5. Efecto de muestreo	72
4.6. Filtro pasa bajas	75
4.7. Error en la cuantificación de la velocidad en función de la posición . .	75
4.8. Generador de PWM	76
4.9. Conclusión	78
5. Software de usuario	79
5.1. Lenguaje de programación del microprocesador (Lenguaje D)	79
5.2. Compilador de lenguaje de programación	81
5.3. Interfaz de usuario	82
5.4. Conclusión	85
6. Resultados experimentales	87
6.1. Controlador P	88
6.2. Controlador PD	89
6.2.1. Métrica de evaluación para comparar datos	91
6.2.2. Comparación del controlador P	91
6.2.3. Comparación del controlador PD	93
6.3. Conclusión	94
Conclusiones Generales	95
Bibliografía	97
Anexos	101

Índice de figuras

1.	Péndulo simple.	X
2.	Reloj mecánico con péndulo.	X
3.	Péndulo cónico.	XI
4.	Péndulo de Furuta.	XI
5.	Scooter [7].	XII
6.	Hoverboard [8].	XIII
7.	Monociclo Eléctrico con asiento y manubrio [9].	XIII
8.	Monopatín eléctrico tipo patineta [10].	XIII
9.	Monopatín sin manubrio zuimi [11].	XIV
10.	Monopatín con manubrio [12].	XIV
1.1.	Diagrama a bloques del robot.	17
1.2.	Péndulo invertido sobre base móvil del laboratorio del MCEA.	18
1.3.	Carrito del péndulo invertido.	19
1.4.	Vista lateral y superior del carro [15].	19
1.5.	Medidas de las varillas del péndulo [15].	20
1.6.	Base fija.	20
1.7.	Perfil de aluminio utilizado en la base [15].	20
1.8.	Dibujo de la base fija y riel del péndulo invertido [15].	21
1.9.	Soportes de la estructura del péndulo invertido [15].	21
1.10.	Poleas y banda dentada.	22
1.11.	Medidas de las poleas del péndulo invertido [15].	22
1.12.	Base del motor con mecanismo de tornillo sinfín y corona.	23
1.13.	Encoder.	23
1.14.	Finales de carrera.	24
1.15.	Motor DC.	24
2.1.	Diagrama del péndulo invertido.	29
2.2.	Diagrama a bloques del sistema completo.	30
2.3.	Componentes físicos del diagrama del robot (1).	30
2.4.	Componentes físicos del diagrama del robot (2).	31
2.5.	Diagrama de un eje.	39
2.6.	Diagrama de flujo de la simulación.	41
2.7.	Diagrama del sistema de control.	41
2.8.	Gráfica de posición del péndulo con el controlador proporcional.	42
2.9.	Gráfica de velocidad del péndulo con el controlador proporcional.	43
2.10.	Gráfica de posición del péndulo con el controlador PD.	44

2.11. Gráfica de velocidad del péndulo con el controlador proporcional. . .	44
3.1. Diagrama de la electrónica del robot.	47
3.2. Motor DC.	48
3.3. Dinamómetro.	48
3.4. Freno Prony.	48
3.5. Fuente DC variable.	49
3.6. Generador de funciones.	49
3.7. Circuito de conmutación.	49
3.8. Diagrama de la primera prueba.	50
3.9. Diagrama de la segunda prueba.	51
3.10. Gráfica de respuesta del torque entregado a 28kHz.	52
3.11. Fuente conmutada.	53
3.12. Circuito del puente H.	54
3.13. Circuito de protección.	55
3.14. Puente H armado.	55
3.15. Sensor inductivo.	56
3.16. Circuito de acoplamiento de los fines de carrera.	56
3.17. Base de lámina.	57
3.18. Fines de carrera inductivos.	57
3.19. Encoder.	58
3.20. Encoder modelo E50S6-2500-6-l-5 y señales de salida A, B y Z. [16]. .	58
3.21. Encoder modelo DFS60- BHC02000.	59
3.22. Acoplamiento de los encoders.	59
3.23. Lectura del encoder con el microprocesador.	59
3.24. Fuente conmutada de 24V y 5V.	60
3.25. Componentes del puente H.	60
3.26. FPGA, WI-FI, PCB de los encoders.	61
3.27. Conector.	61
3.28. Bases de acrílico.	61
3.29. Vista lateral y superior de la torre.	62
3.30. Torre de electrónica.	62
4.1. Diagrama general del FPGA.	63
4.2. Diagrama a bloques del FPGA para el robot.	66
4.3. Tarjeta de desarrollo DE0-CV [27].	66
4.4. Descripción física de la tarjeta DE0-CV.	67
4.5. Puertos GPIO de la tarjeta DE0-CV [27].	67
4.6. Módulo de comunicación inalámbrica Wifly [20].	68
4.7. Comunicación serial entre dispositivos a través de señal Rx y Tx [20].	68
4.8. Ventana de configuración de módulo Wifi con Teraterm [20].	69
4.9. Encoder incremental [15].	70
4.10. Señal del encoder incremental de dos canales A y B [21].	70
4.11. Diagrama de tiempos, donde se muestra la recuperación de la señal de reloj a partir de las señales A y B de un encoder.	71
4.12. Circuito para recuperación de la señal de reloj.	72

4.13. Diagrama de estados del decoder [21].	72
4.14. Muestreo de una señal $f(t)$	73
4.15. Transformada de Fourier de $f(t)$ y $\text{Comb}(t)$	74
4.16. Resultados de la convolución entre $F(w)$ con el peine de Dirac $\text{Comb}(w)$ [21].	74
4.17. Grafica de Ganancia del Filtro Pasa bajas.	75
4.18. Filtrado de la señal recuperada.	75
4.19. Cuantificación correcta de la velocidad en función de la posición [21].	76
4.20. Diagrama del PWM [21].	77
4.21. Diagrama de tiempos del PWM [21].	77
5.1. Diagrama del proceso de configuración del software.	79
5.2. Compilador de lenguaje de programación de alto nivel a lenguaje máquina [20].	82
5.3. Interfaz de usuario, transferencia de software base [20].	83
5.4. Transferencia de variables a memoria RAM [20].	83
5.5. Transferencia de parámetros de sistema [20].	84
5.6. Pestaña para envío de comandos a microprocesador [20].	85
5.7. Pestaña de software destinada a adquisición de datos [20].	85
6.1. Diagrama del sistema de control.	87
6.2. Gráfica de posición del péndulo con el controlador P.	88
6.3. Gráfica de velocidad del péndulo con el controlador P.	89
6.4. Gráfica de posición del péndulo con el controlador PD.	90
6.5. Gráfica de velocidad del péndulo con el controlador PD.	90
6.6. Comparación de la posición del péndulo experimental y simulada del controlador P.	91
6.7. Comparación de la velocidad del péndulo experimental y simulada del controlador P.	92
6.8. Comparación de la posición del péndulo experimental y simulada del controlador PD.	93
6.9. Comparación de la velocidad del péndulo experimental y simulada del controlador PD.	93

Introducción

Llamamos péndulo a un ente ideal constituido por una masa puntual suspendido de un hilo inextensible, capaz de oscilar libremente y sin rozamiento, es un sistema mecánico que ha sido objeto de estudio desde la antigüedad, sin embargo, no fue hasta el año 1602 cuando Galileo Galilei dio a conocer los resultados de sus estudios acerca de las características del péndulo, tales como el período, la amplitud de la oscilación y las relaciones existentes entre ellas [1].

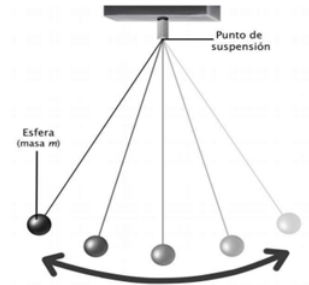


Figura 1: Péndulo simple.

A partir de ese momento otros científicos hicieron uso de estos conocimientos para profundizar en ellos y aplicarlos. Basado sobre las ideas de Galileo, en 1656 Christiaan Huygens patentó un reloj mecánico donde empleó un péndulo para regular el movimiento [2].



Figura 2: Reloj mecánico con péndulo.

También el científico inglés Roberto Hooke ideó el péndulo cónico, consistiendo en un péndulo que está libre hacer pivotar en ambas direcciones. Analizando los movimientos circulares de la sacudida del péndulo, él la utilizó para analizar los movimientos orbitales de los planetas. Foucault, físico francés e inventor del giroscopio, haciendo uso de las propiedades del péndulo pudo demostrar la existencia del movimiento de rotación de la Tierra [3].

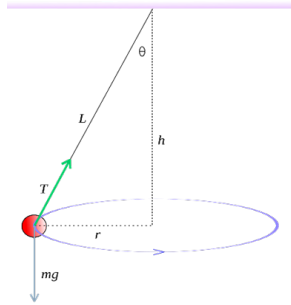


Figura 3: Péndulo cónico.

Aunque el péndulo simple es un sistema que ha sido estudiado durante siglos, la versión invertida, en la que el objetivo es mantener el péndulo en posición vertical en lugar de dejarlo oscilar, es un problema más complejo y moderno. Desde antes de los años 70, se han realizado diversos proyectos con péndulos invertidos, uno de los investigadores en el área más destacados es el profesor Katsuhisa Furuta, quien ha realizado diversos aportes teóricos y experimentales. Creo el péndulo Furuta en el Instituto de Tecnología de Tokio, Japón, que es un sistema subactuado de dos grados de libertad que permite ilustrar diferentes principios de control y que es usado en distintas universidades [4].

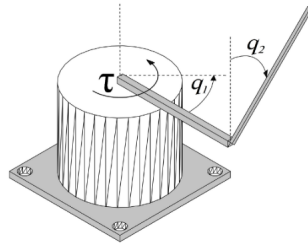


Figura 4: Péndulo de Furuta.

El sistema mecatrónico tipo péndulo invertido está compuesto por una serie de elementos como lo son una estructura mecánica, un sistema sensorial, un sistema de adquisición de datos y un ordenador, además de esto tiene que contar con un software que gestione el área sensorial y el movimiento de la estructura mecánica. El propósito final es el de conservar el péndulo perpendicular ante la presencia de perturbaciones, así el péndulo invertido regresa a la posición vertical cuando al carro se le aplica una fuerza de control apropiada. Para mover el carro este se conecta mediante una cita de transmisión al motor que proporcionará un par proporcional a la tensión aplicada. El lazo de control se cierra mediante la lectura de la posición del carro y del péndulo con encoder. Al final se regresa el carro a una posición inicial de referencia.

Lo que se pretende con este trabajo es que los alumnos puedan probar sus algoritmos de control de forma didáctica y dinámica, y de esta manera complementar sus conocimientos teóricos. Dado que el control automático se ha convertido en parte importante de los procesos industriales, tener un sistema que permita la evaluación y comparación de diferentes algoritmos de control es una herramienta que nos da una

gran ventaja con respecto a las simulaciones comunes. Es por esto que se pretende retomar esta plataforma experimental y actualizarla, para que se pueda contar con un sistema real con el cual los estudiantes pudieran poner en práctica sus conocimientos.

En resumen, el modelo del péndulo invertido es un ejemplo clásico de control con múltiples aplicaciones, principalmente en el control automático. Con este se pueden comprender los principios básicos del control y de la robótica, además el modelo del péndulo invertido puede usarse como base para otros sistemas.

Gracias a esta versatilidad es muy usado tanto en prácticas como en proyectos de investigación, el sistema es conocido por su posibilidad de analogía para el diseño de diversos controladores y aplicaciones didácticas. En su forma más simple está constituido por carro que se mueve libremente a lo largo de un eje y una barra cilíndrica con libertad de oscilar que pende de él. El péndulo invertido es un sistema mecánico de fase no mínima inestable en lazo abierto, que emula la dinámica de un artefacto volador.

En la literatura científica se muestran diversos trabajos relacionados con el modelo P-I, y se usa ampliamente hoy en día y se ha convertido en un desafío de ingeniería común para los investigadores debido a la importante de aplicación en la vida práctica. Algunas de las aplicaciones actuales de los P-I se presentan en los cohetes al lanzar, la guía de misiles, en el brazo robótico cuando se comporta como un péndulo invertido controlado. La aplicación comercial para el modelo P-I es el Segway, los sismómetros, robots humanoides que son estables caminando bajo un modelo de péndulo invertido lineal [6].

En este caso nos enfocaremos en una aplicación específica, la cual es el caso de los monopatines eléctricos, de los cuales existen varios tipos, por ejemplo: Patinetes eléctricos Scooter, Hoverboard, Monociclo Eléctrico, Onewheel y E-Wheel Scooter. Estos se describen brevemente a continuación.

Patinetes eléctricos Scooter

Los Patinetes eléctricos Scooter poseen un manubrio el cual le sirve al tripulante para mantener el equilibrio. Estos dispositivos se pueden pensar como un péndulo invertido sobre un carro móvil, ya que el tripulante debe mantener el equilibrio para no caerse al momento de acelerar o frenar y el scooter es el carro móvil sobre el cual el péndulo está sujeto [7].



Figura 5: Scooter [7].

Hoverboard

Los Hoverboard nos permiten tener las manos libres, ya que no poseen manubrio, y también por ese motivo se puede pensar que es difícil de controlar. Pero estos dispositivos cuentan con sensores que ayudan a percibir a la persona que lo utiliza para hacer uso del giroscopio integrado que ayuda a equilibrar el centro de gravedad del cuerpo que se encuentra sobre el Hoverboard [8].



Figura 6: Hoverboard [8].

Monociclo Eléctrico

El Monociclo Eléctrico es un patinete eléctrico con una sola rueda. A semejanza del Hoverboard, es el mismo patinete quien encuentra el equilibrio basándose en el centro de gravedad del cuerpo de su usuario. Este tipo de dispositivo cuenta con asiento y un manubrio para controlar su movimiento.



Figura 7: Monociclo Eléctrico con asiento y manubrio [9].

Onewheel

Un monopatín eléctrico auto estabilizado con una sola rueda de gran tamaño situada en el centro de la tabla esto es un Onewheel. Cuenta con un motor de 500 W, mientras que el controlador y la batería van situados en el interior de la tabla. El avance del Onewheel se realiza mediante el control del peso del cuerpo, gracias a los giróscopos y acelerómetros que posee instalados en su interior [10].



Figura 8: Monopatín eléctrico tipo patineta [10].

E-Wheel Scooter

Monociclo eléctrico Veteran Sherman S, auto estabilizad, cuenta con un motor de 3500 W, es capaz de alcanzar los 100 km/h, pero lo recomendable es no rebasar los 30 km/h [11].



Figura 9: Monopatín sin manubrio zuimi [11].

Patinete eléctrico de 1500 W, monopatín de una rueda con auto equilibrio, asiento de motocicleta.



Figura 10: Monopatín con manubrio [12].

En los últimos años la tecnología ha avanzado de tal forma que casi todos los sistemas actuales son automáticos, por lo que es importante contar con plataformas experimentales donde los estudiantes de la Facultad de Ciencias de la Electrónica puedan probar sus algoritmos de control y observar su comportamiento, esto con la finalidad de contribuir a mejoramiento de su aprendizaje.

El péndulo invertido sobre base móvil es un sistema interesante y relevante en el campo de la ingeniería, la robótica y la teoría de control por varias razones:

- Complejidad del control: el péndulo invertido sobre base móvil representa un desafío en el diseño de sistemas de control debido a su naturaleza no lineal. El hecho de que el péndulo esté montado en un carro que se mueve introduce dificultades adicionales para mantener el equilibrio.
- Aplicaciones en la vida real: el estudio del péndulo invertido sobre base móvil tiene aplicaciones prácticas en el diseño y control de sistemas físicos del mundo real. La capacidad de estabilizar un péndulo en un entorno dinámico y cambiante es relevante para el control de robots manipuladores, androides, vehículos autónomos y sistemas de estabilización.

- Desarrollo de algoritmos de control robustos: investigar y diseñar estrategias de control para el péndulo invertido sobre base móvil impulsa el desarrollo de algoritmos de control más robustos. Estos algoritmos son relevantes en diversas aplicaciones de la ingeniería y la robótica.
- Estudio de sistemas no lineales: la complejidad del péndulo invertido sobre base móvil lo convierte en un excelente ejemplo para comprender y controlar sistemas no lineales. La comprensión de sistemas no lineales es importante en la ingeniería, ya que muchos sistemas del mundo real tienen comportamientos no lineales que deben ser controlados.
- Investigación en aprendizaje automático y control adaptativo: la aplicación de técnicas de aprendizaje automático y control adaptativo para estabilizar el péndulo invertido sobre base móvil es un campo de investigación activo. Estos enfoques pueden permitir sistemas más flexibles y adaptables, lo que es esencial en entornos cambiantes [13].

Por esto, contar con una plataforma experimental basada en el péndulo invertido sobre base móvil es de gran utilidad para los alumnos, ya que, debido a su complejidad, sus desafíos de control y sus aplicaciones prácticas, es una muy buena herramienta para reforzar sus conocimientos sobre estrategias de control, lo que les permitirá mejorar sus habilidades de diseño de sistemas robóticos, de control y de automatización en entornos dinámicos.

Objetivo general

Actualizar el modelado, control y sistema electrónico de un péndulo invertido sobre base móvil

Objetivos particulares

1. Estudiar la metodología de parámetros agrupados.
2. Desarrollo del modelo matemático del péndulo invertido bajo el modelado de parámetros agrupados.
3. Estudio del sistema embebido.
4. Diseño de la actualización de la electrónica de potencia y control del péndulo invertido.
5. Elección del algoritmo de control.
6. Implementación del controlador elegido en el sistema embebido.
7. Realizar pruebas experimentales.
8. Publicación de resultados.

9. Escritura de tesis.

La tesis está estructurada en seis capítulos, en los cuales se describen a continuación.

En el **capítulo I** se describe al péndulo invertido sobre base móvil, se presenta un diagrama del robot, así como la descripción de cada una de sus partes.

En el **capítulo II** se presenta el proceso de obtención del modelo dinámico de un sistema péndulo invertido sobre una base móvil. Para ello, se introduce la metodología de parámetros agrupados junto con el método de Euler-Lagrange. Posteriormente, se desarrolla el modelo cinemático del sistema, seguido del modelo dinámico correspondiente. Finalmente, se definen los parámetros utilizados en la simulación y se implementan dos esquemas de control con los cuales se realiza la simulación del comportamiento del sistema.

En el **capítulo III** se presentan los cambios realizados a la electrónica del robot, comenzando con la etapa de potencia, seguido de la electrónica para la adquisición de señales y por último se presenta la integración de toda la electrónica en una estructura que facilite su transporte.

En el **capítulo IV** se describe el firmware proporcionado en el laboratorio de robótica y control, el cual consiste en un microcontrolador embebido en una tarjeta de adquisición FPGA, también se describe la técnica de decodificación de los encoder, así como los problemas que se presentan durante la digitalización y como se utiliza el filtrado para solucionar este problema.

En el **capítulo V** se describe el software utilizado para la configuración del sistema embebido, el cual consta de un lenguaje de programación en alto nivel, un programa desarrollado en LabVIEW para compilar dicho código y otro programa de usuario en el cual se carga el código al FPGA para comenzar la ejecución.

En el **capítulo VI** se presentan los resultados experimentales obtenidos al implementar los dos controladores seleccionados, así como la comparación de estos con los resultados simulados.

Capítulo 1

Descripción del robot

A continuación se describe al péndulo invertido sobre base móvil, el cual consta de una base fija que soporta a los rieles sobre los cuales se desplaza una base móvil que a su vez lleva consigo al péndulo al péndulo invertido, el mecanismo es un sistema subactuado, ya que cuenta solo con un motor y posee dos grados de libertad. En este caso el motor le inyecta energía al primer grado que es la base móvil, el cual después debido a su movimiento le transfiere energía al segundo grado de libertad, el cual es el péndulo.

1.1. Diagrama a bloques

En la Figura 1.1 se presenta el diagrama a bloques del sistema, en el cual se pueden observar cómo se relacionan cada uno de los elementos que componen al robot, la descripción del diagrama se presenta a continuación.

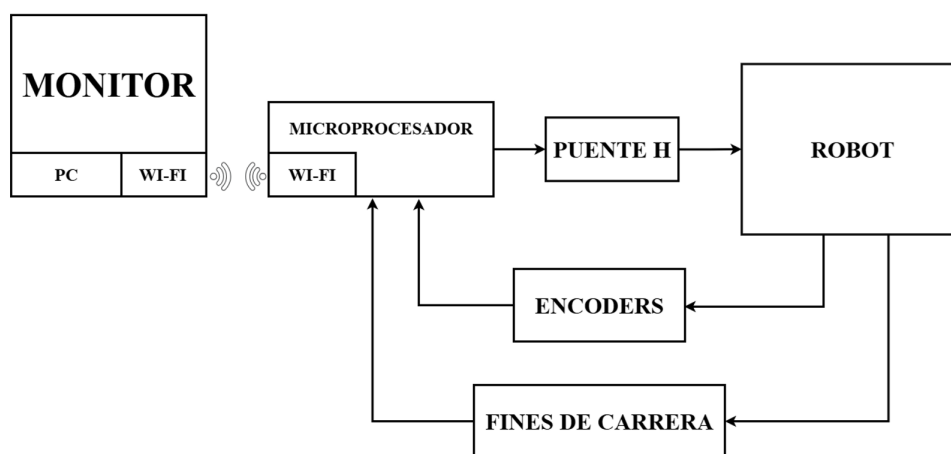


Figura 1.1: Diagrama a bloques del robot.

Como se observa la Figura 1.1 el sistema está compuesto por una PC, la cual consta de un monitor con el cual el usuario puede desarrollar su código en lenguaje D para configurar al microcontrolador que se encuentra en el robot, así como visualizar los datos que este le mande de vuelta sobre la variable seleccionada, la comunicación

entre la PC y el microcontrolador se realiza por medio de WI-FI, esta comunicación solo es utilizada para la comunicación de configuración del microcontrolador y para monitoreo del sistema, todo el procesamiento de señales y el controlador se encuentran configurados en el microcontrolador por lo que los tiempos de envío y recepción del WI-FI no afectan al rendimiento del sistema ya que todo se hace directamente en el microcontrolador.

Una vez que ya se ha descargado el programa al microcontrolador, este ya se encuentra recibiendo señales del robot, como son las posiciones de la base móvil y del péndulo, así como los dos finales de carrera colocados en el riel para evitar colisiones, con estos datos, el controlador implementado calcula el par que se le debe inyectar al robot para realizar la tarea asignada, que en nuestro caso es mantener al péndulo en una posición vertical a 180° , una vez que se obtiene este dato, el microcontrolador envía una señal de PWM que llega a las entradas del circuito del puente H, esta señal es de dos líneas ya que depende del sentido de giro necesario, la señal es enviada por una o la otra, con esto la base móvil recibe energía para desplazarse en un sentido u otro, con el fin de ir oscilando al péndulo hasta lograr estabilizarlo en la vertical superior.

1.2. Sistema mecánico

En el Laboratorio de Robótica y control de la Maestría en Ciencias de la Electrónica, Opción Automatización se cuenta con un péndulo invertido sobre base móvil, el cual se muestra en la Figura 1.2. A continuación se describen cada uno de los elementos que componen al sistema.

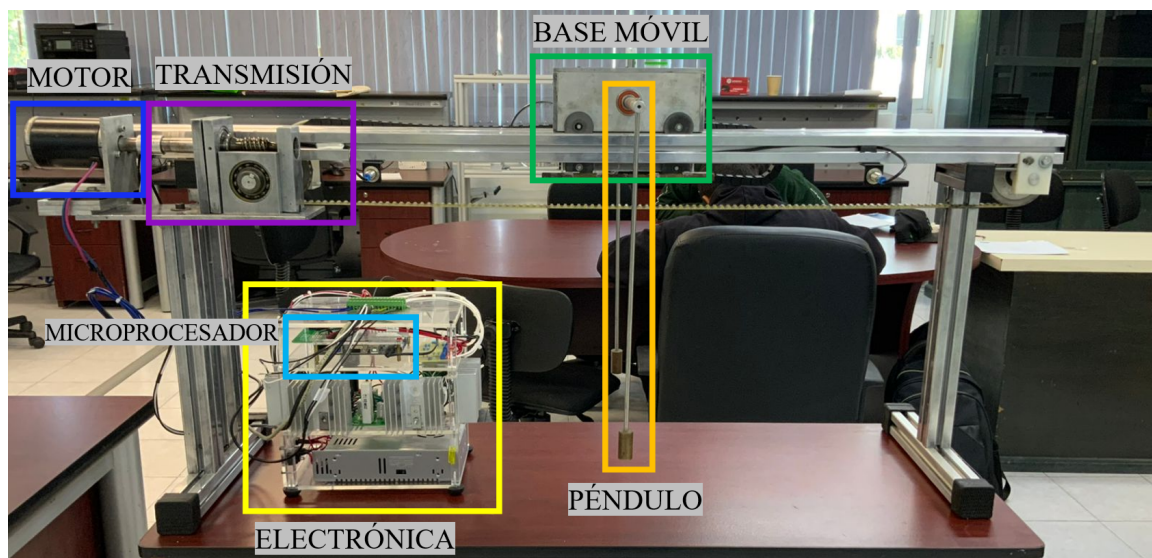


Figura 1.2: Péndulo invertido sobre base móvil del laboratorio del MCEA.

El péndulo invertido sobre base móvil consta de un carro, el cual está formado por una caja hueca fabricado de aluminio con 4 llantas de Nylamid SL que reposan sobre el riel y cuatro llantas de Nylamid SL que lo sujetan por debajo del riel, cuenta con dos barras de aluminio, una a cada lado del carro con el propósito de mantener el equilibrio y estos puedan girar libremente alrededor del eje que las une al carro, al final de cada barra se encuentra una barra más delgada de aluminio con un peso de bronce en las puntas las cuales forman al péndulo como se muestra en la Figura 1.3 [15].

El péndulo invertido sobre base móvil consta de un carro, el cual está formado por una caja hueca fabricado de aluminio con 4 llantas de Nylamid SL que reposan sobre el riel y cuatro llantas de Nylamid SL que lo sujetan por debajo del riel, cuenta con dos barras de aluminio, una a cada lado del carro con el propósito de mantener el equilibrio y estos puedan girar libremente alrededor del eje que las une al carro, al final de cada barra se encuentra una barra más delgada de aluminio con un peso de bronce en las puntas las cuales forman al péndulo como se muestra en la Figura 1.3 [15].

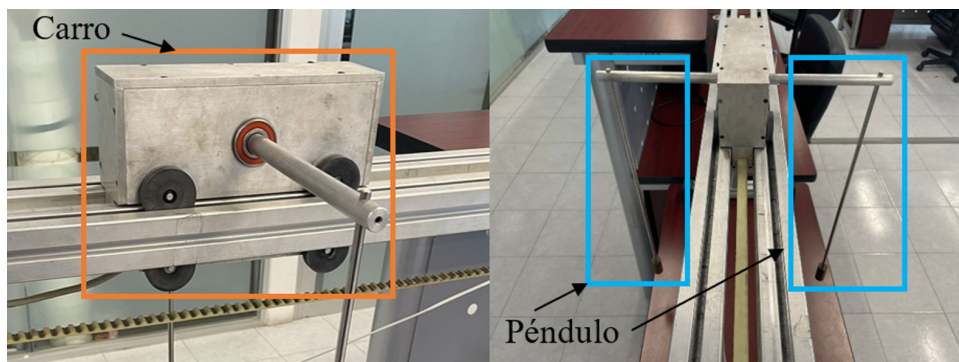


Figura 1.3: Carrito del péndulo invertido.

Las medidas del carro se muestran en la Figura 1.4, así como en la Figura 1.5 podemos observar las medidas de las barras que forman al péndulo y su peso de bronce.

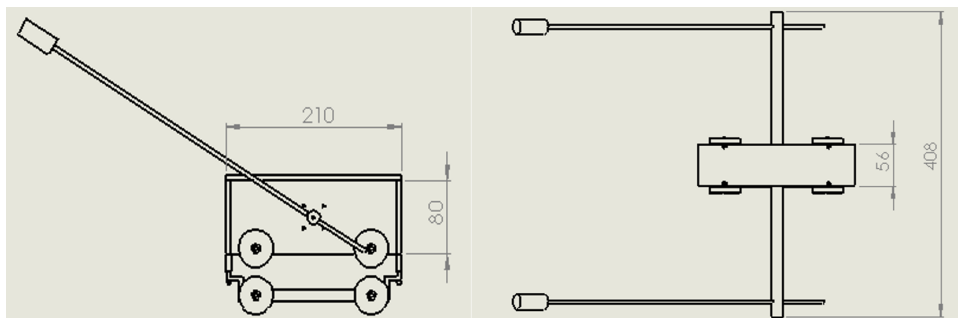


Figura 1.4: Vista lateral y superior del carro [15].

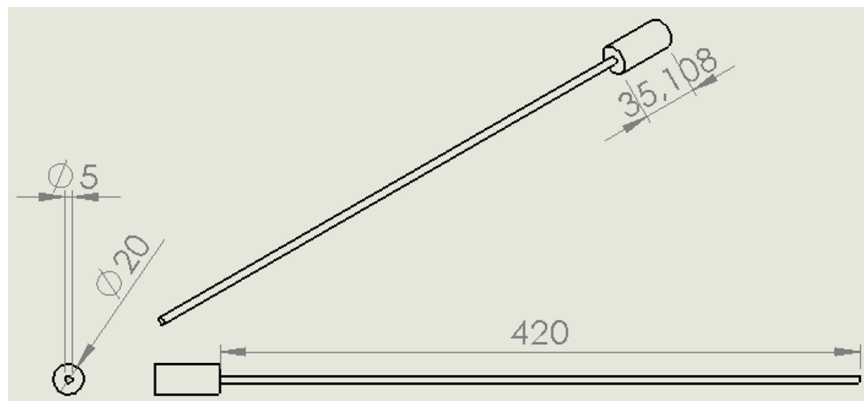


Figura 1.5: Medidas de las varillas del péndulo [15].

El carro se encuentra sobre una estructura fija construida en perfil de aluminio #21223 de 40x40 mm 6051, el riel consta de dos barras, y está montado sobre dos soportes en los extremos como se observa en la Figura 1.6.

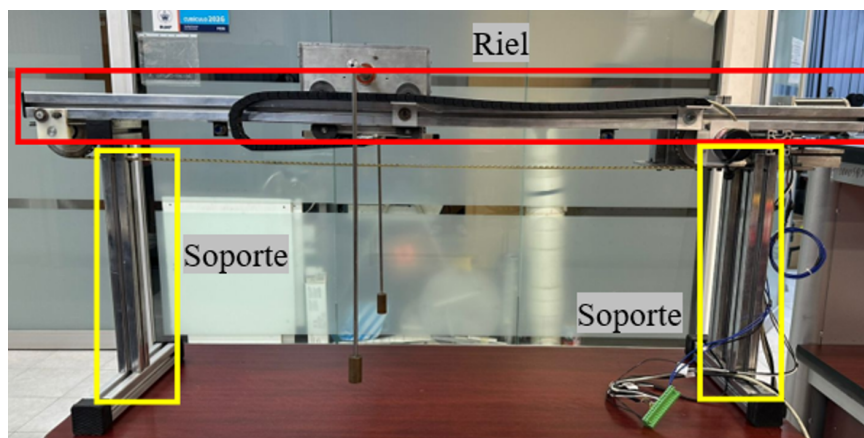


Figura 1.6: Base fija.

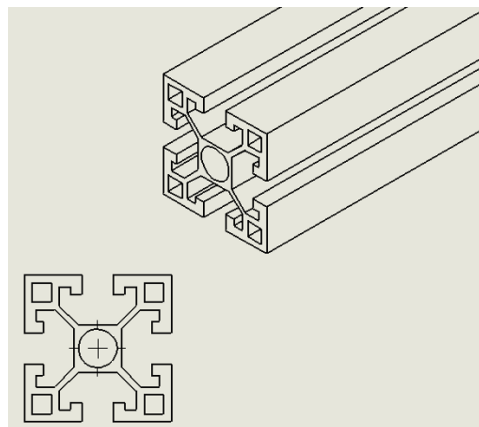


Figura 1.7: Perfil de aluminio utilizado en la base [15].

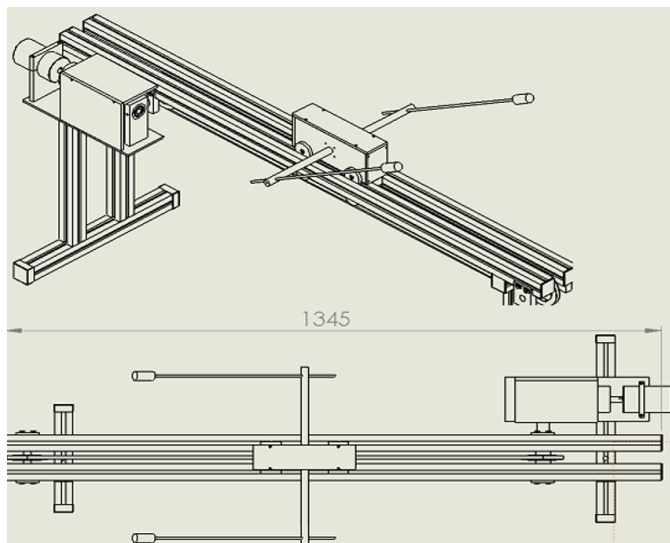


Figura 1.8: Dibujo de la base fija y riel del péndulo invertido [15].

En la Figura 1.9, se muestran las patas con sus medidas, las cuales se encuentran en milímetros, donde el dibujo se obtuvo de SolidWorks.

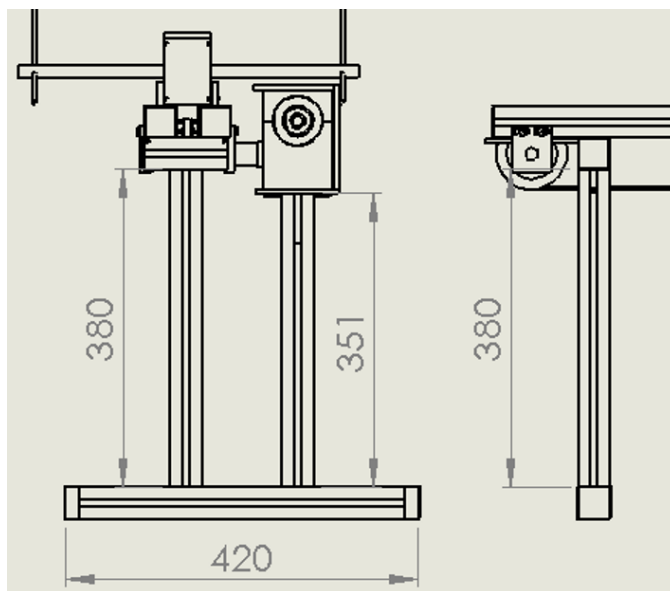


Figura 1.9: Soportes de la estructura del péndulo invertido [15].

En la estructura fija se encuentran dos bases donde se sitúan dos poleas que son parte de la transmisión que conectan el carro con el motor, una de las bases es fija y la otra es ajustable para poder tensar la banda dentada que se sitúa entre ellas y la cual está conectada al carro como se puede apreciar en la Figura 1.10.

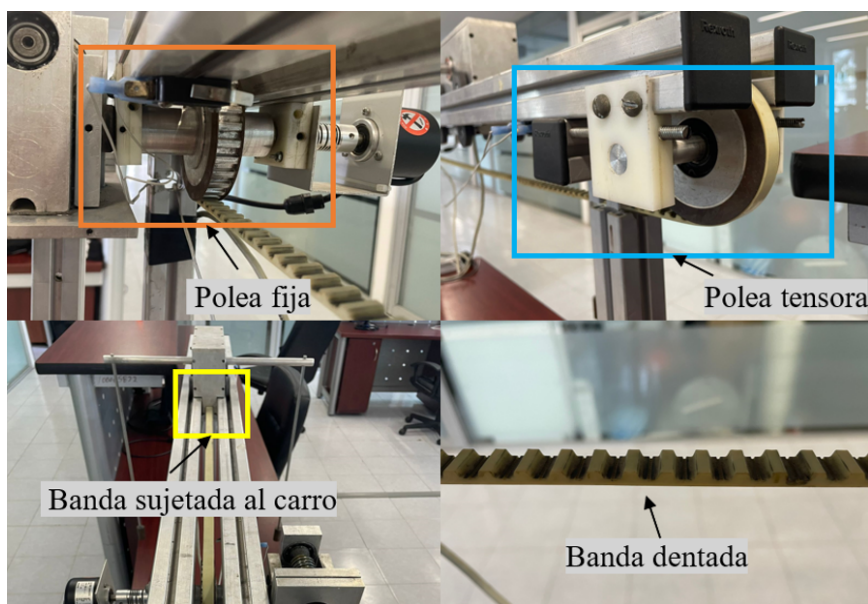


Figura 1.10: Poleas y banda dentada.

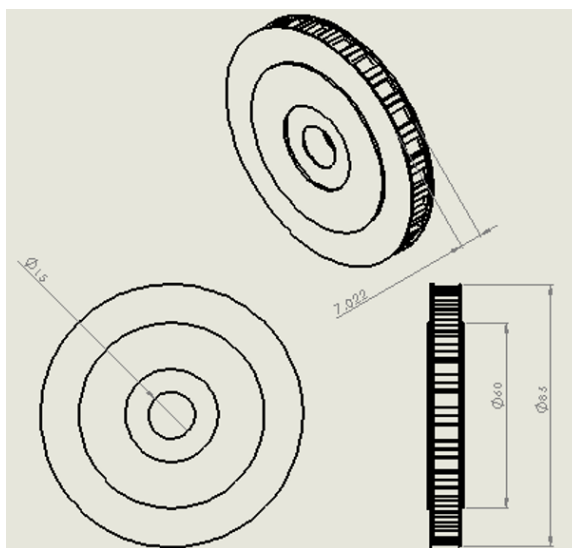


Figura 1.11: Medidas de las poleas del péndulo invertido [15].

De igual forma la estructura posee una base a un costado donde se sitúa el motor junto con una caja de aluminio en la cual se encuentra un mecanismo de tornillo sinfín y corona, este mecanismo se encuentra cerrado para evitar accidentes en el momento en que el sistema se encuentre en funcionamiento ya que el motor puede girar a velocidades altas, en este sistema de transmisión el eje del tornillo se encuentra acoplado al eje del motor y el eje de la corona se encuentra acoplado al eje de la polea fija, como se puede apreciar en la Figura 1.12.

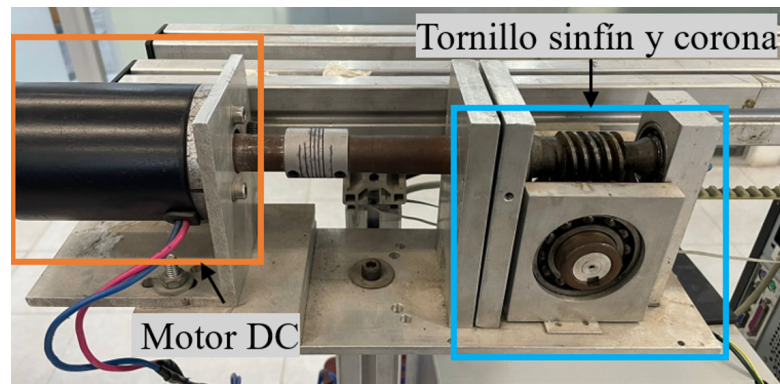


Figura 1.12: Base del motor con mecanismo de tornillo sinfín y corona.

Para conocer algunos datos del sistema se utilizan sensores, los cuales son dispositivos que miden una variable física y la transforman en una señal eléctrica que es procesada por un sistema de adquisición de datos, el péndulo cuenta con dos encoders, el encoder es un transductor rotativo que transforma un movimiento angular en una serie de pulsos digitales, se pueden usar para controlar desplazamientos de tipo angular o de tipo lineal, uno de ellos se encuentra dentro del carro, el cual está conectado a las barras de aluminio de los lados para poder medir la posición del péndulo, el otro se encuentra conectado al eje de la polea fija, este se utiliza para medir la posición del motor, ambos sensores tienen la misma precisión, como se muestra en la Figura 1.13.

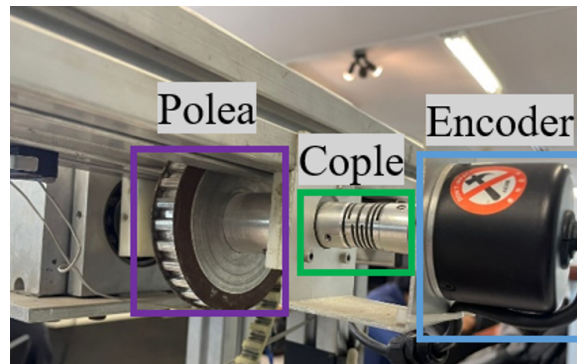


Figura 1.13: Encoder.

El péndulo invertido también cuenta con dos finales de carrera, Figura 1.14, los cuales se sitúan debajo del riel para limitar la zona en la que el carro puede moverse libremente ya que en el caso de que el controlador falle o el carro necesite desplazarse una mayor distancia que la del riel, este no se salga o choque con el motor y lo dañe dado que el carro alcanza una gran velocidad esto puede ser muy peligroso. Para los sensores de final de carrera se usaron sensores de contacto microswitch que son dispositivos electromecánicos que detectan cambios a través del contacto físico, de esta forma se usan como paradas de emergencia. Constan de un accionador unido mecánicamente a una serie de contactos, cuando el objeto entra en contacto con el accionador el dispositivo activa los contactos para establecer o interrumpir la conexión eléctrica. Son resistentes, de fácil instalación y confiables [24].

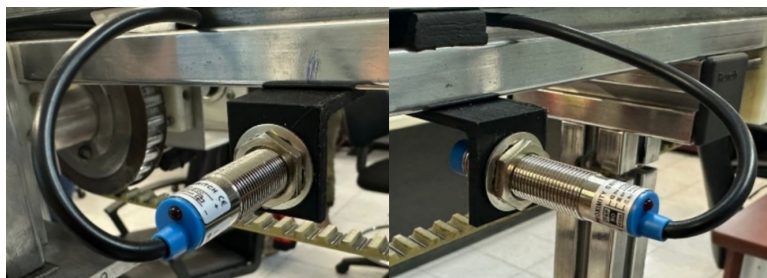


Figura 1.14: Finales de carrera.

El motor con el que cuenta el péndulo invertido es un motor de corriente continua DOGA, del cual solo se conoce que su voltaje nominal es de 24 VDC a 20 A.

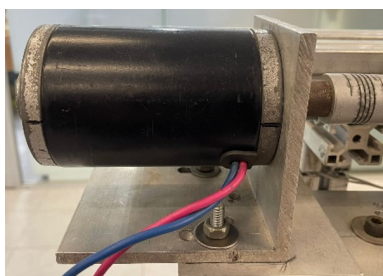


Figura 1.15: Motor DC.

1.3. Conclusión

El péndulo invertido sobre base móvil del laboratorio de Robótica y Control de la MCEA es un sistema que integra elementos mecánicos, electrónicos y de control para desarrollar y probar estrategias avanzadas de automatización. Su diseño mecánico destaca por la construcción robusta del carro en aluminio, equipado con llantas de Nylamid SL que garantizan un desplazamiento fluido sobre un riel de perfil de aluminio. Este riel está soportado por una base fija diseñada para estabilidad y durabilidad, complementada con un sistema de transmisión basado en poleas y una banda dentada que conecta al carro con el motor.

El sistema de transmisión incluye un mecanismo de tornillo sinfín y corona, protegido para evitar accidentes durante la operación, lo que asegura un funcionamiento seguro incluso a altas velocidades. Los sensores incorporados, como encoders y finales de carrera, aportan una precisión significativa en la medición y control del movimiento. Los encoders permiten monitorear tanto la posición del péndulo como la del motor, mientras que los finales de carrera garantizan límites físicos para evitar daños en situaciones críticas.

En conjunto, el diseño del sistema es una plataforma experimental sólida y versátil, ideal para implementar y probar algoritmos avanzados de control. Este diseño no solo asegura la funcionalidad y seguridad del equipo, sino que también provee un entorno confiable para la investigación y la enseñanza en el ámbito de la robótica y la automatización.

Capítulo 2

Modelo dinámico

En este capítulo se presenta el proceso de obtención del modelo dinámico de un sistema péndulo invertido sobre una base móvil. Para ello, se introduce la metodología de parámetros agrupados junto con el método de Euler-Lagrange. Posteriormente, se desarrolla el modelo cinemático del sistema, seguido del modelo dinámico correspondiente. Finalmente, se definen los parámetros utilizados en la simulación y se implementan dos esquemas de control con los cuales se realiza la simulación del comportamiento del sistema.

2.1. Metodología para la obtención del modelo dinámico

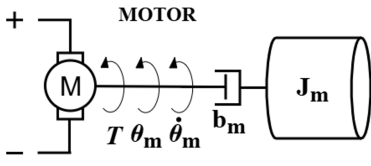
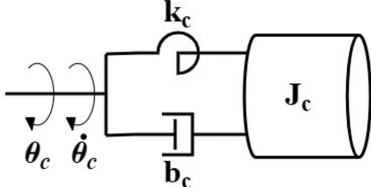
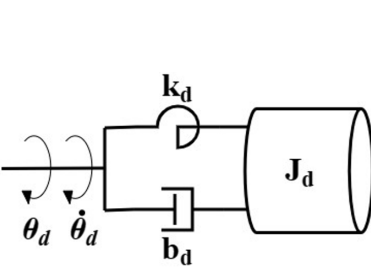
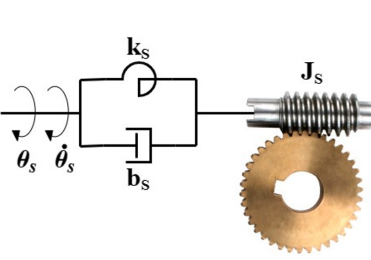
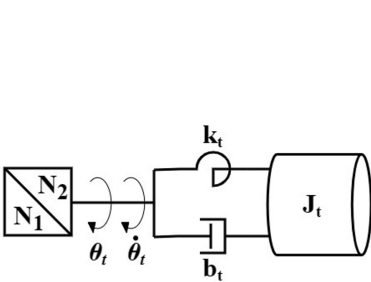
La metodología a emplear para la obtención del modelo dinámico consiste en dos pasos, la primera es realizar el análisis de parámetros agrupados a nuestro robot y la segunda es aplicar el método de Euler-Lagrange al análisis del paso 1, esto se describe a continuación.

2.1.1. Análisis de parámetros agrupados

El análisis de parámetros agrupados consiste en identificar los elementos que conforman a nuestro sistema para después señalar los parámetros involucrados en su movimiento, una vez realizado esto se prosigue a realizar un diagrama de parámetros agrupados en donde estos elementos se representan de forma visual para continuar con la segunda parte del análisis.

Simbología

A continuación se presenta una tabla que contiene las diferentes representaciones gráficas que se utilizarán para desarrollar el análisis de nuestro modelo dinámico, ver Tabla 2.1.

Nombre	Símbolo	Significado y Unidades
Motor		τ = Par entregado por el motor. b_m = Coeficiente de amortiguamiento torsional del eje del motor. θ_m = Es el desplazamiento angular del eje motor. $\dot{\theta}_m$ = Es la velocidad angular del eje motor. J_m = Momento de inercia del eje motor.
Cople		k_c = Rigidez torsional del cople. b_c = Coeficiente de amortiguamiento torsional del cople. θ_c = Es el desplazamiento angular del cople. $\dot{\theta}_c$ = Es la velocidad angular del cople. J_c = Momento de inercia del cople.
Eje Acoplador de Tornillo Sinfín		k_d = Rigidez torsional del eje acoplador tornillo sinfín. b_d = Coeficiente de amortiguamiento torsional del eje acoplador tornillo sinfín. θ_d = Es el desplazamiento angular del eje acoplador tornillo sinfín. $\dot{\theta}_d$ = Es la velocidad angular del eje acoplador tornillo sinfín. J_d = Momento de inercia del eje acoplador tornillo sinfín.
Tornillo Sinfín		k_s = Rigidez torsional del tornillo sinfín. b_s = Coeficiente de amortiguamiento torsional del tornillo sinfín. θ_s = Es el desplazamiento angular del tornillo sinfín. $\dot{\theta}_s$ = Es la velocidad angular del tornillo sinfín. J_s = Momento de inercia del tornillo sinfín.
Eje de Salida de la Corona		N_1 = Numero de dientes del engrane. N_2 = Numero de vueltas del tornillo. k_t = Rigidez torsional del eje de salida de la corona. b_t = Coeficiente de amortiguamiento torsional del eje de salida de la corona. θ_t = Es el desplazamiento angular del eje de salida de la corona. $\dot{\theta}_t$ = Es la velocidad angular del eje de salida de la corona. J_t = Momento de inercia del eje de salida de la corona.

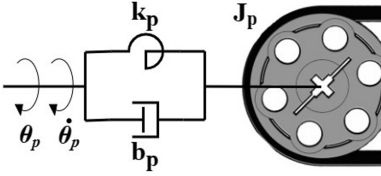
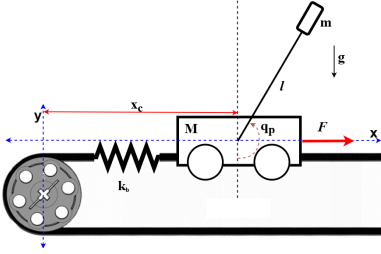
Polea		k_p = Rigidez torsional de la polea. b_p = Coeficiente de amortiguamiento torsional de la polea. θ_p = Es el desplazamiento angular de la polea. $\dot{\theta}_p$ = Es la velocidad angular de la polea. J_p = Momento de inercia de la polea.
Base Móvil con Péndulo		M = Masa del carro. m = Masa del péndulo. l = Longitud del péndulo. g = Gravedad. x_c = Desplazamiento en x del carro. q_p = posición del péndulo respecto a la vertical superior. F = Fuerza aplicada al carro.

Tabla 2.1: Símbolos de parámetros agrupados.

2.1.2. Método de Euler-Lagrange

Esta técnica de modelado se basa en examinar el cambio de energía del sistema dinámico. En este caso, los sistemas dinámicos considerados pueden contener componentes de diversas disciplinas de la ingeniería. En particular, podemos tener componentes mecánicos, eléctricos, de fluidos y/o térmicos presentes en nuestro modelo de sistema dinámico. El método de parámetros agrupados permite considerar todas las energías del sistema utilizando la mecánica clásica para describir cada componente [18].

Las ecuaciones de Euler-Lagrange parten del Lagrangiano de un sistema:

$$\mathcal{L}(\mathbf{q}, \dot{\mathbf{q}}) = \mathcal{K}(\mathbf{q}, \dot{\mathbf{q}}) - \mathcal{U}(\mathbf{q}), \quad (2.1)$$

definido como la diferencia entre la energía cinética $\mathcal{K}(\mathbf{q}, \dot{\mathbf{q}})$ menos la energía Potencial $\mathcal{U}(\mathbf{q})$ del sistema. Utilizando de la ecuación (2.1), se define la ecuación de movimiento de Euler-Lagrange, con la cual se obtiene la ecuación de movimiento generalizada de Lagrange;

$$\frac{d}{dt} \left[\frac{\partial \mathcal{K}(\varphi, \dot{\varphi})}{\partial \dot{\varphi}} \right] - \frac{\partial \mathcal{K}(\varphi, \dot{\varphi})}{\partial \varphi} + \frac{\partial \mathcal{U}(\varphi)}{\partial \varphi} + \frac{\partial \mathcal{D}t(\varphi, \dot{\varphi})}{\partial \dot{\varphi}} = \frac{\partial \delta \mathcal{W}(\varphi, \dot{\varphi})}{\partial \dot{\varphi}} - \tau_s, \quad (2.2)$$

donde $\mathcal{L}$ representa el Lagrangiano del sistema, $\mathcal{D}t$ es la energía disipativa del sistema, $\delta \mathcal{W}$ es el trabajo virtual que interactúa en el sistema agrupado, τ_s es una perturbación o fuerza externa o un error no modelado.

En el libro Analytical System Dynamics Modeling and Simulation nos presentan la Tabla 2.2 de variables fundamentales [18].

	Esfuerzo e	Flujo f	Desplazamiento q	Momento p
Traslación mecánica	Fuerza, F	Velocidad, v	Posición, x	Momento lineal, p
Rotación mecánica	torque, τ	velocidad angular, ω	ángulo, θ	momento angular, H
Eléctrico	voltaje, v	corriente, i	carga, q	enlace de flujo, λ
Fluidos	presión, P	tasa de flujo volumétrico, Q	volumen, V	momento de presión, Γ
Térmico	temperatura, T	tasa de flujo de entropía, $\dot{S}$	entropía, S	-

Tabla 2.2: Variables fundamentales.

2.1.3. Esfuerzos generalizados

Como vimos en la ecuación en la ecuación del método de parámetros agrupados, todas las energías del sistema están igualadas a $\delta\mathcal{W}$, lo que nos indica los trabajos realizados o fuerzas generalizadas del sistema. Como sabemos el trabajo es la fuerza que se aplica sobre un cuerpo para desplazarlos de un punto a otro.

Normalmente, las fuentes que generan un tipo de esfuerzo se dan en términos de configuración de desplazamiento, no los desplazamientos generalizados. La transformación de coordenadas que relaciona la configuración de desplazamiento con el desplazamiento generalizado se utiliza en la expresión de trabajo virtual para determinar los esfuerzos generalizados.

El trabajo virtual viene dado por la siguiente expresión:

$$\delta\mathcal{W} = \sum_{i=1}^n e_i \delta q_i, \quad (2.3)$$

donde $\delta\mathcal{W}$ es el trabajo virtual, δq_i son los desplazamientos virtuales, e_i son los esfuerzos aplicados, a continuación se presentan los distintos trabajos virtuales realizados por distintos esfuerzos.

Disciplina	Trabajo Virtual	
	$e\delta q$	$f\delta p$
Traslación mecánica	$F\delta x$	$v\delta p$
Rotación mecánica	$\tau\delta\theta$	$\omega\delta H$
Eléctrica	$v\delta q$	$i\delta\lambda$
Flujo	$P\delta V$	$W\delta\Gamma$
Térmico	$T\delta S$	Nulo

Tabla 2.3: Trabajos virtuales

2.2. Modelo cinemático

En el modelo cinemático se analiza el movimiento del robot con respecto a un sistema de referencia, este nos describe el comportamiento analítico del movimiento del sistema como una función del tiempo, y nos da las coordenadas de la posición y orientación del extremo final.

Para el desarrollo de la cinemática del péndulo invertido sobre base móvil, nos apoyamos de la Figura 2.1.

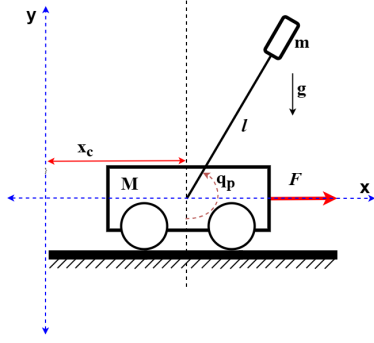


Figura 2.1: Diagrama del péndulo invertido.

Parámetros	y variables
Símbolo	Descripción
x_c	Posición en el eje x del carro.
q_p	Posición angular del péndulo.
M	Masa del carro.
m	Masa del péndulo.
l	Longitud del péndulo.
g	Constante de gravedad.
F	Fuerza aplicada.

Tabla 2.4: Variables del péndulo.

A partir de la Figura 2.1 obtenemos las coordenadas del péndulo:

$$x_m = x_c + l \sin(q_p), \quad (2.4)$$

$$y_m = l(1 - \cos(q_p)). \quad (2.5)$$

Ahora necesitamos obtener su velocidad, para lo cual se derivan las ecuaciones anteriores y sumamos sus cuadrados:

$$\dot{x}_m = \dot{x}_c + l\dot{q}_p \cos(q_p), \quad (2.6)$$

$$\dot{y}_m = l\dot{q}_p \sin(q_p), \quad (2.7)$$

$$\dot{x}_m^2 = (\dot{x}_c + l\dot{q}_p \cos(q_p))^2, \quad (2.8)$$

$$\dot{x}_m^2 = \dot{x}_c^2 + 2l\dot{x}_c\dot{q}_p \cos(q_p) + l^2\dot{q}_p^2 \cos^2(q_p), \quad (2.9)$$

$$\dot{y}_m^2 = (l\dot{q}_p \sin(q_p))^2, \quad (2.10)$$

$$\dot{y}_m^2 = l^2\dot{q}_p^2 \sin^2(q_p), \quad (2.11)$$

$$\dot{x}_m^2 + \dot{y}_m^2 = \dot{x}_c^2 + 2l\dot{x}_c\dot{q}_p \cos(q_p) + l^2\dot{q}_p^2 \cos^2(q_p) + l^2\dot{q}_p^2 \sin^2(q_p), \quad (2.12)$$

$$\dot{x}_m^2 + \dot{y}_m^2 = \dot{x}_c^2 + 2l\dot{x}_c\dot{q}_p \cos(q_p) + l^2\dot{q}_p^2 (\sin^2(q_p) + \cos^2(q_p)), \quad (2.13)$$

$$\dot{x}_m^2 + \dot{y}_m^2 = \dot{x}_c^2 + 2l\dot{x}_c\dot{q}_p \cos(q_p) + l^2\dot{q}_p^2. \quad (2.14)$$

Obteniendo estas relaciones ahora podemos comenzar con el análisis del modelo dinámico de nuestro sistema.

2.3. Modelo dinámico

El modelo dinámico del péndulo invertido sobre base móvil se desarrolla utilizando la técnica de modelado de parámetros agrupados. Para comenzar el análisis dinámico del sistema, primero definimos los parámetros que vamos a considerar de nuestro sistema, en la Figura 2.2, se presenta el diagrama de nuestro sistema.

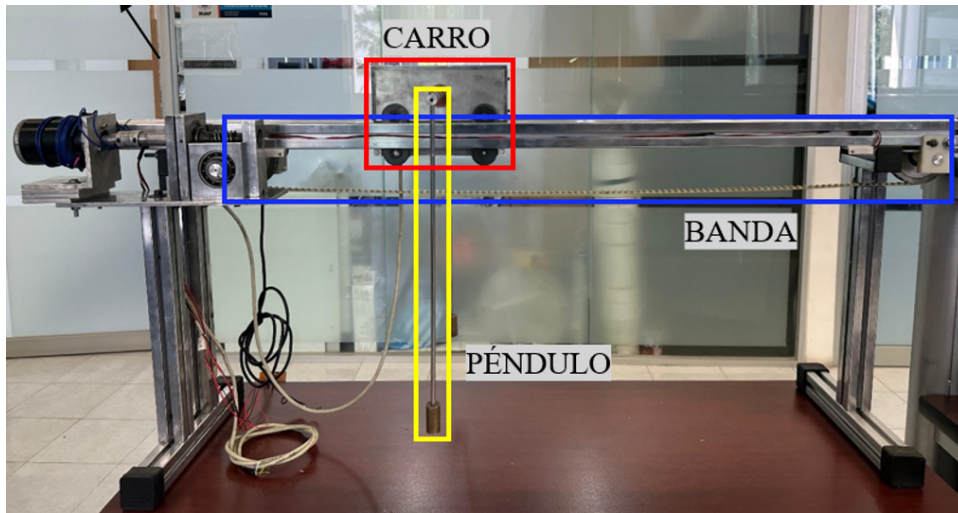


Figura 2.4: Componentes físicos del diagrama del robot (2).

A continuación se presenta la Tabla 2.5 donde se describen cada uno de los parámetros referentes al modelo dinámico.

Símbolo	Significado y Unidades
K_c	Rigidez del eje del cople $[Nm/rad]$
K_d	Rigidez del eje de salida del cople $[Nm/rad]$
K_s	Rigidez del tornillo sinfín $[Nm/rad]$
K_t	Rigidez del eje de salida de la corona $[N/m]$
K_p	Rigidez de la polea $[Nm/rad]$
K_b	Rigidez de la banda $[Nm/rad]$
b_m	Coeficiente de fricción del eje del motor $[Nms/rad]$
b_c	Coeficiente de fricción del cople $[Nms/rad]$
b_d	Coeficiente de fricción del eje de salida del cople $[Nms/rad]$
b_s	Coeficiente de fricción del tornillo sinfín $[Nms/rad]$
b_t	Coeficiente de fricción del eje de salida de la corona $[Nm/s]$
b_p	Coeficiente de fricción de la polea $[Nm/s]$
b_b	Coeficiente de fricción de la banda $[Nm/s]$
b_q	Coeficiente de fricción del péndulo $[Nm/s]$
J_m	Inercia del motor $[kgm^2]$
J_c	Inercia del cople $[kgm^2]$
J_d	Inercia del eje de salida del cople $[kgm^2]$
J_s	Inercia del tornillo sinfín $[kgm^2]$
J_t	Inercia del eje de salida de la corona $[m]$
J_p	Inercia de la polea $[m]$
M	Masa de la base móvil $[m]$
m	Masa del péndulo $[m]$

l	Longitud del péndulo al centro de masa $[m]$
g	Constante gravitacional $[m]$
N_1	Numero de vueltas del tornillo $[m]$
N_2	Numero de dientes de la corona $[m]$

Tabla 2.5: Parámetros del sistema.

Símbolo	Definición
θ_m	Posición angular del eje del motor
θ_c	Posición angular del cople
θ_d	Posición angular del eje de salida del cople
θ_s	Posición angular del tornillo sinfín
θ_t	Posición angular del eje de salida de la corona
θ_p	Posición angular de la polea
x_c	Posición lineal de la base móvil
q_p	Posición angular del péndulo
$\dot{\theta}_m$	Velocidad angular del eje del motor
$\dot{\theta}_c$	Velocidad angular del cople
$\dot{\theta}_d$	Velocidad angular del eje de salida del cople
$\dot{\theta}_s$	Velocidad angular del tornillo sinfín
$\dot{\theta}_t$	Velocidad angular del eje de salida de la corona
$\dot{\theta}_p$	Velocidad angular de la polea
$\dot{x}_c$	Velocidad lineal de la base móvil
$\dot{q}_p$	Velocidad angular del péndulo

Tabla 2.6: Variables del sistema.

Siguiendo la metodología de parámetros agrupados y la ecuación de Lagrange, debemos plantear las ecuaciones generales de energías correspondientes, las cuales quedan de la siguiente manera:

$$\mathcal{K}(\varphi, \dot{\varphi}) = \frac{1}{2}J_m\dot{\theta}_m^2 + \frac{1}{2}J_c\dot{\theta}_c^2 + \frac{1}{2}J_d\dot{\theta}_d^2 + \frac{1}{2}J_s\dot{\theta}_s^2 + \frac{1}{2}J_t\dot{\theta}_t^2 + \frac{1}{2}J_p\dot{\theta}_p^2 + \frac{1}{2}(M + m)\dot{x}_c^2 + ml\dot{x}_c\dot{q}_p \cos(q_p) + \frac{1}{2}ml^2\dot{q}_p^2, \quad (2.15)$$

$$\mathcal{U}(\varphi) = \frac{1}{2}K_c(\theta_c - \theta_m)^2 + \frac{1}{2}K_d(\theta_d - \theta_c)^2 + \frac{1}{2}K_s(\theta_s - \theta_d)^2 + \frac{1}{2}K_t(G\theta_s + \theta_t)^2 + \frac{1}{2}K_p(\theta_t - \theta_p)^2 + \frac{1}{2}K_b(x_c - R\theta_p)^2 + mgl(1 - \cos q_p), \quad (2.16)$$

$$\mathcal{D}(\varphi, \dot{\varphi}) = \frac{1}{2}b_m\dot{\theta}_m^2 + \frac{1}{2}b_c(\dot{\theta}_m - \dot{\theta}_c)^2 + \frac{1}{2}b_d(\dot{\theta}_c - \dot{\theta}_d)^2 + \frac{1}{2}b_s(\dot{\theta}_d - \dot{\theta}_s)^2 + \frac{1}{2}b_t(-\dot{\theta}_t - G\dot{\theta}_s)^2 + \frac{1}{2}b_p(\dot{\theta}_p - \dot{\theta}_t)^2 + \frac{1}{2}b_b(\dot{x}_c - R\dot{\theta}_p)^2 + \frac{1}{2}b_q\dot{q}_p^2, \quad (2.17)$$

$$\delta\mathcal{W}(\varphi) = \tau_m\delta\theta_m. \quad (2.18)$$

Tomando las ecuaciones generales de energía y aplicando la ecuación de Lagrange, se obtiene el sistema de ecuaciones que describe el comportamiento de cada uno de los componentes del robot analizados, estas ecuaciones se desarrollan a continuación.

Para θ_m tenemos que:

$$\frac{\partial \delta W(q, \dot{q})}{\partial \dot{\theta}_m} - \tau_s = \frac{d}{dt} \left[\frac{\partial \mathcal{K}(q, \dot{q})}{\partial \dot{\theta}_m} \right] - \frac{\partial \mathcal{K}(q, \dot{q})}{\partial \theta_m} + \frac{\partial U(q)}{\partial \theta_m} + \frac{\partial \mathcal{D}_t(q, \dot{q})}{\partial \dot{\theta}_m}, \quad (2.19)$$

$$\frac{\partial \mathcal{K}(q, \dot{q})}{\partial \dot{\theta}_m} = J_m \dot{\theta}_m, \quad (2.20)$$

$$\frac{d}{dt} \left[\frac{\partial \mathcal{K}(q, \dot{q})}{\partial \dot{\theta}_m} \right] = J_m \ddot{\theta}_m, \quad (2.21)$$

$$\frac{\partial \mathcal{K}(q, \dot{q})}{\partial \theta_m} = 0, \quad (2.22)$$

$$\frac{\partial U(q)}{\partial \theta_m} = -k_c(\theta_c - \theta_m), \quad (2.23)$$

$$\frac{\partial \mathcal{D}_t(q, \dot{q})}{\partial \dot{\theta}_m} = b_m \dot{\theta}_m + b_c(\dot{\theta}_c - \dot{\theta}_m), \quad (2.24)$$

$$\frac{\partial \delta W(q, \dot{q})}{\partial \dot{\theta}_m} = \tau_m. \quad (2.25)$$

Sustituyendo tenemos que:

$$J_m \ddot{\theta}_m - k_c(\theta_c - \theta_m) + b_m \dot{\theta}_m + b_c(\dot{\theta}_m - \dot{\theta}_c) = \tau_m. \quad (2.26)$$

Despejando $\ddot{\theta}_m$ tenemos que:

$$\ddot{\theta}_m = \frac{\tau_m + k_c(\theta_c - \theta_m) - b_m \dot{\theta}_m - b_c(\dot{\theta}_m - \dot{\theta}_c)}{J_m}. \quad (2.27)$$

Para θ_c tenemos que:

$$\frac{\partial \delta W(q, \dot{q})}{\partial \dot{\theta}_c} - \tau_s = \frac{d}{dt} \left[\frac{\partial \mathcal{K}(q, \dot{q})}{\partial \dot{\theta}_c} \right] - \frac{\partial \mathcal{K}(q, \dot{q})}{\partial \theta_c} + \frac{\partial \mathcal{U}(q)}{\partial \theta_c} + \frac{\partial \mathcal{D}_t(q, \dot{q})}{\partial \dot{\theta}_c}, \quad (2.28)$$

$$\frac{\partial \mathcal{K}(q, \dot{q})}{\partial \dot{\theta}_c} = J_c \dot{\theta}_c, \quad (2.29)$$

$$\frac{d}{dt} \left[\frac{\partial \mathcal{K}(q, \dot{q})}{\partial \dot{\theta}_c} \right] = J_c \ddot{\theta}_c, \quad (2.30)$$

$$\frac{\partial \mathcal{K}(q, \dot{q})}{\partial \theta_c} = 0, \quad (2.31)$$

$$\frac{\partial \mathcal{U}(q)}{\partial \theta_c} = k_c(\theta_c - \theta_m) - k_d(\theta_d - \theta_c), \quad (2.32)$$

$$\frac{\partial \mathcal{D}_t(q, \dot{q})}{\partial \dot{\theta}_c} = -b_c(\dot{\theta}_m - \dot{\theta}_c) + b_d(\dot{\theta}_c - \dot{\theta}_d), \quad (2.33)$$

$$\frac{\partial \delta W(q, \dot{q})}{\partial \dot{\theta}_c} = 0. \quad (2.34)$$

Sustituyendo tenemos que:

$$J_c \ddot{\theta}_c + k_c(\theta_c - \theta_m) - k_d(\theta_d - \theta_c) - b_c(\dot{\theta}_m - \dot{\theta}_c) + b_d(\dot{\theta}_c - \dot{\theta}_d) = 0. \quad (2.35)$$

Despejando $\ddot{\theta}_c$ tenemos que:

$$\ddot{\theta}_c = \frac{-k_c(\theta_c - \theta_m) + k_d(\theta_d - \theta_c) + b_c(\dot{\theta}_m - \dot{\theta}_c) - b_d(\dot{\theta}_c - \dot{\theta}_d)}{J_c}. \quad (2.36)$$

Para θ_d tenemos que:

$$\frac{\partial \delta W(q, \dot{q})}{\partial \dot{\theta}_d} - \tau_s = \frac{d}{dt} \left(\frac{\partial \mathcal{K}(q, \dot{q})}{\partial \dot{\theta}_d} \right) - \frac{\partial \mathcal{K}(q, \dot{q})}{\partial \theta_d} + \frac{\partial U(q)}{\partial \theta_d} + \frac{\partial \mathcal{D}_t(q, \dot{q})}{\partial \dot{\theta}_d}, \quad (2.37)$$

$$\frac{\partial \mathcal{K}(q, \dot{q})}{\partial \dot{\theta}_d} = J_d \dot{\theta}_d, \quad (2.38)$$

$$\frac{d}{dt} \left(\frac{\partial \mathcal{K}(q, \dot{q})}{\partial \dot{\theta}_d} \right) = J_d \ddot{\theta}_d, \quad (2.39)$$

$$\frac{\partial U(q)}{\partial \theta_d} = k_d(\theta_d - \theta_c) - k_s(\theta_s - \theta_d) \quad (2.40)$$

$$\frac{\partial \mathcal{D}_t(q, \dot{q})}{\partial \dot{\theta}_d} = -b_d(\dot{\theta}_c - \dot{\theta}_d) + b_s(\dot{\theta}_d - \dot{\theta}_s), \quad (2.41)$$

$$\frac{\partial \delta W(q, \dot{q})}{\partial \dot{\theta}_d} = 0. \quad (2.42)$$

Sustituyendo tenemos que:

$$J_d \ddot{\theta}_d + k_d(\theta_d - \theta_c) - k_s(\theta_s - \theta_d) - b_d(\dot{\theta}_c - \dot{\theta}_d) + b_s(\dot{\theta}_d - \dot{\theta}_s) = 0. \quad (2.43)$$

Despejando $\ddot{\theta}_d$ tenemos que:

$$\ddot{\theta}_d = \frac{-k_d(\theta_d - \theta_c) + k_s(\theta_s - \theta_d) + b_d(\dot{\theta}_c - \dot{\theta}_d) - b_s(\dot{\theta}_d - \dot{\theta}_s)}{J_d}. \quad (2.44)$$

Para θ_s tenemos que:

$$\frac{\partial \delta W(q, \dot{q})}{\partial \dot{\theta}_s} - \tau_s = \frac{d}{dt} \left[\frac{\partial K(q, \dot{q})}{\partial \dot{\theta}_s} \right] - \frac{\partial K(q, \dot{q})}{\partial \theta_s} + \frac{\partial U(q)}{\partial \theta_s} + \frac{\partial D_t(q, \dot{q})}{\partial \dot{\theta}_s}, \quad (2.45)$$

$$\frac{\partial K(q, \dot{q})}{\partial \dot{\theta}_s} = J_s \dot{\theta}_s, \quad (2.46)$$

$$\frac{d}{dt} \left[\frac{\partial K(q, \dot{q})}{\partial \dot{\theta}_s} \right] = J_s \ddot{\theta}_s, \quad (2.47)$$

$$\frac{\partial K(q, \dot{q})}{\partial \theta_s} = 0, \quad (2.48)$$

$$\frac{\partial U(q)}{\partial \theta_s} = k_s(\theta_s - \theta_d) + Gk_t(G\theta_s + \theta_t), \quad (2.49)$$

$$\frac{\partial Dt(q, \dot{q})}{\partial \dot{\theta}_s} = -b_s(\dot{\theta}_d - \dot{\theta}_s) - Gb_t(-\dot{\theta}_t - G\dot{\theta}_s), \quad (2.50)$$

$$\frac{\partial \delta W(q, \dot{q})}{\partial \dot{\theta}_s} = 0. \quad (2.51)$$

Sustituyendo tenemos que:

$$J_s \ddot{\theta}_s + k_s(\theta_s - \theta_d) + Gk_t(G\theta_s + \theta_t) - b_s(\dot{\theta}_d - \dot{\theta}_s) - Gb_t(-\dot{\theta}_t - G\dot{\theta}_s) = 0. \quad (2.52)$$

Despejando $\ddot{\theta}_s$ tenemos que:

$$\ddot{\theta}_s = \frac{-k_s(\theta_s - \theta_d) - Gk_t(G\theta_s + \theta_t) + b_s(\dot{\theta}_d - \dot{\theta}_s) + Gb_t(-\dot{\theta}_t - G\dot{\theta}_s)}{J_s}. \quad (2.53)$$

Para θ_t tenemos que:

$$\frac{\partial \delta W(q, \dot{q})}{\partial \dot{\theta}_t} - \tau_s = \frac{d}{dt} \left[\frac{\partial K(q, \dot{q})}{\partial \dot{\theta}_t} \right] - \frac{\partial K(q, \dot{q})}{\partial \theta_t} + \frac{\partial U(q)}{\partial \theta_t} + \frac{\partial Dt(q, \dot{q})}{\partial \dot{\theta}_t}, \quad (2.54)$$

$$\frac{\partial K(q, \dot{q})}{\partial \dot{\theta}_t} = J_t \dot{\theta}_t, \quad (2.55)$$

$$\frac{d}{dt} \left[\frac{\partial K(q, \dot{q})}{\partial \dot{\theta}_t} \right] = J_t \ddot{\theta}_t, \quad (2.56)$$

$$\frac{\partial K(q, \dot{q})}{\partial \theta_t} = 0, \quad (2.57)$$

$$\frac{\partial U(q)}{\partial \theta_t} = k_t(G\theta_s + \theta_t) + k_p(\theta_t - \theta_p), \quad (2.58)$$

$$\frac{\partial Dt(q, \dot{q})}{\partial \dot{\theta}_t} = -b_t(-\dot{\theta}_t - G\dot{\theta}_s) - b_p(\dot{\theta}_p - \dot{\theta}_t), \quad (2.59)$$

$$\frac{\partial \delta W(q, \dot{q})}{\partial \dot{\theta}_t} = 0. \quad (2.60)$$

Sustituyendo tenemos que:

$$J_t \ddot{\theta}_t + k_t(G\theta_s + \theta_t) + k_p(\theta_t - \theta_p) - b_t(-\dot{\theta}_t - G\dot{\theta}_s) - b_p(\dot{\theta}_p - \dot{\theta}_t) = 0. \quad (2.61)$$

Despejando $\ddot{\theta}_t$ tenemos que:

$$\ddot{\theta}_t = \frac{-k_t(G\theta_s + \theta_t) - k_p(\theta_t - \theta_p) + b_t(-\dot{\theta}_t - G\dot{\theta}_s) + b_p(\dot{\theta}_p - \dot{\theta}_t)}{J_t}. \quad (2.62)$$

Para θ_p tenemos que:

$$\frac{\partial \delta W(q, \dot{q})}{\partial \dot{\theta}_p} - \tau_s = \frac{d}{dt} \left[\frac{\partial K(q, \dot{q})}{\partial \dot{\theta}_p} \right] - \frac{\partial K(q, \dot{q})}{\partial \theta_p} + \frac{\partial U(q)}{\partial \theta_p} + \frac{\partial Dt(q, \dot{q})}{\partial \dot{\theta}_p}, \quad (2.63)$$

$$\frac{\partial K(q, \dot{q})}{\partial \dot{\theta}_p} = J_p \dot{\theta}_p, \quad (2.64)$$

$$\frac{d}{dt} \left[\frac{\partial K(q, \dot{q})}{\partial \dot{\theta}_p} \right] = J_p \ddot{\theta}_p, \quad (2.65)$$

$$\frac{\partial K(q, \dot{q})}{\partial \theta_p} = 0, \quad (2.66)$$

$$\frac{\partial U(q)}{\partial \theta_p} = -k_p(\theta_t - \theta_p) - Rk_b(x_c - R\theta_p), \quad (2.67)$$

$$\frac{\partial Dt(q, \dot{q})}{\partial \dot{\theta}_p} = b_p(\dot{\theta}_p - \dot{\theta}_t) - Rb_b(\dot{x}_c - R\dot{\theta}_p), \quad (2.68)$$

$$\frac{\partial \delta W(q, \dot{q})}{\partial \dot{\theta}_p} = 0. \quad (2.69)$$

Sustituyendo tenemos que:

$$J_p \ddot{\theta}_p - k_p(\theta_t - \theta_p) - Rk_b(x_c - R\theta_p) + b_p(\dot{\theta}_p - \dot{\theta}_t) - Rb_b(\dot{x}_c - R\dot{\theta}_p) = 0. \quad (2.70)$$

Despejando $\ddot{\theta}_p$ tenemos que:

$$\ddot{\theta}_p = \frac{k_p(\theta_t - \theta_p) + Rk_b(x_c - R\theta_p) - b_p(\dot{\theta}_p - \dot{\theta}_t) + Rb_b(\dot{x}_c - R\dot{\theta}_p)}{J_p}. \quad (2.71)$$

Para x_c tenemos que:

$$\frac{\partial \delta W(q, \dot{q})}{\partial \dot{x}_c} - \tau_s = \frac{d}{dt} \left[\frac{\partial K(q, \dot{q})}{\partial \dot{x}_c} \right] - \frac{\partial K(q, \dot{q})}{\partial x_c} + \frac{\partial U(q)}{\partial x_c} + \frac{\partial Dt(q, \dot{q})}{\partial \dot{x}_c}, \quad (2.72)$$

$$\frac{\partial K(q, \dot{q})}{\partial \dot{x}_c} = (M + m)\dot{x}_c + ml\dot{q}_p \cos q_p, \quad (2.73)$$

$$\frac{d}{dt} \left[\frac{\partial K(q, \dot{q})}{\partial \dot{x}_c} \right] = (M + m)\ddot{x}_c + ml\ddot{q}_p \cos q_p - ml\dot{q}_p^2 \sin q_p, \quad (2.74)$$

$$\frac{\partial K(q, \dot{q})}{\partial x_c} = 0, \quad (2.75)$$

$$\frac{\partial U(q)}{\partial x_c} = k_b(x_c - R\theta_p), \quad (2.76)$$

$$\frac{\partial Dt(q, \dot{q})}{\partial \dot{x}_c} = b_b(\dot{x}_c - R\dot{\theta}_p), \quad (2.77)$$

$$\frac{\partial \delta W(q, \dot{q})}{\partial \dot{x}_c} = 0. \quad (2.78)$$

Sustituyendo tenemos que:

$$(M + m)\ddot{x}_c + ml\ddot{q}_p \cos q_p - ml\dot{q}_p^2 \sin q_p + k_b(x_c - R\theta_p) + b_b(\dot{x}_c - R\dot{\theta}_p) = 0. \quad (2.79)$$

Despejando $\ddot{x}_c$ tenemos que:

$$\ddot{x}_c = \frac{-ml\ddot{q}_p \cos q_p + ml\dot{q}_p^2 \sin q_p - k_b(x_c - R\theta_p) - b_b(\dot{x}_c - R\dot{\theta}_p)}{M + m}. \quad (2.80)$$

Para q_p tenemos que:

$$\frac{\partial \delta W(q, \dot{q})}{\partial \dot{q}_p} - \tau_s = \frac{d}{dt} \left[\frac{\partial K(q, \dot{q})}{\partial \dot{q}_p} \right] - \frac{\partial K(q, \dot{q})}{\partial q_p} + \frac{\partial U(q)}{\partial q_p} + \frac{\partial Dt(q, \dot{q})}{\partial \dot{q}_p}, \quad (2.81)$$

$$\frac{\partial K(q, \dot{q})}{\partial \dot{q}_p} = ml\dot{x}_c \cos q_p + ml^2\dot{q}_p, \quad (2.82)$$

$$\frac{d}{dt} \left[\frac{\partial K(q, \dot{q})}{\partial \dot{q}_p} \right] = ml\ddot{x}_c \cos q_p - ml\dot{x}_c \dot{q}_p \sin q_p + ml^2\ddot{q}_p, \quad (2.83)$$

$$\frac{\partial K(q, \dot{q})}{\partial q_p} = -ml\dot{x}_c \dot{q}_p \sin q_p, \quad (2.84)$$

$$\frac{\partial U(q)}{\partial q_p} = mgl \sin q_p, \quad (2.85)$$

$$\frac{\partial Dt(q, \dot{q})}{\partial \dot{q}_p} = b_q \dot{q}_p, \quad (2.86)$$

$$\frac{\partial \delta W(q, \dot{q})}{\partial \dot{q}_p} = 0, \quad (2.87)$$

Sustituyendo tenemos que:

$$ml\ddot{x}_c \cos q_p - ml\dot{x}_c \dot{q}_p \sin q_p + ml^2\ddot{q}_p + ml\dot{x}_c \dot{q}_p \sin q_p + mgl \sin q_p + b_q \dot{q}_p = 0, \quad (2.88)$$

$$ml\ddot{x}_c \cos q_p + ml^2\ddot{q}_p + mgl \sin q_p + b_q \dot{q}_p = 0. \quad (2.89)$$

Despejando $\ddot{q}_p$ tenemos que:

$$\ddot{q}_p = \frac{-ml\ddot{x}_c \cos q_p - mgl \sin q_p - b_q \dot{q}_p}{ml^2}. \quad (2.90)$$

Con las ecuaciones anteriores, se procede a obtener su representación en variables de estado, para esto primero asignamos las variables adecuadas al sistema como se presenta en la Tabla 2.7.

Variable del sistema	Variable de estado asociada
θ_m	x_1
$\dot{\theta}_m$	$\dot{x}_1 = x_2$
$\ddot{\theta}_m$	$\dot{x}_2$

θ_c	x_3
$\dot{\theta}_c$	$\dot{x}_3 = x_4$
$\ddot{\theta}_c$	$\dot{x}_4$
θ_d	x_5
$\dot{\theta}_d$	$\dot{x}_5 = x_6$
$\ddot{\theta}_d$	$\dot{x}_6$
θ_s	x_7
$\dot{\theta}_s$	$\dot{x}_7 = x_8$
$\ddot{\theta}_s$	$\dot{x}_8$
θ_t	x_9
$\dot{\theta}_t$	$\dot{x}_9 = x_{10}$
$\ddot{\theta}_t$	$\dot{x}_{10}$
θ_p	x_{11}
$\dot{\theta}_p$	$\dot{x}_{11} = x_{12}$
$\ddot{\theta}_p$	$\dot{x}_{12}$
x_c	x_{13}
$\dot{x}_c$	$\dot{x}_{13} = x_{14}$
$\ddot{x}_c$	$\dot{x}_{14}$
q_p	x_{15}
$\dot{q}_p$	$\dot{x}_{15} = x_{16}$
$\ddot{q}_p$	$\dot{x}_{16}$

Tabla 2.7: Asignación de variables de estado.

Tomando estas variables se procede a realizar la sustitución, quedando de la siguiente manera:

$$\dot{x}_1 = x_2, \quad (2.91)$$

$$\dot{x}_2 = \frac{\tau_m - k_m x_1 + k_c(x_3 - x_1) - b_m x_2 - b_c(x_2 - x_4)}{J_m}, \quad (2.92)$$

$$\dot{x}_3 = x_4, \quad (2.93)$$

$$\dot{x}_4 = \frac{-k_c(x_3 - x_1) + k_d(x_5 - x_3) + b_c(x_2 - x_4) - b_d(x_4 - x_6)}{J_c}, \quad (2.94)$$

$$\dot{x}_5 = x_6, \quad (2.95)$$

$$\dot{x}_6 = \frac{-k_d(x_5 - x_3) + k_s(x_7 - x_5) + b_d(x_4 - x_6) - b_s(x_6 - x_8)}{J_d}, \quad (2.96)$$

$$\dot{x}_7 = x_8, \quad (2.97)$$

$$\dot{x}_8 = \frac{-k_s(x_7 - x_5) - Gk_t(Gx_7 + x_9) + b_s(x_6 - x_8) + Gb_t(-x_{10} - Gx_8)}{J_s}, \quad (2.98)$$

$$\dot{x}_9 = x_{10}, \quad (2.99)$$

$$\dot{x}_{10} = \frac{-k_t(Gx_7 + x_9) - k_p(x_9 - x_{11}) + b_t(-x_{10} - Gx_8) + b_p(x_{12} - x_{10})}{J_t}, \quad (2.100)$$

$$\dot{x}_{11} = x_{12}, \quad (2.101)$$

$$\dot{x}_{12} = \frac{k_p(x_9 - x_{11}) + Rk_b(x_{13} - Rx_{11}) - b_p(x_{12} - x_{10}) + Rb_b(x_{14} - Rx_{12})}{J_p}, \quad (2.102)$$

$$\dot{x}_{13} = x_{14}, \quad (2.103)$$

$$\dot{x}_{14} = \frac{-ml\dot{x}_{16} \cos x_{15} + mlx_{16}^2 \sin x_{15} - k_b(x_{13} - Rx_{11}) - b_b(x_{14} - Rx_{12})}{M + m}, \quad (2.104)$$

$$\dot{x}_{15} = x_{16}, \quad (2.105)$$

$$\dot{x}_{16} = \frac{-ml\dot{x}_{14} \cos x_{15} - mlg \sin x_{15} - b_q x_{16}}{ml^2}. \quad (2.106)$$

Con las ecuaciones obtenidas y su representación en variables de estado, es posible desarrollar un algoritmo en algún lenguaje de programación en el cual se implementen estas ecuaciones y así obtengamos las gráficas de respuesta de cada elemento modelado, para nosotros el software seleccionado es Matlab y Simulink. Esta implementación se realiza planteando las ecuaciones como un sistema ordinario de ecuaciones diferenciales ODE y el método de solución utilizado para esta ODE es el método numérico Runge-Kutta 4/5.

Para poder simular el sistema como antes mencionamos, es necesario conocer los parámetros involucrados, para esto se realizaron mediciones necesarias al robot para obtener de manera teórica los valores de estos parámetros, a continuación, se presentan como se obtuvieron algunos de estos parámetros y los valores que se consideraron para la simulación.

Para el caso de las inercias de cada elemento se sabe que, para cilindros sólidos, el momento de inercia sobre su eje está dado como [33]:



$$J_n = \frac{1}{2}mR^2, \quad (2.107)$$

donde:

- m es la masa
- R el radio del cilindro

Figura 2.5: Diagrama de un eje.

Aplicando esta fórmula se obtienen los momentos de inercia para:

El cople, el cual tiene una masa de 0.033 Kg y un radio de 0.0125 m, al aplicar la ecuación (2.107) se tiene que su inercia es:

$$J_c = \frac{1}{2}mR^2 = \frac{1}{2}(0.033)(0.0125)^2 = 2.58 \times 10^{-6} \text{ kg}\cdot\text{m}^2. \quad (2.108)$$

En el caso del eje de salida del cople se tiene que:

$$J_d = \frac{1}{2}mR^2 = \frac{1}{2}(0.149)(0.01)^2 = 7.45 \times 10^{-6} \text{ kg}\cdot\text{m}^2. \quad (2.109)$$

En el caso del tornillo sinfín y la corona estos se obtuvieron utilizando un software de diseño, el cual fue SolidWorks, para obtener las inercias más aproximadas debido a su geometría, estos datos se encuentran en la Tabla 2.8.

Otro parámetro que se pudo obtener de forma teórica fueron algunas de las constantes de rigidez, para los cuales se utilizó la siguiente fórmula: [34]

$$K_n = \frac{GJ_p}{L}, \quad (2.110)$$

donde:

- G es el módulo de elasticidad del material.
- J_p es el momento polar de inercia para un cilindro solido de radio r el cual está dado por la expresión:

$$J_p = \frac{\pi d^4}{32}, \quad (2.111)$$

- L Es la longitud del cilindro.

Los valores obtenidos para las constantes de rigidez se encuentran en la tabla 2.8, en el caso de la constante de rigidez de la banda no se puede obtener de manera teórica ya que se requiere de unas pruebas en las cuales se puede dañar o destruir el elemento, por tal motivo en algunas se consideró un valor estimado de las hojas de datos.

Otro de los parámetros que se deben definir son los coeficientes de fricción, estos se definieron después de realizar algunas pruebas experimentales que se utilizaron como referencia para su sintonía.

Tabla 2.8: Valores de los parámetros

Parámetro	Valor	Unidades	Parámetro	Valor	Unidades
b_m	0.004	$\frac{\text{Kg}\cdot\text{m}^2}{\text{rad}\cdot\text{s}}$	K_t	125941.99	$\frac{\text{Kg}\cdot\text{m}^2}{\text{rad}\cdot\text{s}^2}$
J_m	2.61×10^{-4}	$\frac{\text{Kg}\cdot\text{m}^2}{\text{rad}}$	b_t	0.004	$\frac{\text{Kg}\cdot\text{m}^2}{\text{rad}\cdot\text{s}}$
K_c	11620.45	$\frac{\text{Kg}\cdot\text{m}^2}{\text{rad}\cdot\text{s}^2}$	J_t	1.372×10^{-4}	$\frac{\text{Kg}\cdot\text{m}^2}{\text{rad}}$
b_c	0.004	$\frac{\text{Kg}\cdot\text{m}^2}{\text{rad}\cdot\text{s}}$	K_p	29900108.0805	$\frac{\text{Kg}\cdot\text{m}^2}{\text{rad}\cdot\text{s}^2}$
J_c	2.72×10^{-4}	$\frac{\text{Kg}\cdot\text{m}^2}{\text{rad}}$	b_p	0.004	$\frac{\text{Kg}\cdot\text{m}^2}{\text{rad}\cdot\text{s}}$
K_d	8260.5784	$\frac{\text{Kg}\cdot\text{m}^2}{\text{rad}\cdot\text{s}^2}$	J_p	3.305×10^{-4}	$\frac{\text{Kg}\cdot\text{m}^2}{\text{rad}}$
b_d	0.004	$\frac{\text{Kg}\cdot\text{m}^2}{\text{rad}\cdot\text{s}}$	K_b	100000	$\frac{\text{Kg}}{\text{s}^2}$
J_d	6.67×10^{-4}	$\frac{\text{Kg}\cdot\text{m}^2}{\text{rad}}$	b_b	0.4	$\frac{\text{Kg}}{\text{s}}$
K_s	802821.13	$\frac{\text{Kg}\cdot\text{m}^2}{\text{rad}\cdot\text{s}^2}$	M	1.53	Kg
b_s	0.02	$\frac{\text{Kg}\cdot\text{m}^2}{\text{rad}\cdot\text{s}}$	m	0.44	Kg
J_s	1.32×10^{-4}	$\frac{\text{Kg}\cdot\text{m}^2}{\text{rad}}$	l	0.36	m
G	$\frac{3}{31}$	—	g	9.81	m/s^2
			b_q	0.032	$\frac{\text{Kg}\cdot\text{m}^2}{\text{s}}$

Una vez ya definidos estos valores se prosiguió a realizar la simulación del sistema en Matlab siguiendo el diagrama de la Figura 2.6.

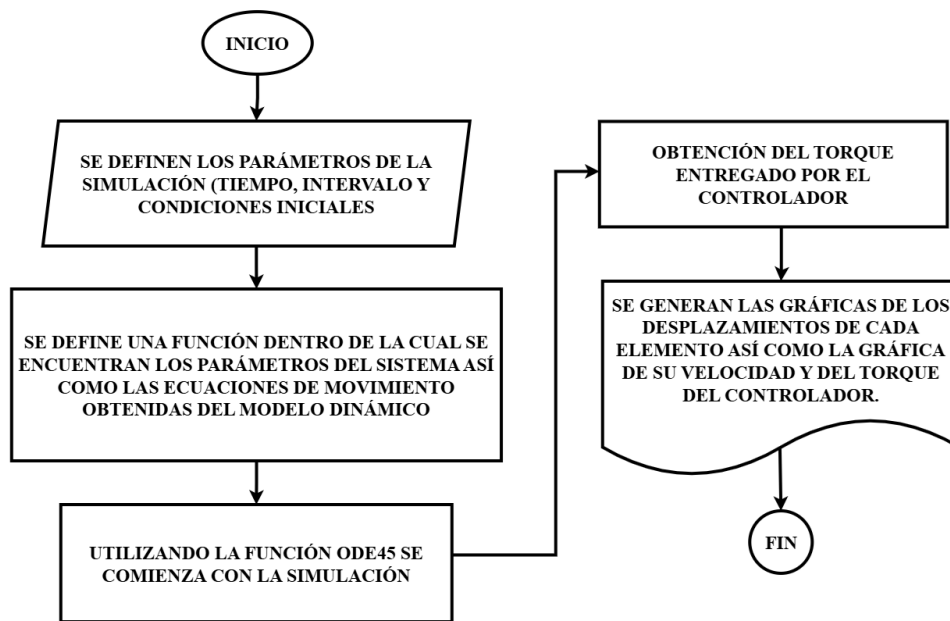


Figura 2.6: Diagrama de flujo de la simulación.

Para comenzar con la simulación se definió una estrategia de control, la cual se dividió en 3 etapas, como se muestra en el diagrama de la Figura 2.7.

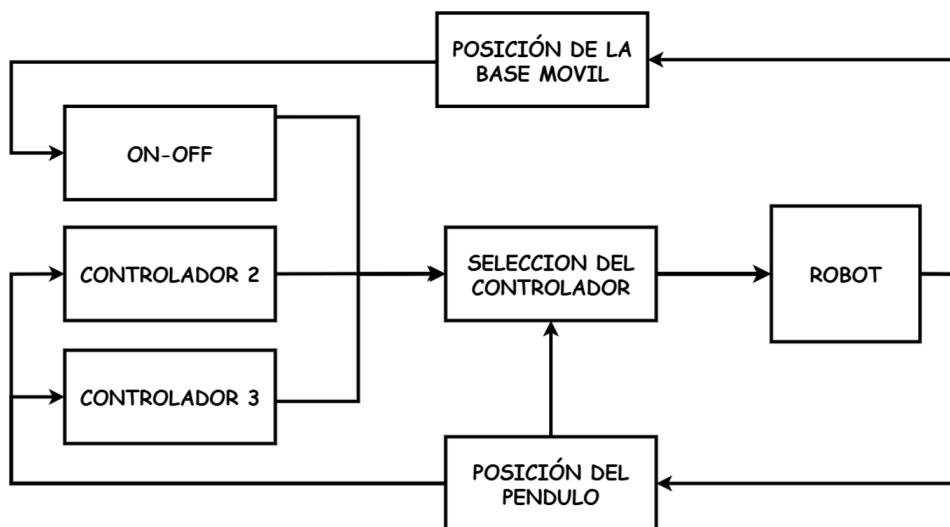


Figura 2.7: Diagrama del sistema de control.

La primera etapa consiste en una señal del tipo ON-OFF, en donde su función es la de oscilar al péndulo para llevarlo de su posición inicial de 0° a la zona del segundo controlador, este controlador depende de la posición de la base móvil, donde se define que la base móvil debe oscilar entre el rango de 0.14 m y 0.38 m, para lograr elevar al péndulo a la posición superior.

Para el segundo controlador tenemos que su rango de operación se encuentra entre $(145^\circ, 165^\circ)$ y $(-145^\circ, -165^\circ)$, este controlador tiene la función de frenar al péndulo antes

de llegar a la zona del tercer controlador, este depende de la posición del péndulo, teniendo la siguiente estructura.

$$\tau = K_{p_1} \tanh(e), \quad (2.112)$$

en donde, τ es el torque del controlador, K_{p_1} es la ganancia proporcional y e es el error de posición definido como $e = q_d - q_p$.

En nuestro caso se definió que $K_{p_1}=0.95$ Nm, para que al utilizar la función de $\tanh$ este controlador no sature al motor y solo le pida el máximo del 25 % del torque disponible.

Controlador P

Ahora en el caso del tercer controlador, el cual se encarga de estabilizar al péndulo en la posición vertical superior a $\pm 180^\circ$, se definieron dos controladores a implementar, siendo el primero un control proporcional de la forma:

$$\tau = K_{p_2} \tanh(-e), \quad (2.113)$$

en donde, τ es el torque del controlador, K_{p_2} es la ganancia proporcional y e es el error de posición definido como $e = q_d - q_p$.

Para este caso se sintonizo la ganancia proporcional, obteniendo que $K_{p_2} = 0.3$, lo que nos indica que el torque máximo entregado por el controlador para esta prueba fuera solo del 30 % del torque total de nuestro motor, con ello se prosiguió a realizar la simulación obteniendo los siguientes resultados.

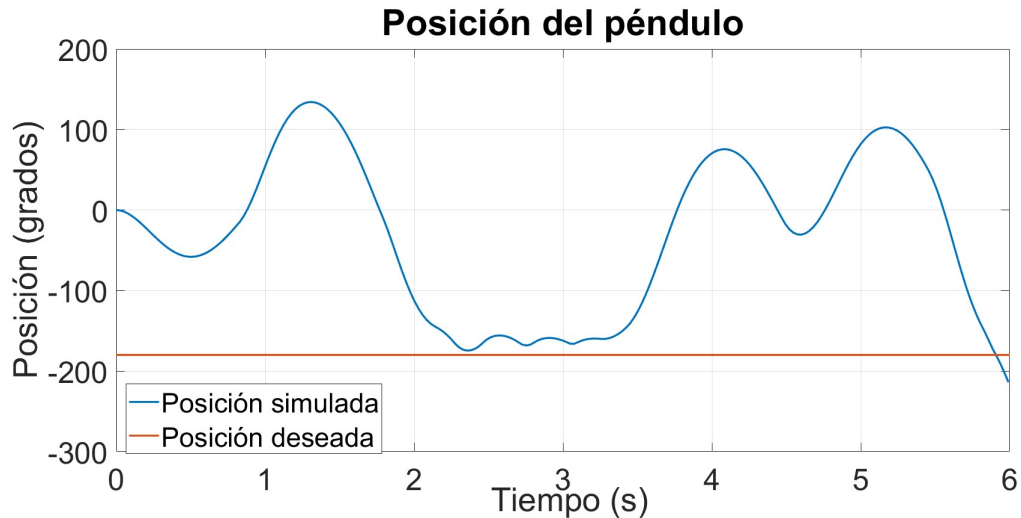


Figura 2.8: Gráfica de posición del péndulo con el controlador proporcional.

En la Figura 2.8, se puede observar que el algoritmo de control oscila inicialmente al péndulo y en 3 oscilaciones, logra subir al péndulo a la zona del segundo y tercer controlador hasta llevarlo a la posición deseada de -180° , la posición la logra mantener

por un momento debido a que la base móvil llega a su límite físico y por ello se cae el péndulo nuevamente.

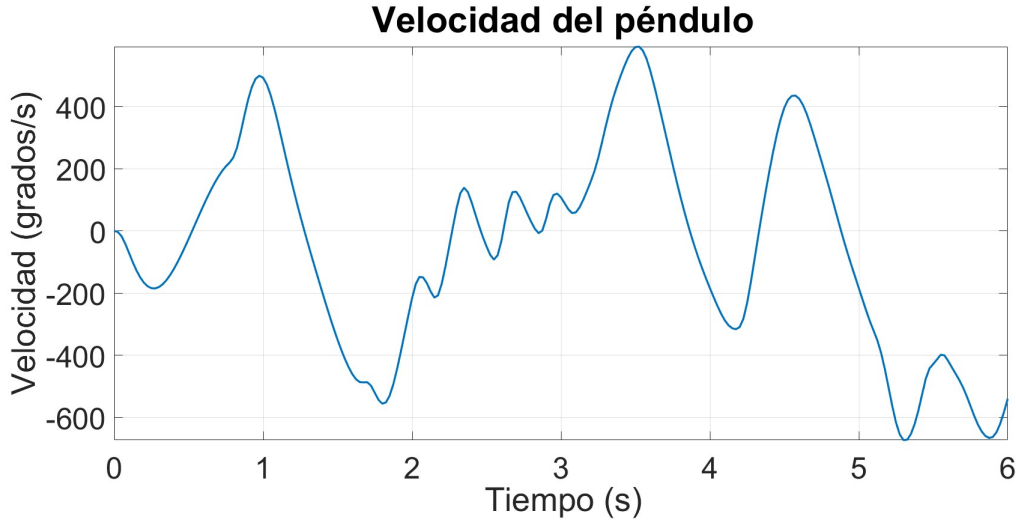


Figura 2.9: Gráfica de velocidad del péndulo con el controlador proporcional.

En la Figura 2.9 se puede observar la velocidad que toma el péndulo durante su movimiento nótese como en los cambios de dirección de la base móvil se generan picos debido a que en ese momento la energía que recibe el péndulo es alta.

Controlador PD

Ahora para el segundo controlador, se utilizó un PD de la forma:

$$\tau = K_{p_3} \tanh(-e_p) + K_v \tanh(e_v), \quad (2.114)$$

en donde, τ es el torque del controlador, K_{p_3} es la ganancia proporcional, e_p es el error de posición definido como $e_p = q_d - q_p$, K_v es la ganancia derivativa y e_v es el error de posición definido como $e_v = v_d - v_p$.

Para este caso se sintonizo nuevamente la ganancia proporcional y ahora también la derivativa, obteniendo que $K_{p_3} = 0.3$ y $K_v = 0.05$, lo que nos indica que el torque máximo entregado por el controlador para esta prueba fuera solo del 35 % del torque total de nuestro motor, con ello se prosiguió a realizar la simulación obteniendo los siguientes resultados.

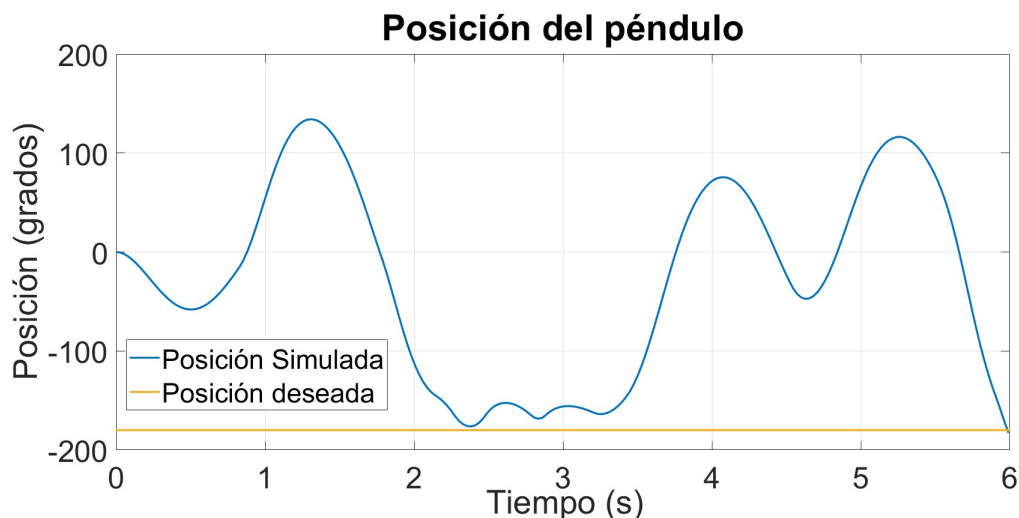


Figura 2.10: Gráfica de posición del péndulo con el controlador PD.

En la Figura 2.10, se puede observar que el algoritmo de control oscila inicialmente al péndulo y en 3 oscilaciones, logra subir al péndulo a la zona del segundo y tercer controlador hasta llevarlo a la posición deseada de -180° , con este controlador se observa una pequeña mejora en cuanto a que el movimiento en la zona de estabilización es ligeramente más suave, de igual forma la posición la logra mantener por un momento debido a que la base móvil llega a su límite físico y por ello se cae el péndulo nuevamente.

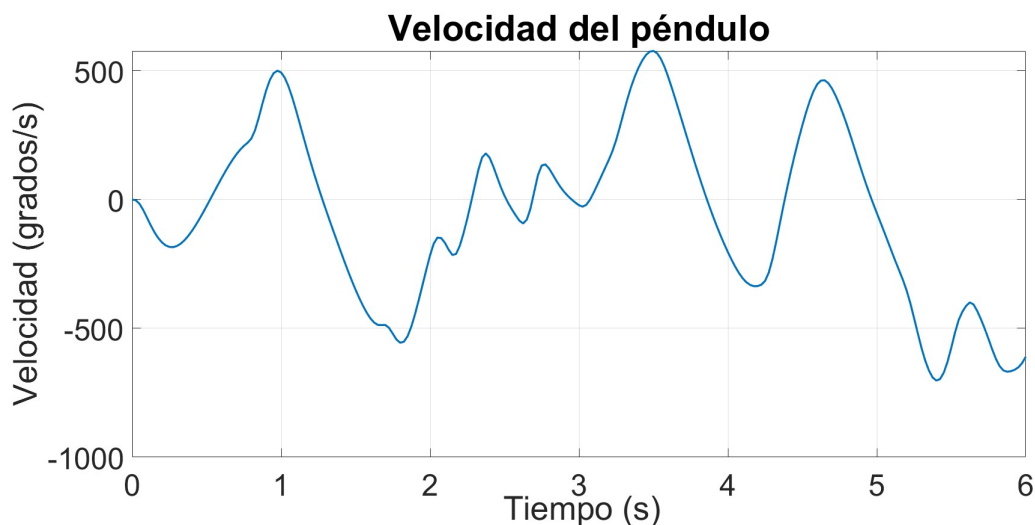


Figura 2.11: Gráfica de velocidad del péndulo con el controlador proporcional.

En la Figura 2.11 se puede observar la velocidad que toma el péndulo durante su movimiento en este caso se observa nuevamente como en los cambios de dirección de la base móvil se generan picos esto debido como comentábamos antes a que en ese momento la energía que recibe el péndulo es alta.

2.4. Conclusión

El desarrollo del modelo dinámico del péndulo invertido sobre base móvil permitió comprender a profundidad el comportamiento físico del sistema, proporcionando una base matemática sólida para el diseño y la implementación de estrategias de control. A través de la formulación de ecuaciones de movimiento de Euler-Lagrange, se logró describir con precisión las interacciones entre el carro y el péndulo, considerando factores como la fricción y las características mecánicas del sistema.

Este modelo dinámico es esencial para la predicción del comportamiento del sistema bajo diferentes condiciones, siendo particularmente relevante para garantizar la estabilidad del péndulo en su posición vertical. Además, permitió determinar la relación entre las variables de entrada, como el par aplicado por el motor, y las variables de salida, como las posiciones y velocidades del péndulo y la base móvil.

La implementación de este modelo en simulaciones y su validación experimental garantizaron que las aproximaciones teóricas sean representativas de la realidad, estableciendo un puente confiable entre la teoría y la práctica. En conclusión, el modelo dinámico desarrollado es una herramienta fundamental para el diseño de controladores efectivos, contribuyendo al desempeño robusto y eficiente del sistema en tareas complejas como la estabilización del péndulo y su operación autónoma.

Capítulo 3

Electrónica del robot

En este capítulo se abordarán los cambios realizados a la electrónica del robot, comenzando con la etapa de potencia, en donde se realiza la caracterización del motor y con los datos obtenidos se prosigue con el diseño del puente H para su activación y la selección de la fuente de alimentación, seguimos con la electrónica para la adquisición de señales, en donde tenemos el acoplamiento de las señales de los encoders y los fines de carrera, por último se presenta la integración de toda la electrónica en una estructura que facilite su transporte.

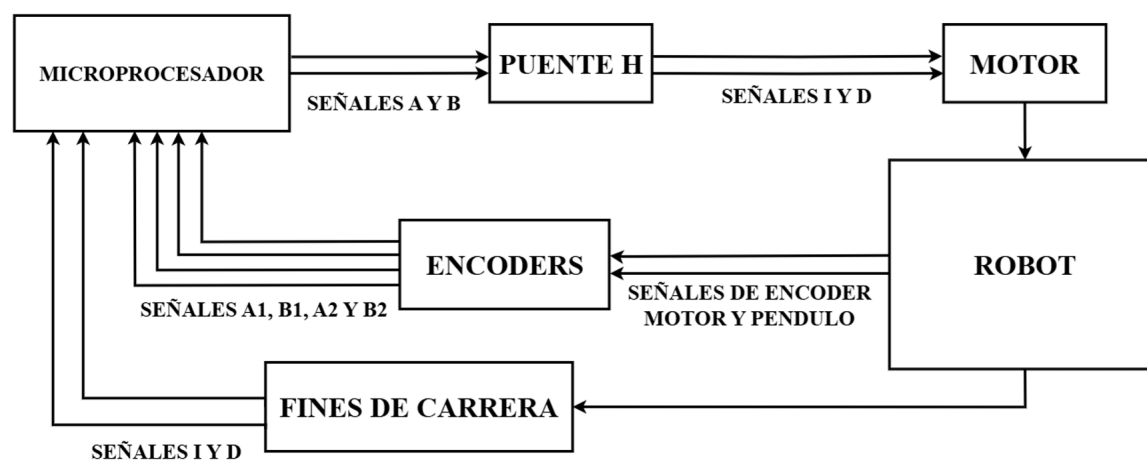


Figura 3.1: Diagrama de la electrónica del robot.

3.1. Electrónica de potencia

Comenzando con la electrónica de potencia, en donde se realizará la caracterización del motor, esto para obtener nuevamente las condiciones en las que su comportamiento se asemeje a un motor de transmisión directa, el proceso de caracterización se describe a continuación.

3.1.1. Caracterización del motor

La caracterización de un motor de corriente directa se realiza para conocer las condiciones en las que un motor DC se comporta como uno de transmisión directa, para esto se requiere de los siguientes elementos.

Motor DC

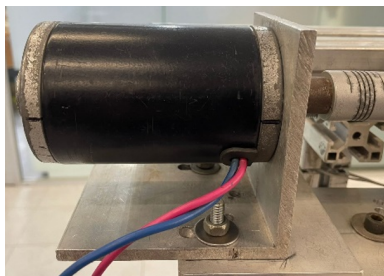


Figura 3.2: Motor DC.

Un dinamómetro, este se utiliza para conocer el torque entregado por el motor, al momento de alimentarlo y frenarlo.



Figura 3.3: Dinamómetro.

Un freno prony, el cual se utiliza como base donde se monta el motor con un sproket y una cadena a una rueda de mayor diámetro conocido, con el fin de reducir el torque de salida para no dañar el sensor de medición.



Figura 3.4: Freno Prony.

Una fuente de DC variable, con la cual se alimenta el motor o el circuito de conmutación durante las pruebas.



Figura 3.5: Fuente DC variable.

Un generador de funciones, esta se utiliza para generar la señal de PWM que activara el circuito de conmutación durante las pruebas.



Figura 3.6: Generador de funciones.

Un circuito conmutador, este se utiliza para aplicar una señal de alimentación en forma de PWM al motor, es necesario ya que la señal no tiene las características para encender de manera adecuada el motor de manera directa, por lo cual se utiliza una fuente de DC capaz de entregar al motor la energía necesaria conectada a través de este circuito conmutador.

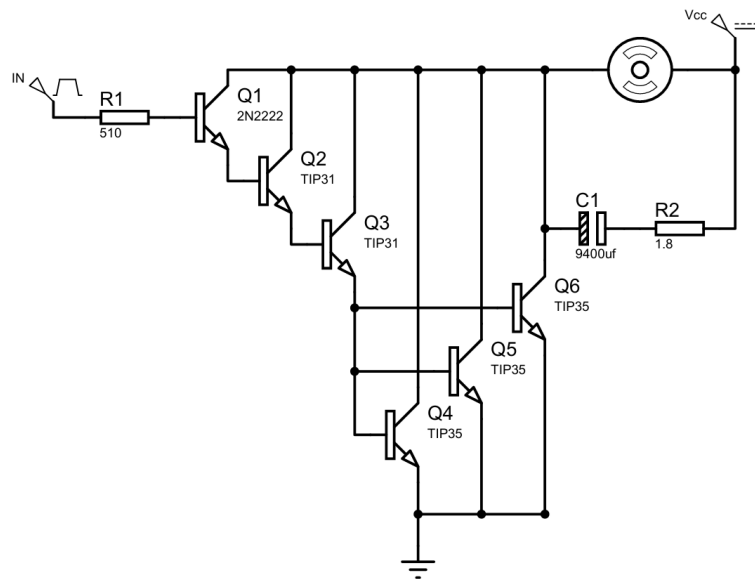


Figura 3.7: Circuito de conmutación.

Se sabe que el motor es de 24 V a 20 A, por lo tanto, se utilizan estos datos para diseñar el circuito de conmutación presentado anteriormente, en donde sabiendo la corriente nominal se tiene que:

Utilizando una configuración en Darlington de transistores NPN, como son dos 2n2222, un TIP31C y un TIP35C, los cuales poseen una beta promedio de 100, 40 y 15 respectivamente, entonces:

$$\beta_f = \frac{100 \cdot 25 \cdot 25 \cdot 15}{10} = 93750, \quad (3.1)$$

Si la corriente de entrada es de 1mA, entonces:

$$I_C = \beta_f I_B, \quad (3.2)$$

$$I_C = 93750 \cdot 0.001A = 93.75A. \quad (3.3)$$

Esto es más que suficiente para garantizar que la corriente que se le entregue al motor sea la solicitada, ahora es necesario obtener el valor de la resistencia de entrada, la cual se obtiene a continuación:

$$V_{IN} - I_B R_B - 0.7V - 0.7V - 0.7V - 0.7V = 0V, \quad (3.4)$$

$$I_B R_B = V_{IN} - 2.8V, \quad (3.5)$$

$$R_B = \frac{V_{IN} - 2.8V}{I_B} = \frac{3.3V - 2.8V}{1 \text{ mA}} = 500 \Omega \approx 510 \Omega. \quad (3.6)$$

Una vez que ya disponemos de estos elementos se realiza lo siguiente.

1. Primero se debe determinar el voltaje nominal, esto se puede conocer al medir la corriente y el torque al variar su voltaje, hasta encontrar el voltaje donde no aumenta la corriente ni el torque. El torque se mide utilizando el freno prony.

Para esta prueba la configuración utilizada consta de una fuente de DC variable, el dinamómetro, el freno prony y el motor, colocados de la siguiente manera.

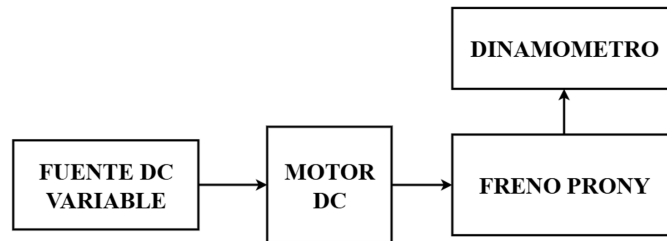


Figura 3.8: Diagrama de la primera prueba.

Una vez conectados estos elementos, se realiza el experimento obteniendo los siguientes datos:

Voltaje(V)	Corriente(A)	Torque(Nm)
1	1.1	0.07848
2	1.96	0.329616
3	3.49	0.62784
4	4.99	0.816192
5	6.22	1.02024
6	7.6	1.239984
7	8.07	1.5696
8	10.5	1.80504
9	11.58	1.946304
10	13.83	2.213136
11	15.1	2.370096
12	17.15	2.621232
13	17.99	2.762496
14	19.13	2.935152
15	20.4	3.013632
16	22.14	3.1392
17	22.34	3.45312
18	22.81	3.704256
19	25.06	3.76704
20	25.08	3.829824

Tabla 3.1: Obtención del voltaje nominal.

- Lo segundo es aplicar un barrido de frecuencia de una señal PWM con un ciclo de trabajo del 50 % para encontrar la frecuencia en la cual se consigue la mitad del torque máximo encontrado en el punto anterior.

Para esta prueba la configuración utilizada consta de una fuente de DC variable que alimenta al circuito de conmutación, un generador de funciones que activa el circuito de conmutación, el dinamómetro, el freno prony y el motor, colocados de la siguiente manera:

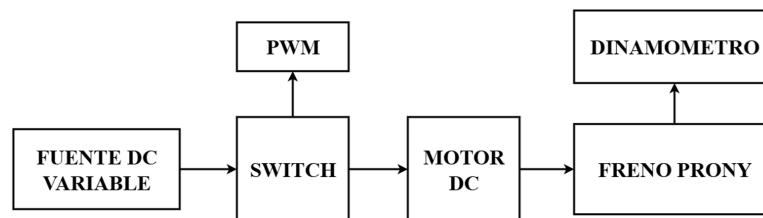


Figura 3.9: Diagrama de la segunda prueba.

Una vez conectados estos elementos se procede a realizar el experimento obteniendo los siguientes resultados.

Frecuencia(Hz)	Corriente(A)	Torque(Nm)
50	1.1	0.07848
100	1.96	0.329616
200	4.99	0.816192
500	7.6	1.239984
1000	10.5	1.80504
2000	13.83	2.213136
5000	17.15	2.621232
10000	19.13	2.935152
20000	22.14	3.1392
28000	22.34	3.45312
30000	22.81	3.704256
50000	25.06	3.76704
100000	25.08	3.829824

Tabla 3.2: Obtención del voltaje nominal.

3. El siguiente paso consiste en seleccionar las frecuencias o frecuencia que proporcione el valor que sea más cercano a la mitad del torque máximo que se calculó en el paso 1, para aplicarle un barrido del ciclo de trabajo del PWM del 10 % al 90 %, se observa que en la tabla anterior las frecuencias que se encuentran cercanas a la mitad del torque máximo son 50 Hz y 28 kHz, debido a que al introducir la señal de 50 Hz el motor vibra mucho se opta por utilizar la frecuencia de 28 kHz, con lo cual al aplicar el barrido del ciclo de trabajo se obtiene la siguiente grafica.

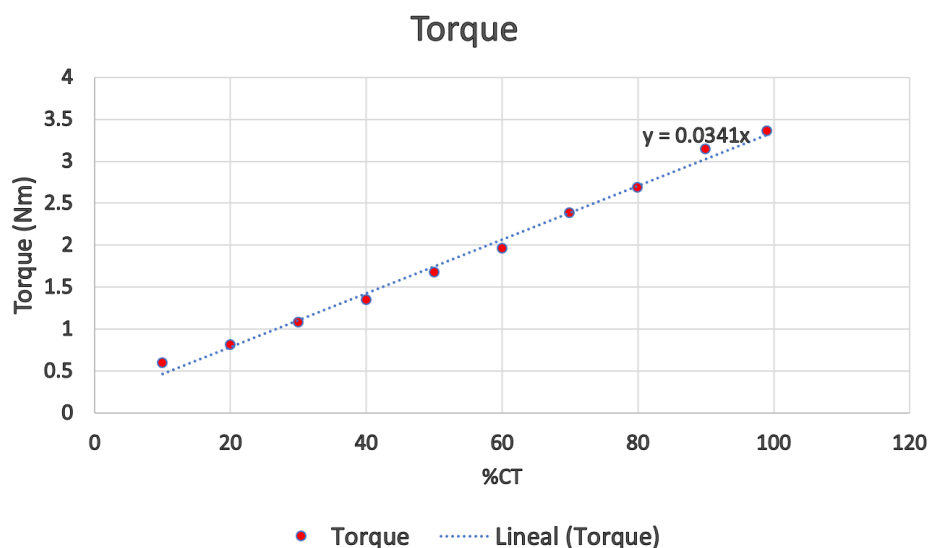


Figura 3.10: Gráfica de respuesta del torque entregado a 28kHz.

4. Al final, se deben revisar las gráficas obtenidas y analizarlas para seleccionar la frecuencia que proporcione la respuesta más cercana posible a un comportamiento lineal para el motor.

Como se pudo observar en la gráfica obtenida en el paso 3, la señal de PWM a 28kHz nos proporciona una respuesta muy cercana a un comportamiento lineal del motor, por lo cual podemos concluir que el experimento nos logró proporcionar un comportamiento muy cercano al de un motor de transmisión directa tal y como se esperaba.

La ecuación linealizada del troque obtenida fue:

$$\tau = 0.0341 \%CT. \quad (3.7)$$

3.1.2. Fuente de alimentación

una vez realizada la caracterización del motor, seleccionamos una fuente conmutada que logre suministrar la energía necesaria al motor para su correcto funcionamiento, este tipo de fuentes regulan el voltaje de salida utilizando transistores que se encuentran conmutando entre sus regiones de corte y saturación a altas frecuencias, a continuación, se presenta la fuente seleccionada con sus características.



Característica	Valor
Voltaje de alimentación	110 Vca 60Hz
Voltaje de salida	24 Vcd
Corriente de salida	20 A
Potencia	480 W

Tabla 3.3: Características de la fuente conmutada.

Figura 3.11: Fuente conmutada.

Sin embargo, hay que aclarar que el motor se va a utilizar en alrededor del 50 % de su ciclo de trabajo, por lo cual su consumo se encuentra por debajo de los valores nominales de la fuente seleccionada.

3.1.3. Puente H

En el control de motores de corriente continua, es común requerir no solo el encendido y apagado del motor, sino también la inversión de su sentido de giro de forma eficiente y segura. Un puente H es un circuito electrónico que permite lograr este control bidireccional utilizando conmutación de transistores, facilitando tanto el movimiento hacia adelante como hacia atrás sin necesidad de modificar la conexión física del motor. Para nuestro motor se decidió diseñar un puente H utilizando transistores BJT, el diagrama de conexiones se presenta en la Figura 3.12.

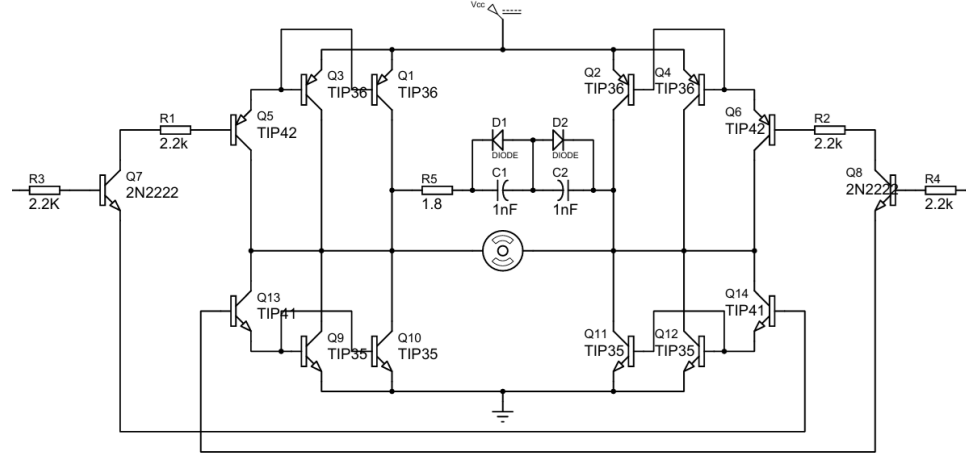


Figura 3.12: Circuito del puente H.

Para obtener los valores de las resistencias se tienen los siguientes cálculos.

La señal de control nos entrega un voltaje de $V_{IN} = 3.3V$ y una corriente de $I_{IN} = 1mA$. Basados en esto tenemos que, para el transistor de activación se utiliza un transistor NPN 2n2222, el cual tiene una ganancia media de $\beta_{IN} = 100$, entonces:

$$\beta_{f_{IN}} = \frac{\beta_{IN}}{10} = \frac{100}{10} = 10, \quad (3.8)$$

$$I_{C_{IN}} = \beta_{f_{IN}} I_{B_{IN}} = 10(1mA) = 10mA. \quad (3.9)$$

Así tenemos que la corriente de entrada a la configuración en Darlington de transistores PNP es de 10mA, con lo cual continuamos con el análisis.

$$I_{C_D} = \beta_e I_{B_D}, \quad (3.10)$$

$$20 \text{ A} = \beta_e (10 \text{ mA}), \quad (3.11)$$

$$\beta_e = 2000. \quad (3.12)$$

Si utilizamos 3 transistores para el Darlington siendo en el caso de los PNP un BC557, un TIP42C y un TIP35C, y para los NPN un 2n222, un TIP41C y un TIP35C, los cuales entre sus características tienen una β media de 100, 40 y 15 respectivamente, entonces la β_e resulta como:

$$\beta_e = 100 \cdot 40 \cdot 15 = 60000, \quad (3.13)$$

$$\beta_{f_e} = \frac{\beta_e}{10} = \frac{60000}{10} = 6000. \quad (3.14)$$

Con esta configuración se tiene una β_e mayor a la necesaria por lo cual podemos garantizar que la corriente entregada será la adecuada, ahora continuamos para obtener las resistencias con las cuales garanticemos la saturación.

Para el caso de la resistencia de protección R_{B_D} se tiene la siguiente ecuación:

$$0 = V_{cc} - 0.7V - 0.7V - 0.7V - R_{B_D}I_{B_D} - 1V - 0.7V - 0.7V - 0.7V, \quad (3.15)$$

$$0 = V_{cc} - 5.2V - R_{B_D}I_{B_D}, \quad (3.16)$$

$$R_{B_D} = \frac{V_{cc} - 5.2V}{I_{B_D}} = \frac{26V - 5.2V}{10 \text{ mA}} = 2080 \Omega \approx 2.2 \text{ k}\Omega. \quad (3.17)$$

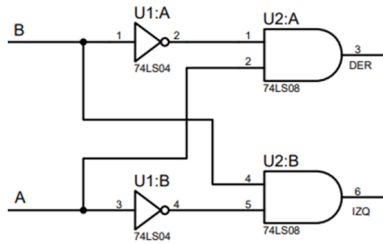
Para el caso de la resistencia de base del transistor de activación se tiene la siguiente ecuación:

$$0 = V_{IN} - R_B I_B - 0.7V - 0.7V - 0.7V - 0.7V, \quad (3.18)$$

$$0 = V_{IN} - R_B I_B - 2.8V, \quad (3.19)$$

$$R_B = \frac{V_{IN} - 2.8V}{I_B} = \frac{5V - 2.8V}{1 \text{ mA}} = 2.2 \text{ k}\Omega. \quad (3.20)$$

Una vez verificado su funcionamiento de manera independiente, lo siguiente fue realizar un circuito de protección para evitar que en el caso de algún fallo en la implementación del algoritmo de control en el FPGA y este nos envíe ambas salidas activas, el circuito no entre encorto y provoque un accidente, por eso se agregó el siguiente circuito a sus entradas para evitar este posible escenario.



A	B	DER	IZQ
1	1	0	0
1	0	1	0
0	1	0	1
0	0	0	0

Figura 3.13: Circuito de protección.

Tabla 3.4: Tabla de verdad para el control de direcciones.

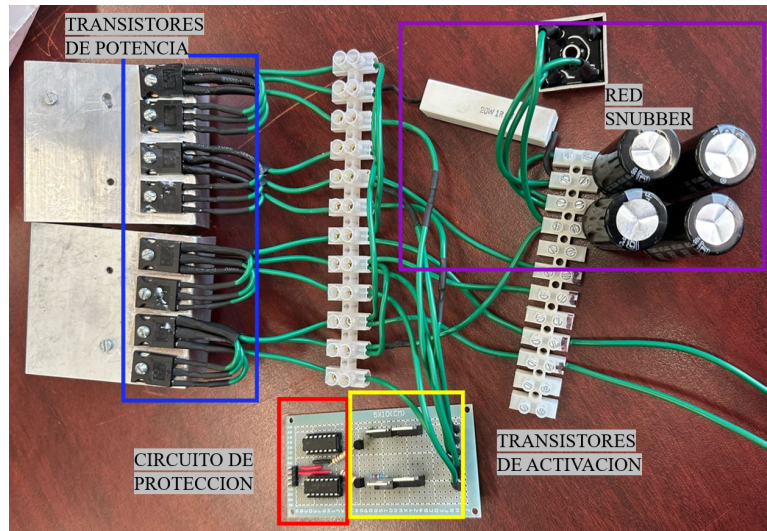


Figura 3.14: Puente H armado.

3.2. Electrónica de adquisición

En este apartado se revisarán las características de las señales que lee el FPGA para cerrar el lazo de control, donde nos tenemos que asegurar que los sensores se alimenten adecuadamente para un correcto funcionamiento y que la señal que entreguen a los puertos de FPGA sean de 3.3V.

3.2.1. Fines de carrera

Los fines de carrera inductivos son dispositivos activados sin contacto, estos se utilizarán como alternativa a los microswitch ya que estos tienden a dañarse muy rápido debido a que el contacto es directo y a veces muy brusco. Las ventajas que ofrecen este tipo de interruptores son el trabajo sin desgaste, presentan resistencia a ambientes agresivos y ofrecen una vida de trabajo prácticamente ilimitada. Los fines de carrera inductivos a utilizar en este caso se presentan a continuación.



Figura 3.15: Sensor inductivo.

Estos sensores son de tipo PNP, entre sus características se encuentran que:

- Voltaje de alimentación es entre 5vdc-36vdc.
- Distancia de detección 4mm.

Debido a que su alimentación mínima es de 5vdc y los puertos del FPGA soportan solo 3.3vdc, se prosiguió a realizar un circuito divisor de voltaje para acoplar la señal de salida del sensor a la entrada de nuestra tarjeta, el diagrama del circuito se presenta a continuación:

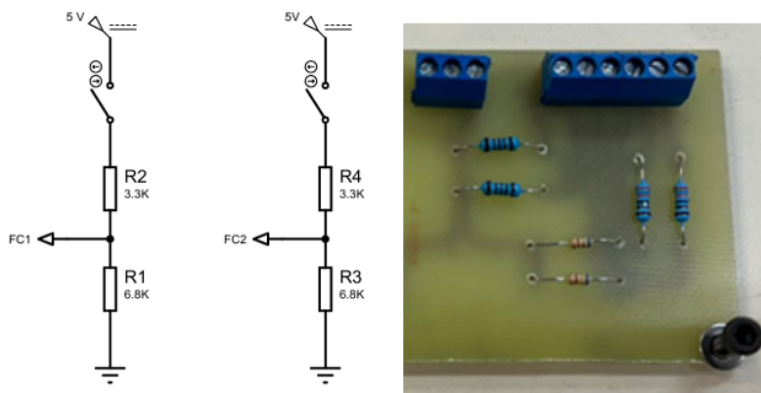


Figura 3.16: Circuito de acoplamiento de los fines de carrera.

Lo siguiente fue realizar el impreso del circuito en una placa fenólica para después realizar las pruebas de su funcionamiento y asegurarnos que las salidas que irán conectadas al FPGA sean de 3.3 Vdc.

Una vez verificado su funcionamiento se prosiguió a colocarlos en el robot, lo primero que se realizó fue colocar una lámina de acero en la parte inferior del carro de tal forma que no interfiriera con su movimiento, para ello se imprimió una pieza en 3D la cual nos serviría de base para la lámina y nos ayudaría a separarla de las llantas del carrito.

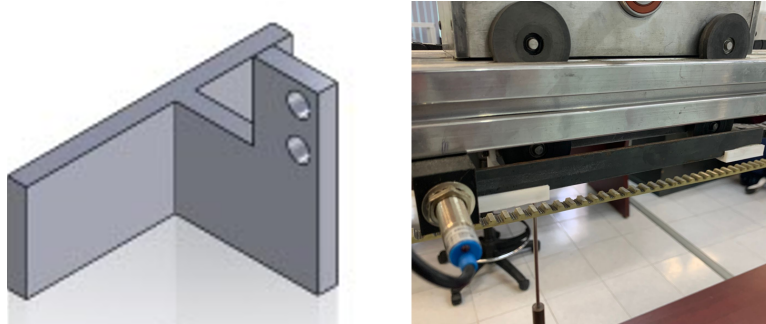


Figura 3.17: Base de lámina.

Una vez colocada la lámina, se prosiguió a determinar la distancia segura a la que deberían estar los sensores para que al momento de detectar el metal estos apaguen el PWM y la distancia que el carro se desplace aun no llegue a estrellarse contra los límites físicos del riel, para lo primero que se realizó fue colocar el sensor a una distancia aproximada de 20 cm del límite derecho, después se posicionó al carro a la izquierda, para verificar la distancia máxima que podría recorrer en un caso donde se descontrola se le proporcionó el 100 % de torque al motor y se midió la distancia que recorrió al llegar al fin de carrera, la cual fue de aproximadamente 8 cm, este experimento se repitió varias veces para asegurarnos que esa era la distancia máxima que podría llegar a pasarse el carro y así fue como obtuvimos que los sensores se colocaron a una distancia de 9 cm de cada límite del riel para garantizar que en algún momento el carro no colisione contra los límites del riel y ocasione algún daño a la estructura.

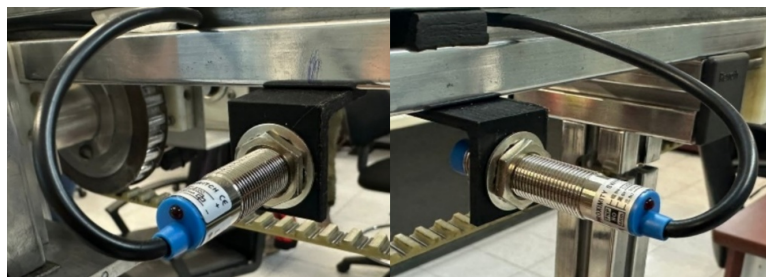


Figura 3.18: Fines de carrera inductivos.

En la Figura 3.18, se pueden observar los fines de carrera de ambos lados ya colocados en el robot.

3.2.2. Encoder

Para conocer algunos datos del sistema se utilizan sensores, los cuales son dispositivos que miden una variable física y la transforman en una señal eléctrica que es procesada por un sistema de adquisición de datos, el péndulo cuenta con dos encoders, el encoder es un transductor rotativo que transforma un movimiento angular en una serie de pulsos digitales, se pueden usar para controlar desplazamientos de tipo angular o de tipo lineal, uno de ellos se encuentra dentro del carro, el cual está conectado a las barras de aluminio de los lados para poder medir la posición del péndulo, el otro se encuentra conectado al eje de la polea fija, este se utiliza para medir la posición del motor, ambos sensores tienen la misma precisión, como se muestra en la Figura 3.19.

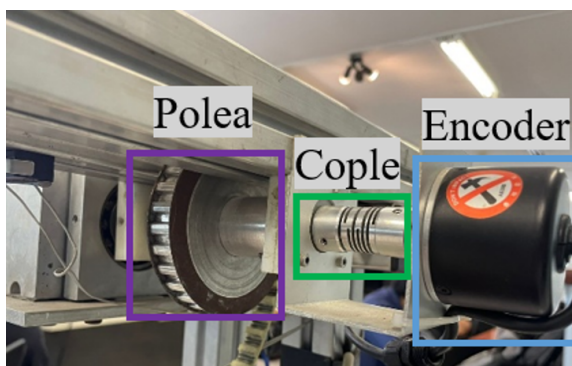
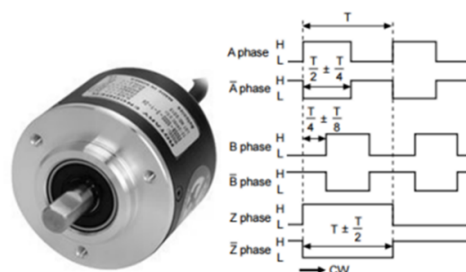


Figura 3.19: Encoder.

El encoder del motor es de la marca Autonics modelo E50S6-2500-6-1-5 ver Figura 3.20, el cual cuenta con las siguientes características [16]:



- Voltaje de alimentación 5 Vdc.
- 6 señales de salida A , $\bar{A}$, B , $\bar{B}$, Z , $\bar{Z}$
- 2500 pulsos por revolución.
- Salida Line Driver.

Figura 3.20: Encoder modelo E50S6-2500-6-1-5 y señales de salida A, B y Z. [16].

Para el péndulo se utiliza un encoder de eje hueco de la empresa SICK modelo DFS60E-BHC02000 (ver Figura 3.21) con las siguientes características:



Figura 3.21: Encoder modelo DFS60-BHC02000.

- Voltaje de alimentación 5 Vdc.
- 6 señales de salida A , $\bar{A}$, B , $\bar{B}$, Z , $\bar{Z}$
- 2000 pulsos por revolución.
- Salida Line Driver.

Debido a que la señal de salida de los encoders es igual a la de su alimentación y la alimentación mínima para su funcionamiento es de 5V entonces es necesario realizar un acoplamiento de la señal para reducir la señal de 5V a 3.3V por lo cual se tiene el circuito mostrado en la Figura 3.22..

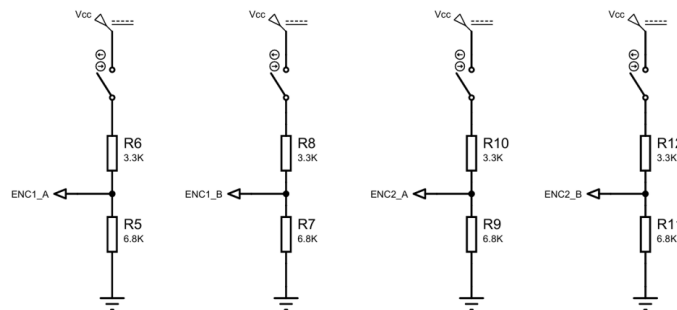


Figura 3.22: Acoplamiento de los encoders.

Ahora para verificar su funcionamiento se realizaron algunas mediciones utilizando un código de propósito general que nos fue otorgado en el laboratorio de Robótica y Control.

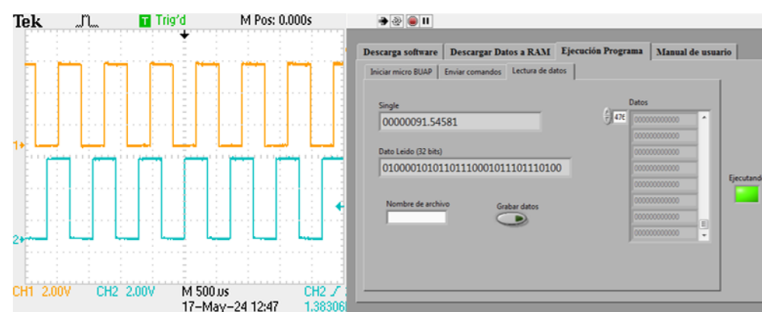


Figura 3.23: Lectura del encoder con el microprocesador.

En la Figura 3.23 se puede observar como el encoder genera las dos señales de cuadratura (a la derecha) y esas son las que ingresan al FPGA que emula el microprocesador, el cual nos envía las lecturas por wifi y la computadora en la interfaz de LabVIEW nos muestra las cuentas que lleva del encoder.

3.3. Integración de la electrónica

Una vez que se verificó el funcionamiento de cada elemento se prosiguió a realizar las conexiones necesarias para integrar todos los componentes en una estructura móvil que nos permita transportar de manera eficiente y sin riesgo a que algún componente se desconecte toda la electrónica utilizada para el control del sistema, esta integración se describe a continuación.

Lo primero fue determinar cuántos niveles se utilizarían, así como que elementos y como se acomodarían en cada nivel, esto con el fin de determinar las dimensiones adecuadas de las bases y los pilares de separación.

Se decidió que se utilizarían 4 etapas, en la primera se colocarían las fuentes que utilizara el sistema, siendo la fuente conmutada de 24 Vdc para el motor y una fuente de 5 Vdc de alimentación del FPGA, así como los encoders y los finales de carrera.



Figura 3.24: Fuente conmutada de 24V y 5V.

En el siguiente nivel se encontrará únicamente el puente H, esto debido a que tiene un volumen considerable y también para evitar que se induzca ruido en las señales de baja potencia de los sensores, debido a que la corriente que circulará por él es considerablemente alta.

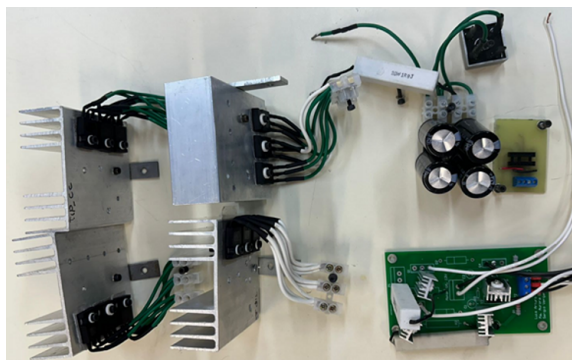


Figura 3.25: Componentes del puente H.

En el siguiente nivel se colocará el FPGA, así como el PCB del acondicionamiento de las señales de los encoders y los finales de carrera.

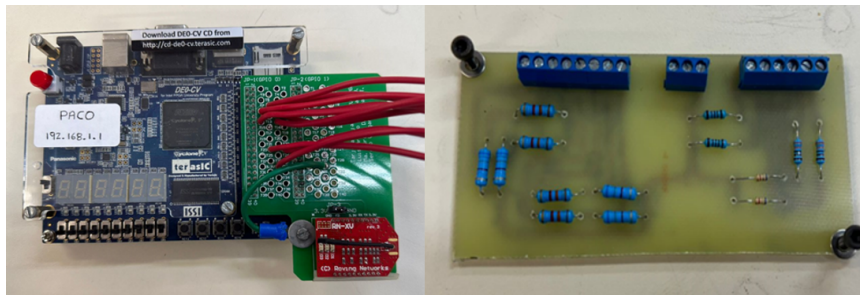


Figura 3.26: FPGA, WI-FI, PCB de los encoders.

Por último, en la parte superior se colocará una clema de conexiones, en la cual por un lado estarán todas las terminales que van al FPGA y al PCB de los sensores así como las terminales de alimentación de 5 Vdc y la salida del puente h que alimenta al motor, por el otro lado estarán todas las conexiones que provienen del robot, esto con la finalidad de poder desconectar la electrónica del robot para transporte o para su resguardo cuando no se esté utilizando.

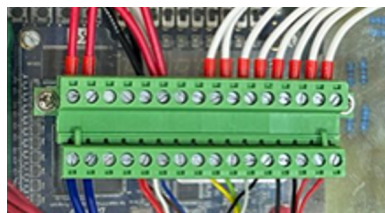


Figura 3.27: Conector.

Una vez que ya tenemos definido como se ubicarán todos los elementos se prosigue a dimensionar las bases, las cuales se cortaron de 25cm x 16cm cada una, así como se decidió la altura de cada nivel para que entraran todos los elementos de manera adecuada, quedando el primer piso de 8cm, el segundo de 9cm y el último de 6cm las piezas se presentan en la figura 3.28.

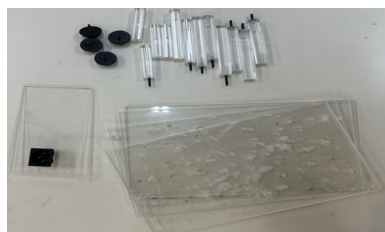


Figura 3.28: Bases de acrílico.

Otra de las cosas que se agregaron fueron un conector para alimentar las fuentes y un interruptor para poder prender y apagar la alimentación del puente H. Una vez cortadas y barrenadas se prosiguió a ensamblar los componentes y ubicarlos de manera adecuada como se decidió anteriormente, el resultado se presenta a continuación.

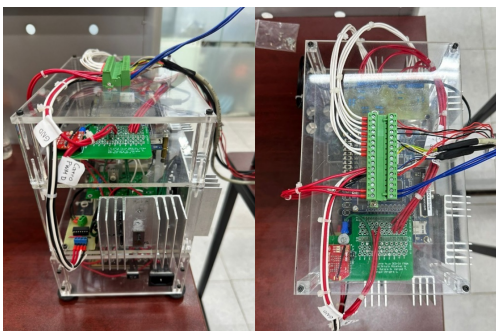


Figura 3.29: Vista lateral y superior de la torre.

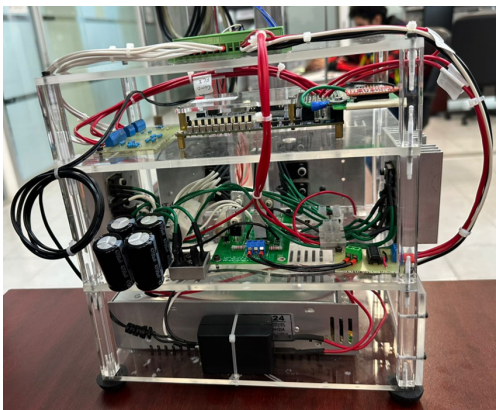


Figura 3.30: Torre de electrónica.

3.4. Conclusiones

El diseño e implementación de la electrónica del robot integran de manera efectiva los sistemas de potencia y adquisición, garantizando el funcionamiento confiable y preciso del péndulo invertido sobre base móvil. La electrónica de potencia incluye una caracterización del motor de corriente continua, lo que permitió determinar sus parámetros operativos y optimizar el desempeño energético del sistema. La fuente de alimentación y el circuito del puente H fueron diseñados para manejar eficientemente las demandas de potencia, asegurando un control preciso en la dirección y velocidad del motor.

Por otro lado, la electrónica de adquisición se centró en la implementación de sensores como los finales de carrera y los encoders, los cuales desempeñan un papel crucial en la supervisión del sistema. Los finales de carrera proporcionan un mecanismo de seguridad para prevenir daños en el robot al limitar físicamente el movimiento del carro. Los encoders, con su alta precisión, permiten medir las posiciones y velocidades tanto del péndulo como de la base móvil, asegurando la retroalimentación necesaria para los algoritmos de control.

En conjunto, la electrónica del robot proporciona una infraestructura robusta que combina potencia, monitoreo y control, asegurando un desempeño óptimo del sistema.

Capítulo 4

Firmware del microcontrolador

En este capítulo se describirá el firmware proporcionado en el laboratorio de robótica y control, el cual consiste en un microcontrolador embebido en una tarjeta de adquisición FPGA, el cual cuenta con puertos de salida y entrada para diferentes aplicaciones, también se describe la técnica de decodificación de los encoder, así como los problemas que se presentan durante la digitalización y como se utiliza el filtrado para solucionar este problema.

4.1. Sistema embebido a implementar como lazo de control

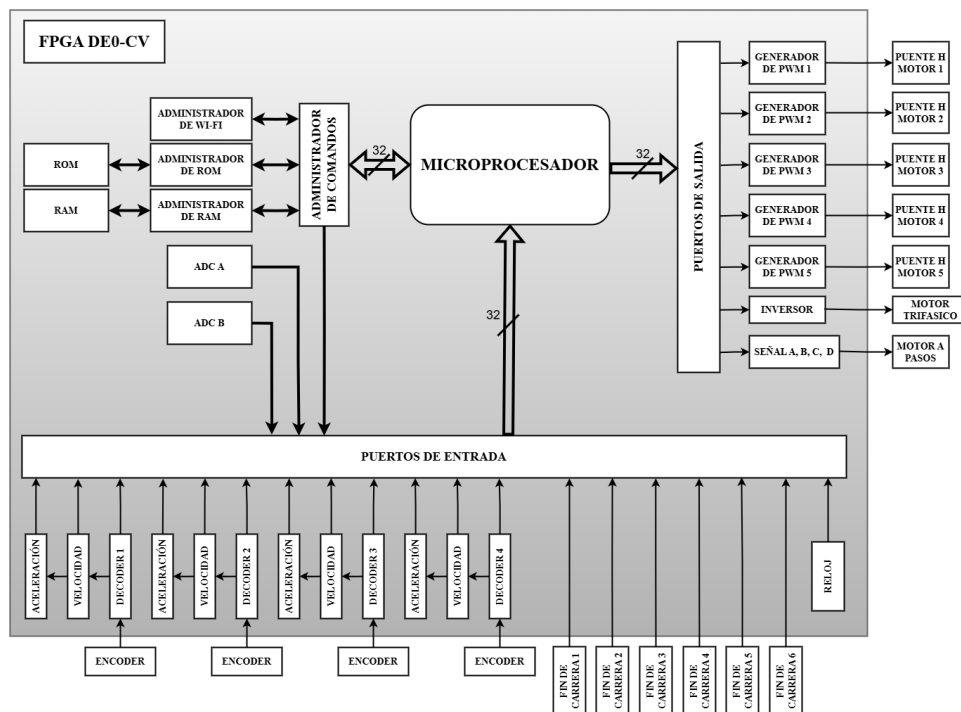


Figura 4.1: Diagrama general del FPGA.

El sistema del péndulo invertido se pretende trasladar a un nuevo procesador desarrollado en el laboratorio de Automatización. Este microprocesador está interconectado con “núcleos de propiedad intelectual” (IP Core, por sus siglas en inglés) a través del puerto de entrada y salida (Figura 4.1), con tareas para leer y escribir datos de propósito específico, sin embargo, como se ilustra en el diagrama de la Figura 4.1, el dispositivo recibido tiene recursos limitados, entre ellos el microprocesador, el módulo de administración de datos WIFI, administrador de datos en memoria RAM y EEPROM, por lo que es necesario acondicionar los recursos necesarios para nuestro sistema embebido, entre ellos: módulo generador de señal PWM, decodificador de posición de motor, entre otros [23].

Para el control del robot es necesario decodificar la señal de posición del motor y el péndulo, saber si el carro ha llegado a su fin de carrera, además de mandar la acción de control a través del motor con la señal PWM, para ello, a este sistema se le añadirán los módulos correspondientes para lograr el cometido. Cabe mencionar que el sistema embebido cuenta con memoria RAM y en ella se cuentan con miles de direcciones de 32 bits disponibles (ver la Tabla 4.3) con el fin de almacenar diferentes variables y cálculos necesarios en el lazo de control [20].

El procesador otorgado por el laboratorio de robótica posee 16 puertos de entrada y 16 puertos de salida, cada puerto es de 32 bits y es posible acceder a ellos, ya sea para conectar dispositivos de forma externa, o en su caso, conectar módulos internos IP diseñados en VHDL o AHDL. La Tabla 4.1 representa el conjunto de puertos de entrada disponibles, donde el puerto cero está reservado para la entrada de datos derivada del módulo WIFI, mientras que la Tabla 4.2 representa los puertos de salida con los que cuenta el procesador para administrar recursos variados, de igual forma, el puerto cero es utilizado para la transferencia de datos a través del módulo WIFI. Huelga decir que el bit 32 es un bit de signo por lo que típicamente éste no es utilizado cuando se envían datos tipo entero sin signo [23].

Nº PTO	Disposición de bits de puerto	
PT0	No usar PT0[32]	Reservado para la lectura datos WIFI PT0[31-1]
PT1	No usar PT1[32]	Bits de usuario disponible PT1[31-1]
PT2	No usar PT2[32]	Bits de usuario disponible PT2[31-1]
⋮	⋮	⋮
PT15	No usar PT15[32]	Bits de usuario disponible PT15[31-1]
PT16	No usar PT16[32]	Bits de usuario disponible PT16[31-1]

Tabla 4.1: Distribución de puertos de entrada.

Nº PTO	Disposición de bits de puerto	
PT0	No usar PT0[32]	Reservado para la escritura datos WIFI PT0[31-1]
PT1	No usar PT1[32]	Bits de usuario disponible PT1[31-1]
PT2	No usar PT2[32]	Bits de usuario disponible PT2[31-1]
⋮	⋮	⋮
PT15	No usar PT15[32]	Bits de usuario disponible PT15[31-1]
PT16	No usar PT16[32]	Bits de usuario disponible PT16[31-1]

Tabla 4.2: Distribución de puertos de salida.

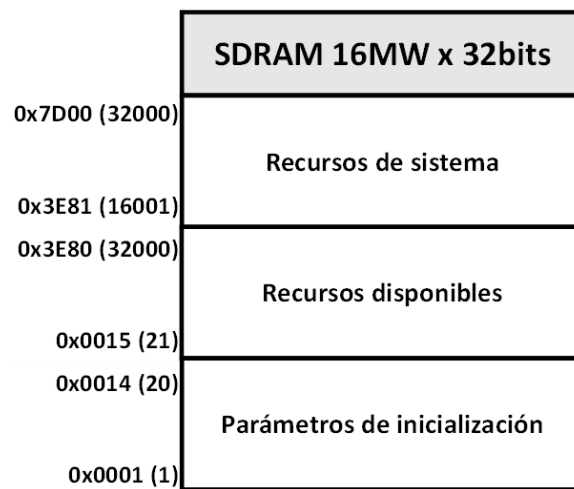


Tabla 4.3: Recursos disponibles y mapeo de memoria RAM.

Para nuestro sistema, se utilizan dos entradas de encoder, y dos salidas de PWM para invertir el giro del motor, además de dos entradas de fines de carrera para delimitar el recorrido de la base móvil. Por lo que el diagrama del sistema utilizado se presenta a continuación.

En la Figura 4.2 se pueden apreciar las conexiones de los elementos que se utilizarán para el monitoreo y control del robot, se aprecian los bloques de los dos fines de carrera, la señal del encoder del péndulo y el motor entra al FPGA y dentro de esta, esas señales llegan a los módulos de decoder para decodificar la posición de estos dos elementos, así como el generador de PWM que envía las señales a la etapa de potencia que alimenta al motor.

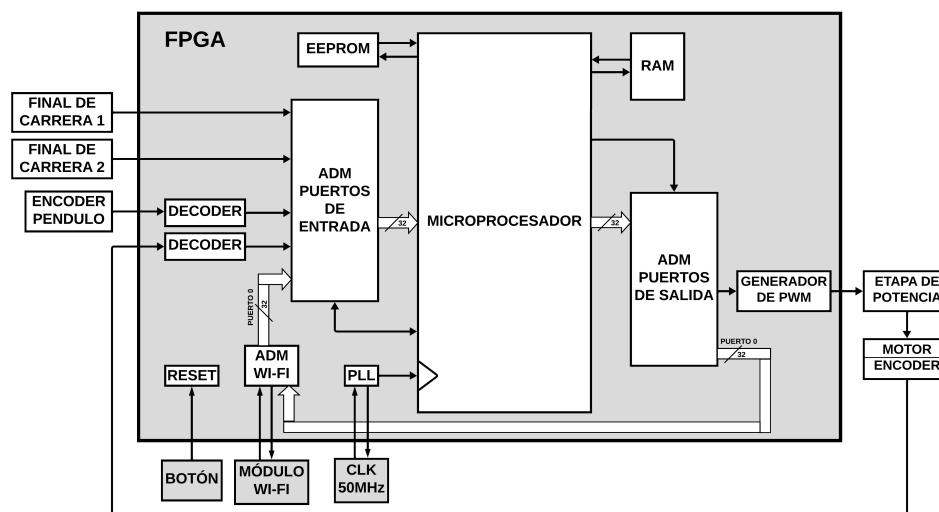


Figura 4.2: Diagrama a bloques del FPGA para el robot.

4.2. FPGA DE0-CV

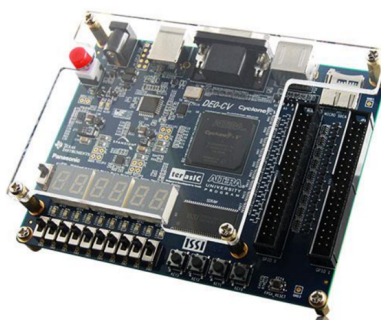


Figura 4.3: Tarjeta de desarrollo DE0-CV [27].

En la Figura 4.3, se presenta la tarjeta de desarrollo que se utiliza para emular el microcontrolador, la tarjeta provee un entorno de desarrollo complejo con puertos de entrada salida de tipo USB, VGA y PS/2; además cuenta con dos bloques de expansión para terminales que nos permiten acceder a 40 pines que pueden ser configurados como entrada o salida en función de los intereses de desarrollo. Es posible acceder a 4 botones y a un bloque de 10 switch que pueden ser utilizados para generar interrupciones o para programar diferentes funciones dentro de la FPGA, como se puede observar en la Figura 4.4. La tarjeta cuenta con capacidades de memoria con ranuras de expansión para tarjetas SD y una SDRAM propia para almacenar las instrucciones de programas y tiene un total de 49000 compuertas lógicas; se ha manejado para emular un microprocesador conectado a núcleos de propiedad intelectual, IP-Core, a través de los puertos de entrada/salida con funcionalidades para la lectura/escritura de datos con propósitos generales y específicos. Para un adecuado funcionamiento es necesario el diseño de los bloques de firmware interno desde los

cuales se manejan los recursos disponibles, la programación del firmware se realiza utilizando el lenguaje de programación AHDL [27].

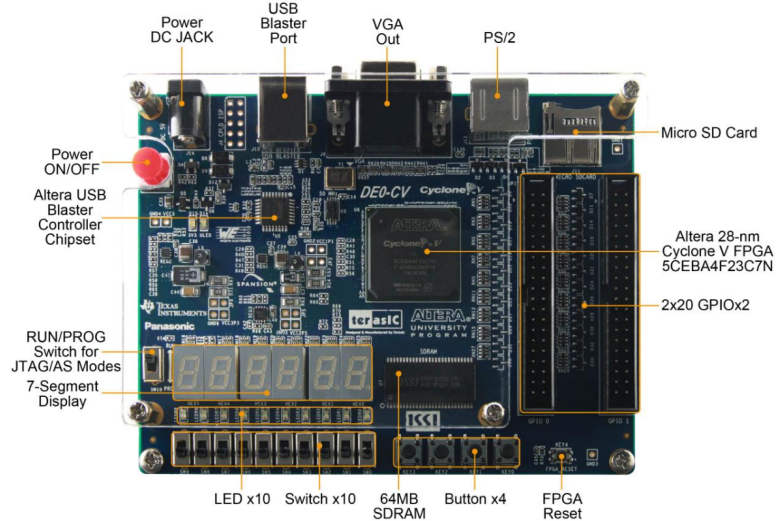


Figura 4.4: Descripción física de la tarjeta DE0-CV.

Algunas especificaciones de la tarjeta de desarrollo DE0-CV son:

- Procesador Cyclone V 5CEBA4F23C7N
- Número de compuertas lógicas: 49000
- Memoria SDRAM de 64 MB
- Software Quartus II
- 2 expansiones de 40 pines GPIO: 36 pines I/O, 2 fuentes (3.5 V y 5V) y 2 de tierra (GND).
- Periféricos genéricos: 4 push buttons, 10 switch, 10 leds, 6 pantallas 7 segmentos.
- Otros periféricos: Salida VG, expansión SD, PS/2.
- Método de programación a través de Quartus II.

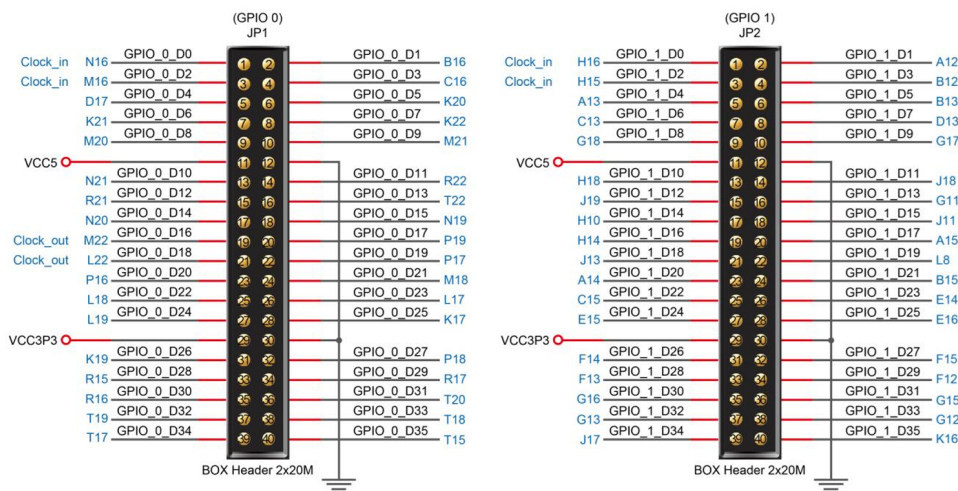


Figura 4.5: Puertos GPIO de la tarjeta DE0-CV [27].

4.3. Módulo WI-FI

El microprocesador emulado dentro de la FPGA tiene la capacidad de comunicarse con diversos dispositivos a través del protocolo WIFI, dicho módulo IP es capaz de administrar los comandos necesarios para comunicar el microprocesador con el módulo de conexión inalámbrica llamado “wifly”, mismo que incorpora el circuito integrado RN-XV, capaz de lograr toda la comunicación por este protocolo (Figura 4.6) [20] [23].

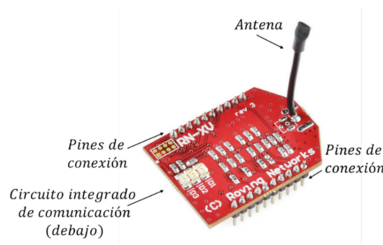


Figura 4.6: Módulo de comunicación inalámbrica Wifly [20].

Como se ha mencionado antes, es posible acceder al módulo WIFI a través del puerto cero (véase la Tabla 4.4) utilizando la instrucción “EPOR Rx PT0” para escribir algún dato o “LPOR PT0 Rx” para leer los datos provenientes del módulo WIFI.

Nº PTO	Disposición de bits de puerto	
PT0	No usar PT0[32]	Reservado para la lectura/escritura datos WIFI PT0[31-1]

Tabla 4.4: Uso del puerto cero para acceso a datos WIFI.

El puerto de entrada y salida PT0 está intercomunicado con un módulo interno de comunicación serial UART, el protocolo de comunicación serial tiene como objetivo entablar comunicación entre dos dispositivos a través de dos señales, Rx y Tx, donde la señal Rx de un dispositivo puede leer los datos recibidos a través de la señal Tx de otro dispositivo y viceversa (Figura 4.7).

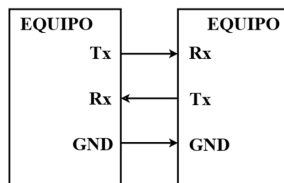


Figura 4.7: Comunicación serial entre dispositivos a través de señal Rx y Tx [20].

Para utilizar el módulo Wifi es necesario configurarlo a los parámetros de operación del módulo ip serial del procesador, estos consisten en:

- Velocidad de transferencia (baudios): 460800

- Tamaño de palabra: 5

Los demás parámetros (canal de operación, frecuencia, etc.) pueden mantenerse con los valores de fábrica. Para lograr la configuración, usamos el programa “Tera-term”, y una vez abierto, es necesario indicar la dirección IP del módulo (192.168.1.1 para este dispositivo), seleccionar servicio Telnet e indicar el puerto TCP 2000 (Figura 4.8).

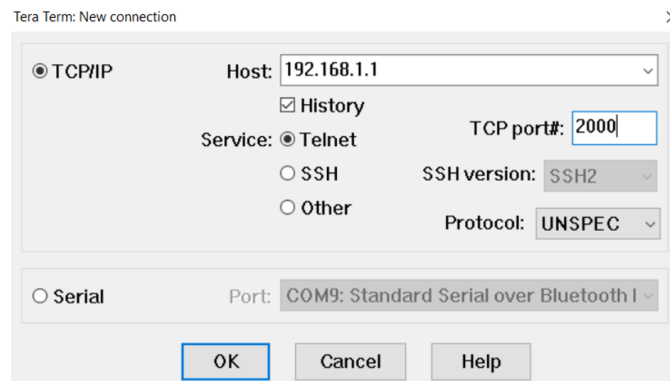


Figura 4.8: Ventana de configuración de módulo Wifi con Teraterm [20].

Una vez abierta, se habilitará la ventana de comandos del terminal y eso significa que el dispositivo está entablando comunicación con la computadora, posteriormente es necesario añadir los comandos correspondientes, estos se enlistan a continuación [20]:

- \$\$\$: este comando nos permite entrar a modo configuración, al dar click “enter”, recibimos el mensaje “CMD”
- set uart baudrate 460800: Con este comando cambiaremos la velocidad de transferencia de datos serial a 460800 baudios por segundo, el haber ejecutado este comando adecuadamente devuelva “AOK” como respuesta.
- set comm size 5: De esta forma se cambia el tamaño de la palabra a 5 palabras de 8 bits a transferir por paquete, de igual forma se recibe “AOK” como respuesta si se configura adecuadamente.
- save: Este comando guarda los cambios hechos en la configuración del módulo.
- Reboot: Finalmente, se reinicia el dispositivo con este comando y así salir de modo programación, con ello será suficiente la configuración del dispositivo para operar con el microprocesador.

4.4. Decoder

Los encoder son sensores que generan señales digitales en respuesta al movimiento. Están disponibles en dos tipos, uno que responde a la rotación y otro que responde al movimiento lineal. Cuando son usados en conjunto con dispositivos mecánicos tales

como engranes, ruedas de medición o flechas de motores, estos pueden ser utilizados para medir movimiento lineal, velocidad o posición, en nuestro caso utilizaremos encoder del tipo incremental.

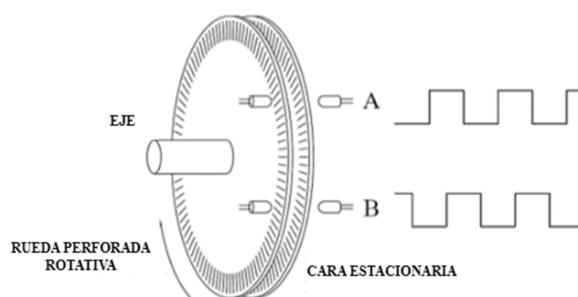


Figura 4.9: Encoder incremental [15].

Este tipo de encoder se caracteriza porque determina su posición, contando el número de impulsos que se generan al detectar el momento cuando un haz de luz atraviesa por unos orificios o marcas opacas en la superficie de un disco unido al eje.

Las señales de salida de un encoder incremental es un tren de impulsos, en forma de señal cuadrada, en el caso de contar con dos canales, como es nuestro caso, una de las dos señales estará desfasada 90° con respecto a la otra, como se puede observar en la Figura 4.10 [21].

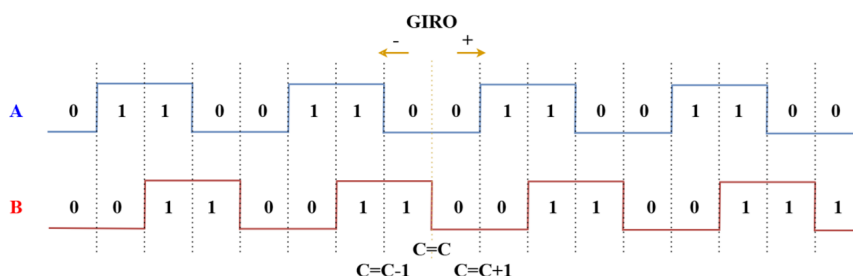


Figura 4.10: Señal del encoder incremental de dos canales A y B [21].

Haciendo el conteo de los flancos (subida y bajada) de ambas salidas se puede obtener un numero de conteos c en cada revolución del motor. Las señales entregadas A y B se encuentran desfasadas 90° , esto nos permite realizar la Tabla 4.5.

	Caso 1	Caso 2	Caso 3	Caso 4
B	0	0	1	1
A	0	1	1	0

Tabla 4.5: Variables Fundamentales.

La frecuencia de estas señales puede variar segundo la velocidad del motor, pero siempre conservan la misma relación de 90° , esto nos permite traducir estos datos sin embargo es requerido generar un complemento con la programación que traduzca la señales a la distancia recorrida [21].

La primera parte es conseguir un ciclo de muestreo que nos permita recuperar los datos en cualquier instante, esto dado que si se llega a perder información de cualquiera de los 2 canales la información recolectada esta errónea y dado que no se mide más información tendremos una implementación errónea.

Para generar el ciclo de muestreo de manera que nunca se pueda perder la información hay que cumplir 2 cosas la primera es que sea lo suficientemente rápido para que no importe si el motor se encuentra en su velocidad máxima se siga recolectando, también que pueda tomar un mínimo de 4 mediciones por ciclo.

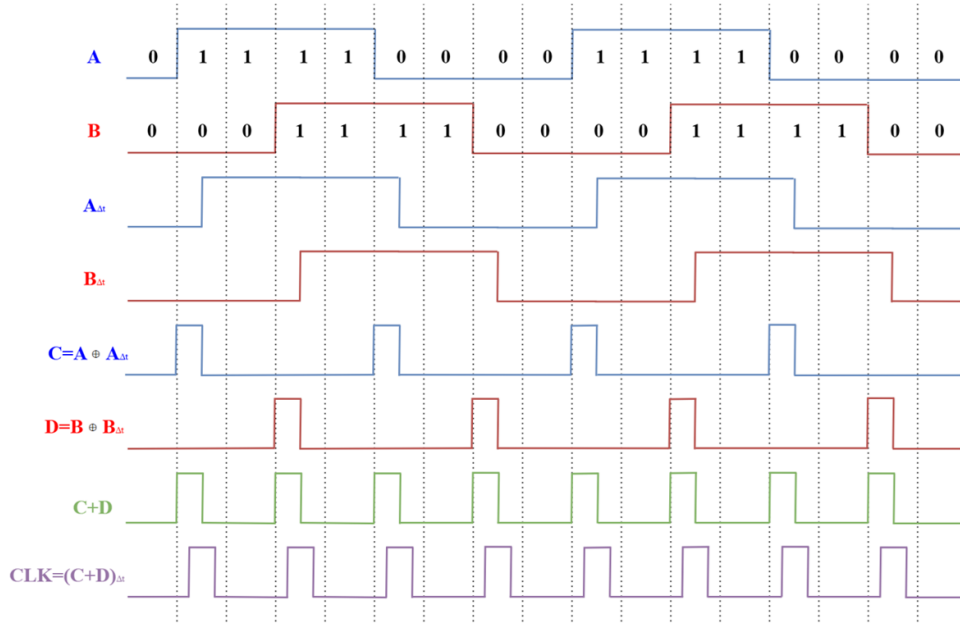


Figura 4.11: Diagrama de tiempos, donde se muestra la recuperación de la señal de reloj a partir de las señales A y B de un encoder.

En la Figura 4.11 se pueden observar los pasos para obtener el reloj, primero se tomarán en paralelo los datos de entrada A y B. Se generará un retraso através de 3 flip-flop, a partir de este punto llamaremos a estas señales Ar y BR. Se realiza una operación de XOR con sus originales respectivamente A con Ar, será denominado como D y B con BR, donde la salida le llamamos E. Se suman ambos resultados D y E, además se realiza un último atraso multiplicando por Delta t [21].

El Delta t se obtienen con el tiempo mínimo se consigue por medio de la frecuencia máxima del motor el cual proviene de la velocidad máxima del motor según las siguientes formulas.

$$\text{FrecuenciaMaxAB} = N_V V_{\max}, \quad (4.1)$$

donde N_V corresponde a la cantidad de eventos necesarios de A y B para que se realice un periodo, y $V_{\max}$ como la velocidad máxima del motor.

$$T_{\min} = \frac{1}{\text{FrecuenciaMaxAB}}. \quad (4.2)$$

Ya con el tiempo mínimo se obtiene Delta T a partir de un octavo de este.

$$\Delta T = \frac{1}{8} T_{min}. \quad (4.3)$$

En la Figura 4.12 se pueden observar el circuito para obtener el reloj para poder recuperar la información.

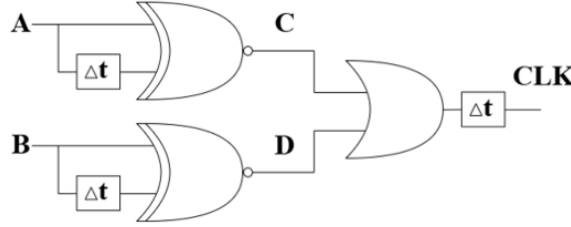


Figura 4.12: Circuito para recuperación de la señal de reloj.

La razón de los Flip-Flop es para que nunca haya un momento en el cual la señal pase sin atraso pues ha habido casos en el que se logra sintonizarse con 2 Flip-Flop, pero con 3 es imposible. Por otro lado, el que ΔT sea un octavo de la mínima es para que tenga el tamaño máximo para suceder en cada una de las medidas de A y B a pesar del retraso. Si se trata de un motor con caja de engranajes es necesario tomar en cuenta la transformación para los cálculos siendo la salida tras esta la contada para los cálculos. Después de ese proceso ya que podemos recibir toda la información hace falta interpretarla para poderlo hacer es necesario desarrollar un programa que funcione según los pasos que vaya dando el motor y acumule esos valores, de forma que al terminar el proceso tengamos conciencia de cuantos pasos haya dado el motor y por lo tanto conocer el movimiento realizado.

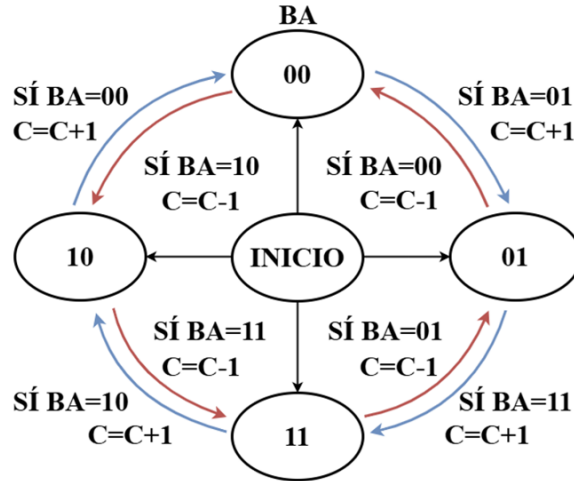


Figura 4.13: Diagrama de estados del decoder [21].

4.5. Efecto de muestreo

Es el proceso por el cual la señal analógica se mide a intervalos regulares de tiempo. La tasa de muestreo determina con qué frecuencia se toman estas muestras.

Cuanto mayor sea la frecuencia de muestreo, mayor será la precisión de la representación digital, ya que se capturan más detalles de la señal analógica original [22].

El muestreo de la función $f(t)$ matemáticamente se puede expresar con la siguiente ecuación:

$$f_m(t) = f(t)Comb(t) = f(t) \sum_{n=-\infty}^{\infty} \delta(f - nT). \quad (4.4)$$

La ecuación anterior la podemos representar gráficamente como se observa en la Figura 4.14.

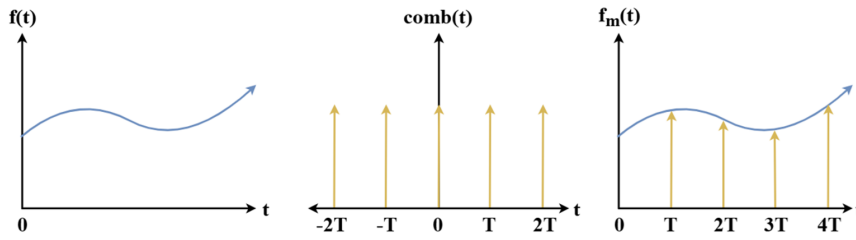


Figura 4.14: Muestreo de una señal $f(t)$.

De acuerdo con las propiedades de la transformada de Fourier la multiplicación de dos funciones en el tiempo es igual a la convolución de sus transformadas de Fourier en frecuencia [22].

$$F(\omega) * \omega_0 \sum_{n=-\infty}^{\infty} \delta(\omega - n\omega_0) = \frac{\omega_0}{2\pi} \int_{-\infty}^{\infty} F(T)Comb(\omega - T) dT, \quad (4.5)$$

con

$$\mathcal{F}[Comb(t)] = \omega_0 \sum_{n=-\infty}^{\infty} \delta(\omega - n\omega_0), \quad (4.6)$$

y

$$\mathcal{F}[f(t)] = \mathcal{F}(\omega). \quad (4.7)$$

La representación gráfica de las transformadas de Fourier tanto para $f(t)$ como para el peine de Dirac $comb(t)$, se muestran en la Figura 4.15.

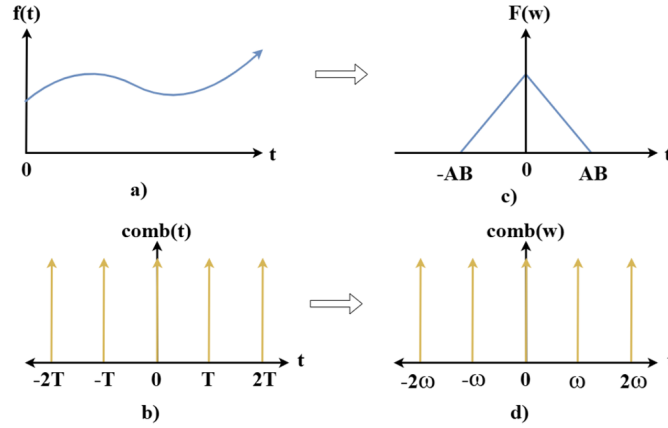


Figura 4.15: Transformada de Fourier de $f(t)$ y $\text{Comb}(t)$.

La convolución puede presentar tres diferentes resultados según el valor del Ancho de Banda (AB) de la señal $f(t)$ respecto de la frecuencia de muestreo (ω_0), los tres posibles resultados muestran en la Figura 4.16.

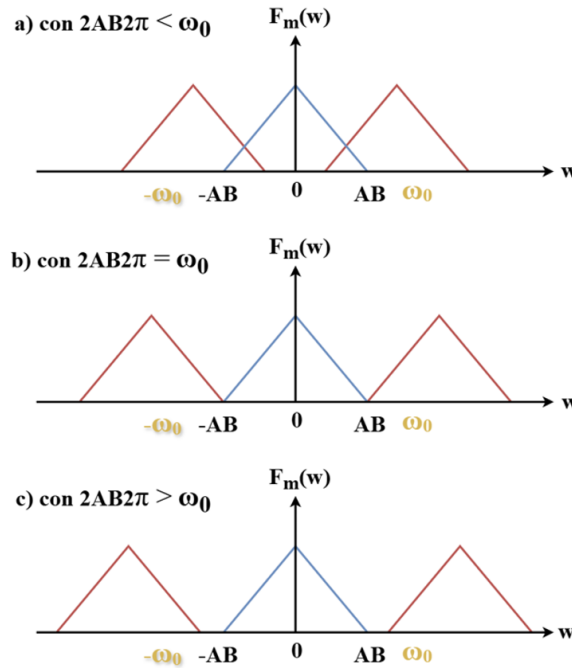


Figura 4.16: Resultados de la convolución entre $F(\omega)$ con el peine de Dirac $\text{Comb}(\omega)$ [21].

En el caso (a) de la Figura 4.16 la información se pierde al encontrarse traslapadas los espectros $F(\omega)$ y no hay forma de recuperar la señal $f(t)$, en el caso (b) de la Figura 4.16, la información se puede recuperar únicamente empleando un filtro pasa bajas ideal que recupere únicamente el espectro $F(\omega)$ centrado en $\omega = 0$, en el caso (c) de la Figura 4.16, se puede recuperar la información con un filtro pasa bajas normal, siempre y cuando solo se conserve el espectro $F(\omega)$ central y elimine los demás espectros no deseados [21].

Por lo anterior, podemos decir que el muestreo genera espectros no deseados en el dominio de la frecuencia, estos espectros pueden ser eliminados por medio de un filtro pasa bajas.

4.6. Filtro pasa bajas

Los filtros son elementos diseñados para dejar pasar una señal a una cierta frecuencia deseada y atenuar las que no se encuentren en ese rango de frecuencia, estos dispositivos pueden modificar la amplitud de la señal de entrada como su fase [22]. Existen distintos tipos de filtros, en este caso particular se necesita emplear un filtro de tipo pasa-bajas, el cual permite el paso de frecuencias muy bajas hasta un valor determinado, como se muestra en la figura 4.17.

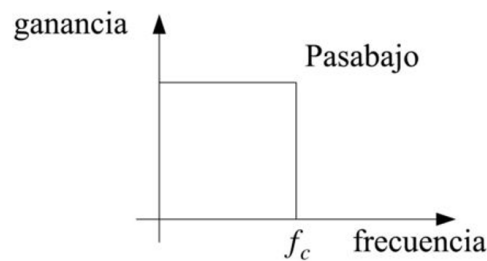


Figura 4.17: Grafica de Ganancia del Filtro Pasa bajas.

Como observamos en la figura 4.18, el espectro de nuestra señal muestreada se va repitiendo cada cierto tiempo, por lo que para recuperar la señal original sin que esas repeticiones tengan efecto, se utiliza un filtro pasa bajas que se diseña de tal forma que solo deje pasar el espectro de la señal original y no las repetidas, como se muestra en la figura 84.

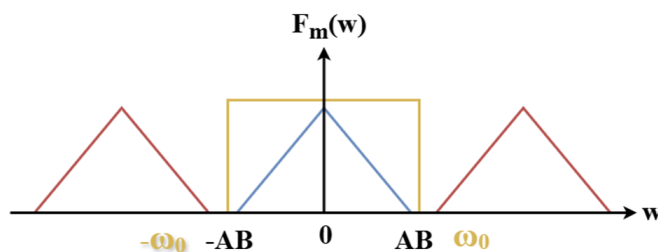


Figura 4.18: Filtrado de la señal recuperada.

4.7. Error en la cuantificación de la velocidad en función de la posición

El proceso de cuantificación de la velocidad consiste en obtener los datos de la velocidad del sistema, tomando en consideración que el reloj del decoder no está sincronizado con el reloj del FPGA, por lo que se hace un ajuste a ecuación tradicional

para contemplar este efecto y así obtener de forma correcta el valor de la velocidad en cada muestra.

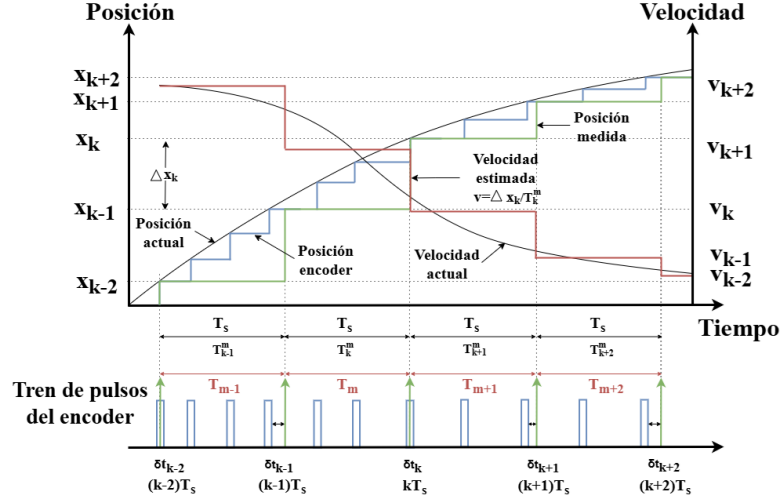


Figura 4.19: Cuantificación correcta de la velocidad en función de la posición [21].

$$\nu_m^{MT} = \frac{x_k - x_{k-1}}{T_s + \delta t_{k-1} - \delta t_k}, \quad (4.8)$$

donde

- ν_m^{MT} es la velocidad muestreada en cada ciclo del FPGA.
- x_k es la posición de la muestra actual.
- x_{k-1} es la posición de la muestra anterior.
- T_s es el periodo de cada muestra.
- δt_k es el tiempo que transcurrió desde el último pulso del encoder hasta la toma de la muestra.
- δt_{k-1} es el tiempo que transcurrió desde la muestra anterior al pulso del encoder anterior a esa muestra.

4.8. Generador de PWM

El PWM o modulación por ancho de pulsos es una técnica utilizada para regular la velocidad de giro de los motores eléctricos de inducción o asíncronos. Mantiene el par motor constante y no supone un desaprovechamiento de la energía eléctrica. Consiste en controlar el tiempo en que el pulso se mantiene en alto y el tiempo que está en bajo.

En este caso se utilizará un FPGA con una frecuencia de trabajo de 100 MHz, por lo que los tiempos usados en el generador de PWM serán controlados por medio de contadores que medirán el tiempo en términos de ciclos de reloj del dispositivo

usado. Para la construcción del bloque de PWM hay que determinar la entrada y salida que usaremos en este caso tenemos como entradas: el periodo de operación, este periodo se obtiene por medio de contar N_{clk} ciclos de reloj respecto al periodo de la señal de reloj del FPGA, en nuestro caso es de 10ns, debido a que la frecuencia del reloj del FPGA es de 100 MHz. El ancho del pulso también será un grupo de bits de entrada, estará determinado por N_{AP} y otro bit de entrada es dirección que indica la dirección de giro del motor. Como salidas tendremos PWM+ y PWM-, cada salida está en función de la dirección de giro requerida. Para ello se propone el siguiente bloque [21]:

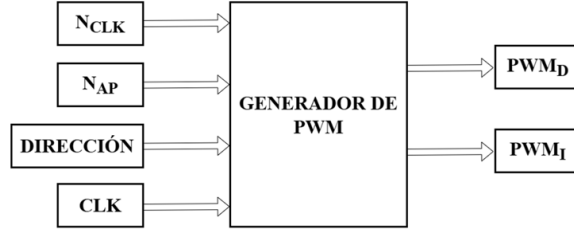


Figura 4.20: Diagrama del PWM [21].

En el cual tenemos las siguientes variables.

$$N_{clk} = \frac{T_{pwm}}{T_{clk}}, \quad (4.9)$$

donde: N_{clk} es el número de ciclos que corresponde a un periodo del PWM, T_{pwm} representa el periodo del PWM, y T_{clk} es el periodo del reloj del FPGA.

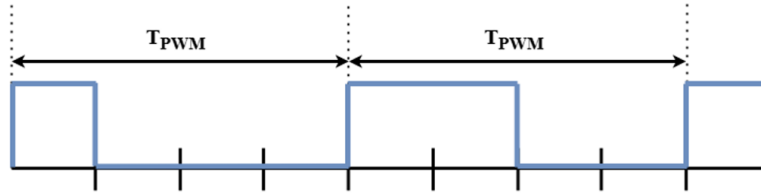


Figura 4.21: Diagrama de tiempos del PWM [21].

$$\Delta AP = \frac{T_{pwm}}{N_{APmax}}, \quad (4.10)$$

donde: ΔAP es un número que representa el paso mínimo para establecer el ancho de pulso, se obtiene dividiendo el periodo del PWM (T_{pwm}) entre el número máximo de particiones deseadas (N_{APmax}); en nuestro caso será de 1000. El número que representa el tiempo del ancho de pulso es:

$$N_{AP} = \frac{AP}{\Delta AP}, \quad (4.11)$$

donde: N_{AP} es el número que representa el ancho de pulso solicitado, AP es el tiempo que el pulso estará en alto, y ΔAP es el número que corresponde a la partición mínima del periodo del PWM. La dirección es determinada por el controlador [21].

4.9. Conclusión

El desarrollo del firmware para el péndulo invertido sobre base móvil permitió establecer un sistema embebido eficiente y preciso que sustenta el control y procesamiento en tiempo real del robot. La implementación en la FPGA DE0-CV proporcionó una plataforma robusta y flexible para integrar los módulos esenciales, como el generador de señales PWM y el decodificador, garantizando un control preciso del motor y una respuesta rápida del sistema.

El uso del módulo Wi-Fi facilitó la comunicación entre la PC y el robot, permitiendo configurar el sistema y monitorear su desempeño sin afectar los tiempos críticos del control, ya que todo el procesamiento se realiza localmente en la FPGA. Además, se incorporaron técnicas avanzadas, como el filtrado pasa bajas y la compensación de errores en la cuantificación de velocidad en función de la posición, que mejoraron significativamente la precisión y estabilidad del sistema.

El análisis del efecto del muestreo y la implementación del filtro digital aseguraron un tratamiento adecuado de las señales, minimizando la distorsión y optimizando la adquisición de datos. En conjunto, el firmware desarrollado combina innovación, eficiencia y precisión, sentando las bases para un control confiable y robusto del péndulo invertido, así como para futuros desarrollos en sistemas embebidos avanzados.

Capítulo 5

Software de usuario

En este capítulo se describirá el software utilizado para la configuración del sistema embebido, el cual consta de un lenguaje de programación en alto nivel en donde se describen las instrucciones para inicializar al sistema, obtener y procesar los datos, y las instrucciones para el controlador a implementar, después el código descrito se pasa por un compilador que lo convierte a un archivo hexadecimal el cual con ayuda de la interfaz de usuario se descarga al FPGA para su implementación, a continuación se presenta un diagrama de flujo de los pasos a seguir para cargar el software.

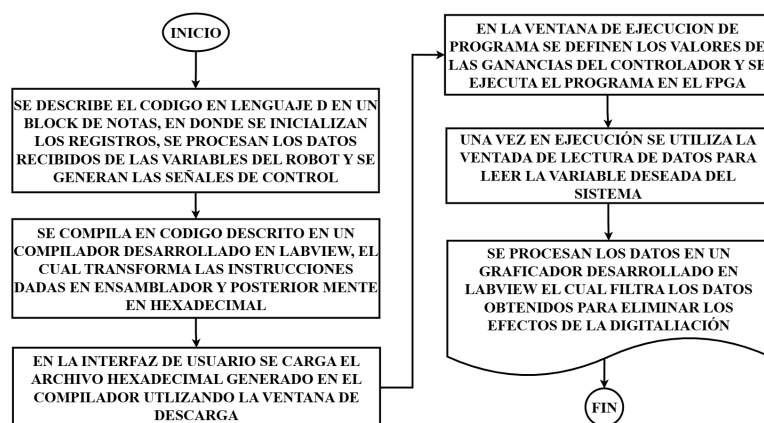


Figura 5.1: Diagrama del proceso de configuración del software.

5.1. Lenguaje de programación del microprocesador (Lenguaje D)

El microprocesador emulado dentro de la tarjeta de desarrollo FPGA contiene una unidad aritmética lógica (ALU, por sus siglas en inglés) capaz de ejecutar, entre otros, funciones matemáticas básicas y trigonométricas, operadores de salto de línea de instrucción, comparadores (IF, WHILE, FOR), operación bit a bit (instrucción AND, OR), entre otros.

Es posible ejecutar las diferentes funciones del microprocesador acorde a las necesidades de nuestro algoritmo a implementar, para lograrlo, se ha desarrollado un

compilador que “traduce” el lenguaje de programación de alto nivel a lenguaje máquina, las instrucciones con las que se cuentan se describen en la tabla 5.1. El lenguaje de programación posee instrucciones similares al lenguaje “C”, las diferencias radican principalmente en que este lenguaje de programación está orientado a registros, es decir, está limitado a apuntadores dentro de la memoria RAM del microprocesador, (en el lenguaje de programación, los registros son invocados con la letra R_i). Las instrucciones principales de este lenguaje de programación son en el poder hacer sumas, productos, funciones trigonométricas, el resultado es almacenado en un registro cualesquiera y el argumento de los operadores pueden ser números naturales o registros, según lo necesario.

Para acceder a la memoria RAM se han implementado las instrucciones LRAM y ERAM, la primera permite leer la memoria RAM la última la escritura en memoria RAM. La sintaxis de lectura de memoria RAM es “LRAM Z R_i ” donde Z es la dirección de la memoria RAM a la cual se desea acceder y R_i representa el registro en el cual se desea almacenar el dato leído en la memoria RAM de la dirección Z; para la escritura en la memoria RAM se implementa la instrucción “ERAM R_i Z” donde R_i es el registro que posee el dato que deseamos almacenar en memoria RAM y Z es la dirección en la cual se almacena el dato.

El procesador trabaja de forma nativa en punto flotante, lo que significa que para hacer operaciones matemáticas y comparaciones, estas se ejecutarán en punto flotante, no obstante, existe la posibilidad de convertir un dato de punto flotante a entero y viceversa, esta posibilidad se da debido a que los apuntadores de memoria RAM, por ejemplo, deben hacerse en formato entero, además, para transferir resultados de diferentes procesos dentro del microcontrolador a la computadora es recomendable hacerlo en dato tipo entero por la estabilidad que implica esta instrucción. Para convertir un dato tipo entero a punto flotante se utiliza la instrucción “ $R_i = \text{INT2FP } R_j$ ” donde R_i almacena el valor R_j en tipo flotante y éste último debería tener un formato tipo entero, para obtener un dato tipo entero a partir de un dato tipo punto flotante recurrimos a “ $R_j = \text{FP2INT } R_i$ ”, donde R_j almacena el dato R_i en formato tipo entero, cuando R_i tiene un formato tipo punto flotante [23].

Instrucción y Sintaxis Lenguaje D				
IF	R	>, <, >=, <=	R o N	
Código a realizar si la instrucción es cierta				
ENDIF				
Código a realizar repetidamente si la instrucción es cierta				
ENDWHILE				
FOR	N (Valor de inicio)	N (Valor de aumento)	N (Valor final)	
Código a realizar repetidamente				
ENDFOR				
DELAY				
R	=	N		

R	=	R	+	R o N
R	=	R	-	R o N
R	=	R	*	R o N
R	=	R	/	R o N
R	=	R o N	^	R o N
R	=	LN	R o N	
R	=	EXP	R o N	
R	=	SQRT	R o N	
R	=	SIN	R o N	
R	=	COS	R o N	
R	=	TAN	R o N	
R	=	CSC	R o N	
R	=	SEC	R o N	
R	=	COT	R o N	
R	=	SINH	R o N	
R	=	COSH	R o N	
R	=	TANH	R o N	
R	=	CSCH	R o N	
R	=	SECH	R o N	
R	=	COTH	R o N	
R	=	ARCSIN	R o N	
R	=	ARCCOS	R o N	
R	=	ARCTAN	R o N	
LRAM	DIRECCION	R		
ERAM	R	DIRECCION		
LPOR	P	R		
EPOR	R	P		

Tabla 5.1: Instrucciones disponibles para programación de microprocesador [23].

5.2. Compilador de lenguaje de programación

El microprocesador en cuestión tiene como recurso un compilador de lenguaje de programación, el cual posee, de forma general, instrucciones de alto nivel con las que se puede describir operaciones dentro de la unidad aritmética lógica (ALU, por sus siglas en inglés) mismas que se describen en la tabla 5.1, sin embargo, es necesario

el lenguaje máquina compilado en la computadora, este lenguaje en formato hexadecimal (o máquina) incluye información relacionada al control de cada grado de libertad, lectura y escritura de direcciones de memoria RAM, algoritmos de toma de decisión en función a diversos escenarios que pueden presentarse, entre otros. La pestaña consiste en un botón denominado “descargar” (recuadro “2”), el cual permite descargar el lenguaje máquina previamente descrito, un indicador de grabado (recuadro “2”), es decir, cómo se comporta la transferencia de información de la computadora al robot, dos vectores de depuración de transferencia de datos (recuadro “1”) y un cuadro de información utilizado para indicar posibles errores durante la ejecución del software (recuadro “3”) [20] [23].



Figura 5.3: Interfaz de usuario, transferencia de software base [20].

El software recibido implementa una pestaña (Figura 5.4, recuadro “B”) que tiene como objetivo transferir datos que son exclusivos para manejar en la memoria RAM, dichos datos tienen capacidad de ser leídos por el procesador e implementar el sistema de control acorde a los parámetros escritos en los mismos. Los recursos disponibles son el botón de descarga que da acceso a la tabla de datos por manipular (recuadro 8), indicador de descarga en memoria RAM (recuadro 9), selección de dirección de memoria con el fin de indicar al procesador a partir de qué posición es necesario almacenar la información (recuadro 7), vectores de datos con fin de depuración (recuadro 6) e indicador de dimensión de tabla de datos por transferir (recuadro 10).



Figura 5.4: Transferencia de variables a memoria RAM [20].

La pestaña de ejecución de programa se ilustra en la Figura 5.5 (recuadro “C”), dicha pestaña tiene 3 subpestañas adicionales (C1, C2 y C3). La primera subpestaña (C1) permite enviar diversos parámetros de configuración, dichos datos son enviados a memoria RAM, para fines de depuración hemos decidido transferir parámetros de ganancias de los lazos de control correspondientes a cada grado de libertad, se transfieren variables de posición deseada, ganancia K_p , K_v , K_s y compensación gravitacional tanto para articulación q_1, q_2, q_3 (recuadro “14”). Adicionalmente tenemos el botón de inicio (recuadro “12”) y detenimiento de ejecución de software (recuadro “13”). Adicionalmente se ha implementado un indicador de ejecución del microprocesador (recuadro “11”) esto sólo con fines de monitoreo.

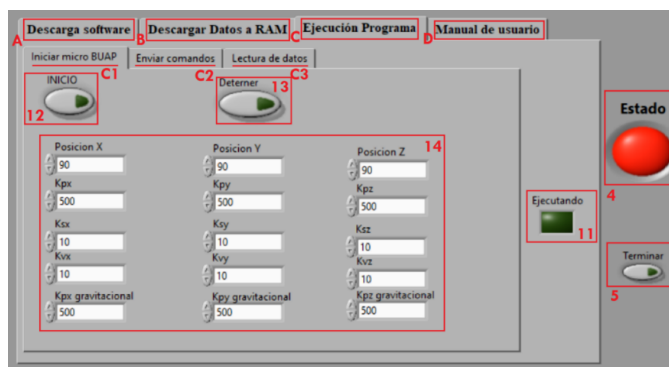


Figura 5.5: Transferencia de parámetros de sistema [20].

La subpestaña enviar comandos, subrayado “C2” (Figura 5.6) es similar a la de parámetros de sistema, con la diferencia de que dicha pestaña está destinada a transferir parámetros en tiempo real, es decir, actualizar algunos de los parámetros enviados al inicio del software. Se pretende enviar un comando tipo entero de 5 Bits (Recuadro “15”) en conjunto de un dato (denominado “Alfa deseada”), tanto los comandos como datos son dinámicos acorde al algoritmo a ejecutar en el procesador, se cree que es posible direccionar los datos mientras se ejecuta el procesador y actualizar variables como ganancias o posiciones deseadas, según corresponda, y este será un elemento a abordar dentro del trabajo de tesis con el fin de depurar el sistema de control. Se mantiene el implemento de un botón de envío (recuadro “16”) y también es posible visualizar los datos que se reciben en el puerto de comunicación WIFI, esto con la finalidad de monitoreo de alguna respuesta del controlador (recuadro “17”) [23].

La última subpestaña denominada Adquisición de datos (C3) (Figura 5.7) permite visualizar los datos recibidos en la computadora (recuadro “18”), los datos recibidos son programados acorde a las necesidades del usuario en el software descargado al procesador, estos van desde flancos de operación, posiciones articulares, hasta intervalos de tiempo de ejecución del procesador. Adicionalmente se ha desarrollado la función de almacenar los datos recibidos (recuadro “19”), consiste en leer cada dato que recibe y generar un archivo de texto con extensión “.DAT”, cada línea de dato corresponde a una lectura hecha por la interfaz, la restricción consiste en transferir los datos del procesador en formato entero y posteriormente estos se visualizan en formato flotante en la computadora, por lo que es necesario hacer un

acondicionamiento de la información para su legibilidad adecuada.

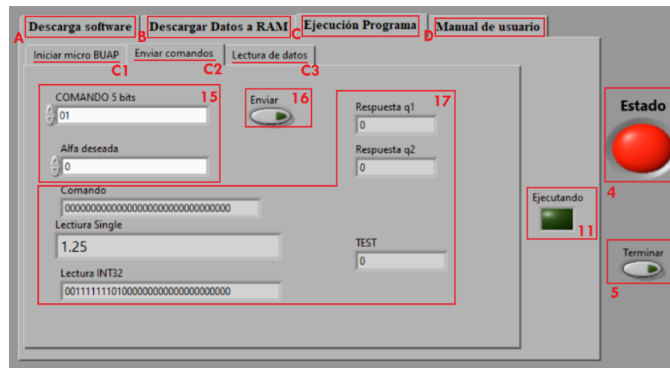


Figura 5.6: Pestaña para envío de comandos a microprocesador [20].

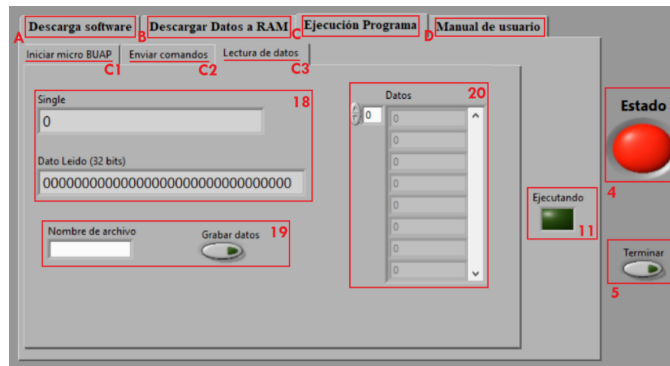


Figura 5.7: Pestaña de software destinada a adquisición de datos [20].

5.4. Conclusión

El desarrollo del software para el péndulo invertido sobre base móvil estableció una herramienta fundamental para la configuración, operación y monitoreo eficiente del sistema. El uso del lenguaje de programación D permitió aprovechar su flexibilidad y rendimiento en el control del microprocesador, facilitando la implementación de algoritmos optimizados y de bajo nivel. La elección del compilador adecuado garantizó la compatibilidad y estabilidad del código, maximizando el desempeño del sistema embebido.

Además, la interfaz de usuario diseñada ofreció una interacción intuitiva y accesible, permitiendo al usuario configurar parámetros, cargar programas al microcontrolador y supervisar en tiempo real las variables del sistema. Esta interfaz no solo mejora la experiencia del usuario, sino que también reduce el margen de error en las operaciones del sistema al proporcionar visualización clara de datos y opciones de control.

En conjunto, el software desarrollado cumple un papel esencial en la integración y funcionalidad del sistema, asegurando una operación fluida, precisa y adaptable. Este enfoque holístico refuerza la capacidad del sistema para abordar tareas complejas y sienta las bases para futuras expansiones y aplicaciones en entornos más exigentes.

Capítulo 6

Resultados experimentales

Luego de ensamblar todos los componentes y realizar las respectivas conexiones, se prosiguió con la realización de las pruebas experimentales, en donde lo primero que se realizó fue un código en lenguaje D en donde se describieron los controladores mencionados anteriormente, el ON-OFF, el controlador de frenado y el de estabilización, además de las secuencias de lectura de las señales y la captura de datos de las variables.

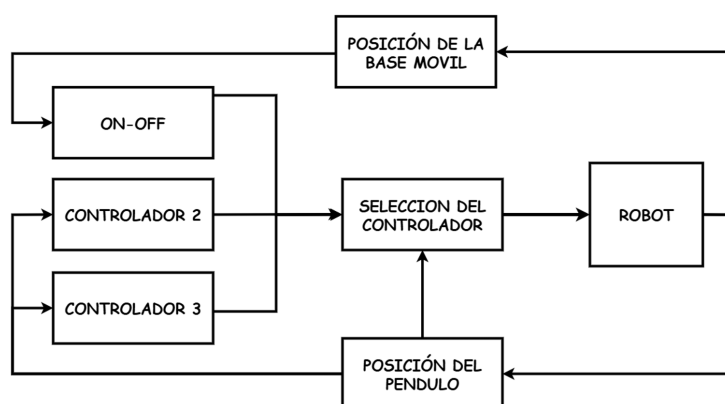


Figura 6.1: Diagrama del sistema de control.

Recordando que el primer controlador consiste en una señal del tipo ON-OFF, donde se define que la base móvil debe oscilar entre el rango de 0.14 m y 0.38 m, para lograr elevar al péndulo a la posición superior.

Para el segundo controlador tenemos que su rango de operación se encuentra entre $(145^\circ, 165^\circ)$ y $(-145^\circ, -165^\circ)$, teniendo la siguiente estructura.

$$\tau = K_{p1} \tanh(e), \quad (6.1)$$

en donde, τ es el torque del controlador, K_p es la ganancia proporcional y e es el error de posición definido como $e = x_d - x_c$.

Al realizar las pruebas se sintonizó la ganancia K_{p1} en donde encontramos que el valor donde funcionaba era $K_{p1} = 0.25$, lo que nos dice que solo utilizamos hasta el 25 % del torque disponible en este controlador y es suficiente.

6.1. Controlador P

El desarrollo del software para el péndulo invertido sobre base móvil estableció una herramienta fundamental para la configuración, operación y monitoreo eficiente del sistema. Ahora en el caso del tercer controlador, se definieron dos controladores a implementar, siendo el primero un control proporcional de la forma:

$$\tau = K_{p_2} \tanh(-e), \quad (6.2)$$

en donde: τ es el torque del controlador, K_{p_2} es la ganancia proporcional y e es el error de posición definido como $e = q_d - q_p$.

Para este caso se sintonizó la ganancia proporcional, obteniendo que $K_{p_2} = 0.3$, lo que nos indica que el torque máximo entregado por el controlador para esta prueba fuera solo del 30 % del torque total de nuestro motor. Una vez obtenido estos datos se comenzó con la obtención de los resultados experimentales con el controlador proporcional, los cuales se presentan a continuación.

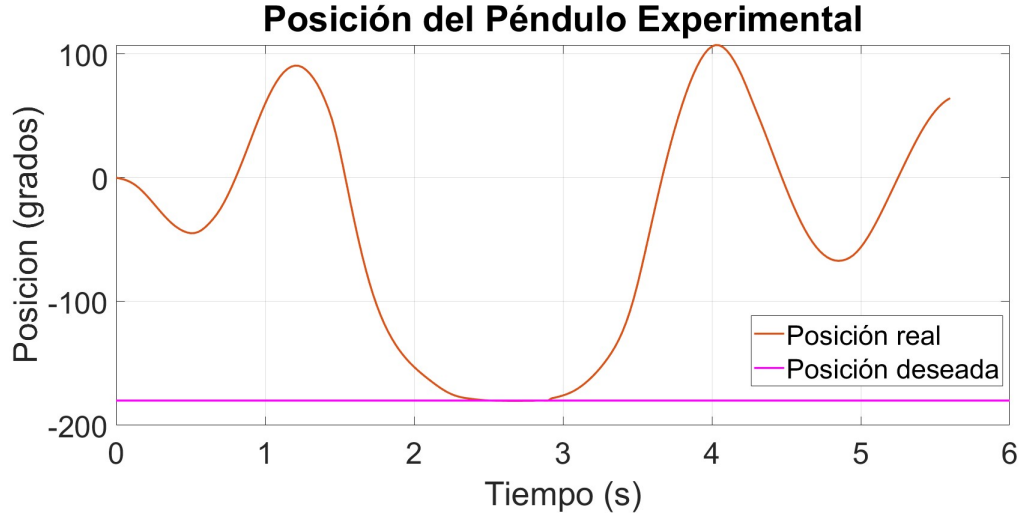


Figura 6.2: Gráfica de posición del péndulo con el controlador P.

En la Figura 6.2 se puede observar el movimiento del péndulo, donde observamos que la estrategia de control implementada logra estabilizar por un momento al péndulo en la posición deseada, la cual logra mantener por un momento debido a que la base móvil llega al fin de carrera y se detiene, provocando que el péndulo vuelva a caer.

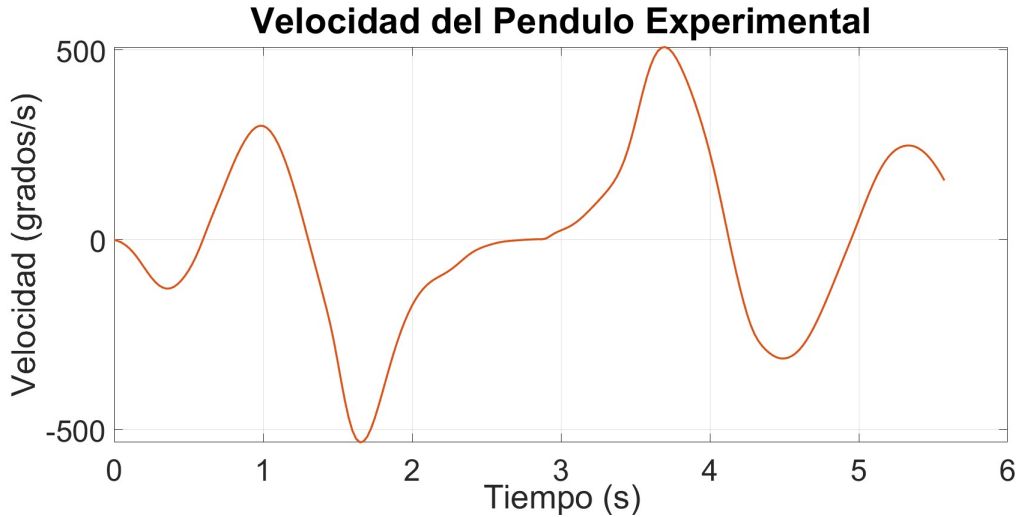


Figura 6.3: Gráfica de velocidad del péndulo con el controlador P.

En la Figura 6.3 se puede observar la velocidad que toma el péndulo durante su movimiento nótese como la velocidad se mantiene casi en 0 mientras el péndulo estaba estable, lo que nos indica que estuvo casi quieto en ese momento.

6.2. Controlador PD

Ahora para el segundo controlador, se utilizó un PD de la forma:

$$\tau = K_{p3} \tanh(-e_p) + K_v \tanh(e_v), \quad (6.3)$$

en donde, τ es el torque del controlador, K_{p3} es la ganancia proporcional, e_p es el error de posición definido como $e_p = q_d - q_p$, K_v es la ganancia derivativa y e_v es el error de posición definido como $e_v = v_d - v_p$.

Para este caso se sintonizó nuevamente la ganancia proporcional y ahora también la derivativa, obteniendo de las pruebas experimentales, que el valor adecuado de estas ganancias son $K_{p3} = 0.3$ y $K_v = 0.05$, lo que nos indica que el torque máximo entregado por el controlador es el 35 % del torque total de nuestro motor, con ello se prosiguió a realizar la captura de los datos de la posición del péndulo, en donde se obtuvieron los siguientes resultados.



Figura 6.4: Gráfica de posición del péndulo con el controlador PD.

En la Figura 6.4 se puede observar el movimiento del péndulo, donde se puede observar que la estrategia de control implementada es capaz de estabilizar al péndulo en la posición deseada, lo que también nos indica que las ganancias fueron seleccionadas adecuadamente, de igual forma debido a que la base móvil llega al fin de carrera y se detiene, el péndulo también vuelve a caer.

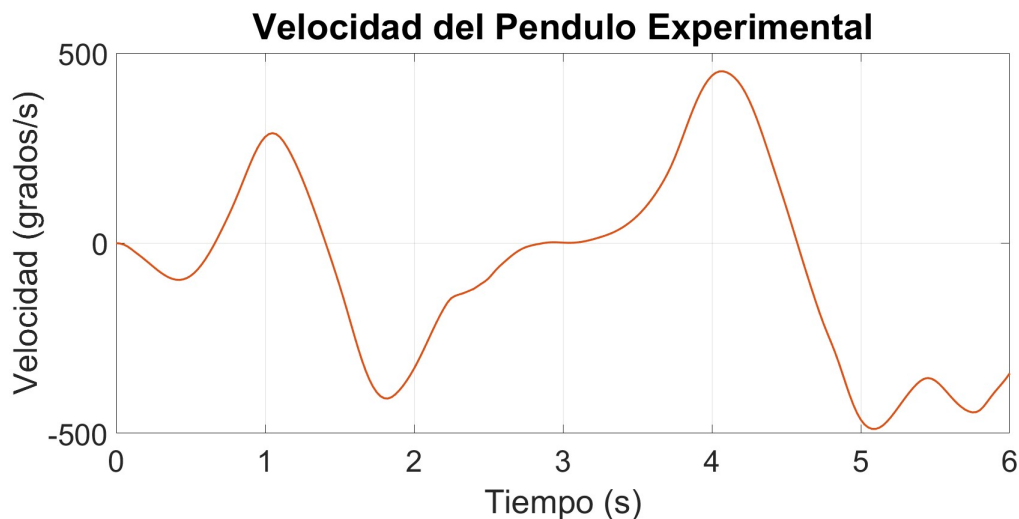


Figura 6.5: Gráfica de velocidad del péndulo con el controlador PD.

En la Figura 6.5 se puede observar la velocidad que toma el péndulo durante su movimiento en este caso se observa un movimiento más suave al momento de estabilizar, y como pudo llegar a quedarse quieto, pero vuelve a llegar la base móvil al fin de carrera y vuelve a perder el control del péndulo. Una vez obtenidas estas graficas se prosigue a compararlas con los resultados obtenidos de las simulaciones del modelo dinámico, estas comparaciones se presentan a continuación.

6.2.1. Métrica de evaluación para comparar datos

La métrica utilizada para evaluar la similitud entre los datos obtenidos de la simulación y los datos experimentales que esta descrita a continuación, primero se evalúa la similitud en el dominio del tiempo y se guarda en la variable a, después se obtiene el porcentaje de similitud de las dos señales, donde retorna el 100 % si son idénticas, lo siguiente es el cálculo del espectro de frecuencia utilizando la transformada discreta del coseno para determinar la similitud en frecuencia de ambas señales que se guarda en la variable b, por ultimo se multiplican estos dos porcentajes y se ajustan con una potencia de 0.4 para equilibrar los resultados de ambas similitudes.

$$a = 1 - \frac{\left| \sum_{i=1}^N x_i - \sum_{i=1}^N y_i \right|}{\left| \sum_{i=1}^N (|x_i| + |y_i|) \right|}, \quad (6.4)$$

$$X = \text{dct}_{II}(x), \quad (6.5)$$

$$Y = \text{dct}_{II}(y), \quad (6.6)$$

$$X_n = \frac{X}{\text{máx}(X)}, \quad (6.7)$$

$$Y_n = \frac{Y}{\text{máx}(Y)}, \quad (6.8)$$

$$b = 1 - \frac{\left| \sum_{i=1}^N |X_{n_i} - Y_{n_i}| \right|}{\left| \sum_{i=1}^N (|X_{n_i}| + |Y_{n_i}|) \right|}, \quad (6.9)$$

$$\text{metr2ev} = (ab)^{0.4}. \quad (6.10)$$

6.2.2. Comparación del controlador P

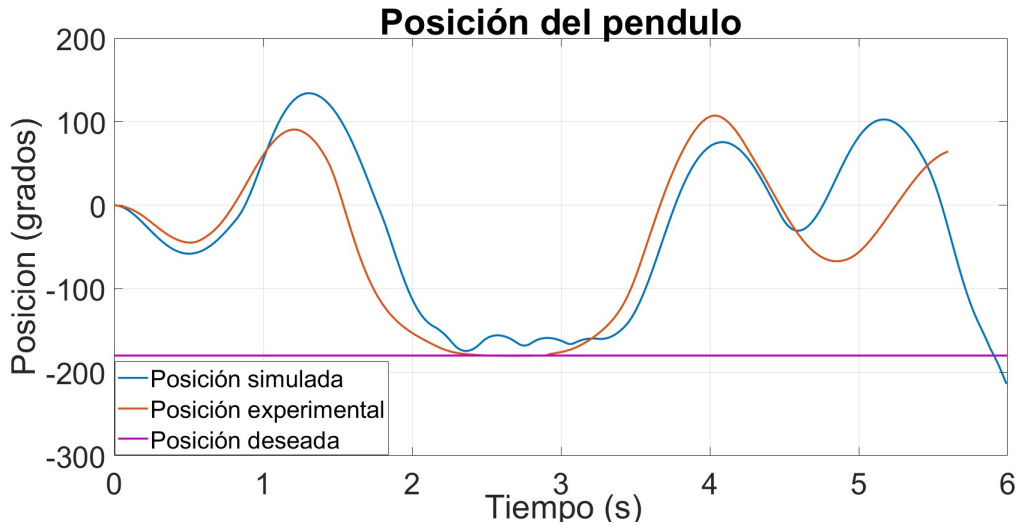


Figura 6.6: Comparación de la posición del péndulo experimental y simulada del controlador P.

En la Figura 6.6 se observa la posición del péndulo simulada y experimental, en donde podemos ver que son muy parecidas, se aprecia que al principio la oscilación de la simulación es mayor que la real y al momento de llegar a la posición deseada la simulación tiene una mayor oscilación que la real, y cuando cae, ahora la real es mayor que la simulación, estas diferencias se pueden deber a varios factores, entre los cuales se encuentra que algunos de los parámetros del sistema para realizar la simulación fueron calculadas por aproximación y no son exactamente las reales del sistema pero son muy próximas, además de otros efectos que podrían existir y no fueron modelados, utilizando la métrica de la transformada discreta del coseno, la cual fue descrita anteriormente, se obtuvo que la relación que existe entre la posición simulada y la experimental en este caso es del 78.7 %.

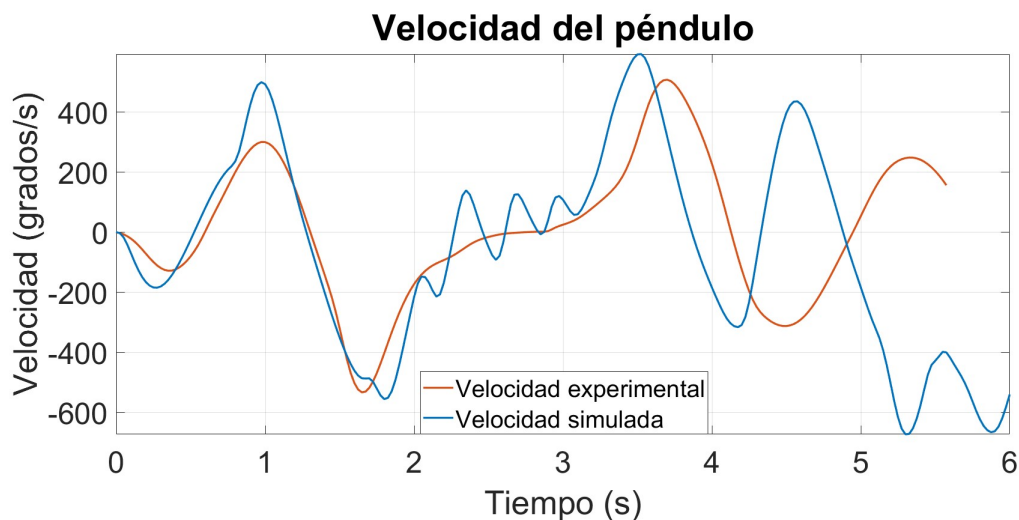


Figura 6.7: Comparación de la velocidad del péndulo experimental y simulada del controlador P.

En la Figura 6.7 se observa la comparación de la velocidad simulada y la real del péndulo, aquí se aprecia más claramente como la simulación tienen una mayor oscilación que la real al momento de llegar a la posición deseada, como mencionamos anteriormente, además que ambas son muy parecidas, también se pueden ver los momentos en los que la base móvil cambia de sentido en la simulación ya que se ven los impulsos que genera en el péndulo, en el caso de la experimental estos impulsos no son visibles esto podría deberse a los efectos no modelados existentes en el mecanismo, y siguiendo la misma métrica que en la anterior se obtuvo que en este caso la coincidencia entre las dos señales es del 67.07 %.

6.2.3. Comparación del controlador PD

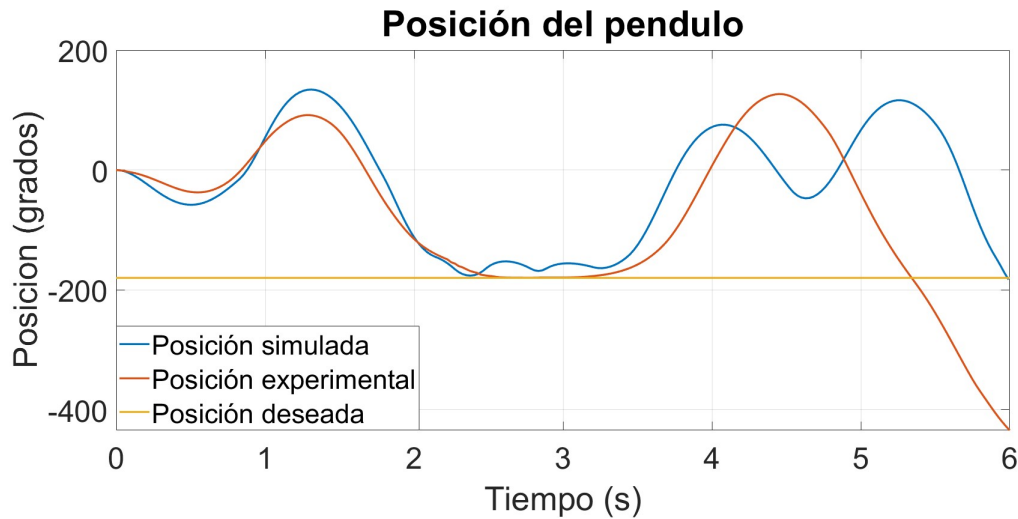


Figura 6.8: Comparación de la posición del péndulo experimental y simulada del controlador PD.

En el caso de la Figura 6.8 se observa la comparación de la posición simulada y real del péndulo, en donde podemos ver que son muy parecidas, siendo un poco más diferente al momento que el péndulo vuelve a salirse del punto de equilibrio, igual se aprecia que el péndulo llega a la posición deseada un poco más suave que con el controlador anterior. Aquí se nota más la necesidad de tener una herramienta que nos permita obtener los valores de los parámetros del sistema, por último, utilizando nuevamente la transformada discreta del coseno se obtuvo que en este caso la coincidencia entre las dos señales es del 79.76 %.

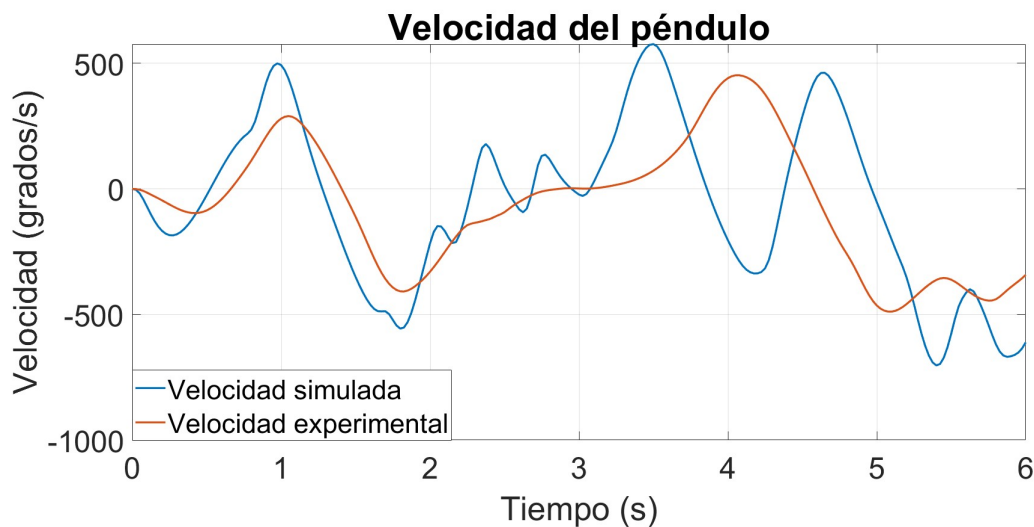


Figura 6.9: Comparación de la velocidad del péndulo experimental y simulada del controlador PD.

En la Figura 6.9, se observa la comparación de la velocidad simulada y la real del péndulo, aquí se aprecia más claramente como la simulación tienen una mayor oscilación que la real, como observamos anteriormente, lo cual puede ser debido a que no se consideraron más efectos que existen en el sistema real, y siguiendo la misma métrica que en la anterior se obtuvo que en este caso la coincidencia entre las dos señales es del 72.9 %.

A continuación, se presenta una tabla comparativa de ambos controladores.

Tabla 6.1: Valores de los parámetros

Controlador	Porcentaje de coincidencia	
	Posición	Velocidad
P	78.7 %	67.07 %
PD	79.76 %	72.9 %

Como podemos observar, Tabla 6.1, el controlador PD es el que tuvo una mejor coincidencia que el P, aun que esta diferencia es pequeña en la posición, en el caso de la velocidad tiene una mejor respuesta.

6.3. Conclusión

Realizar la implementación experimental del sistema fue una tarea un poco complicada debido a algunas restricciones que existen al momento de realizar el código en lenguaje D, aún así se lograron obtener los resultados esperados, al comparar los resultados experimentales con las simulaciones logramos obtener una coincidencia muy buena comparado con otros trabajos relacionados con el péndulo invertido sobre base móvil, como por ejemplo el caso de la tesis de maestría titulada desarrollo de un sistema mecatrónico tipo péndulo invertido sobre base móvil [15], en donde la coincidencia es alrededor del 35 %, con esto podemos reafirmar lo que otros compañeros han encontrado en otros robots, como el caso del robot cartesiano [38] [39], que al seguir la metodología presentada para la obtención del modelo dinámico esta nos logra ofrecer una mejor coincidencia entre lo simulado y lo real. Esta primera aproximación nos ha dado resultados muy buenos y aún es posible mejorarlo al incluir más fenómenos como la fricción y al hacer una identificación paramétrica con redes neuronales y utilizando un control de trayectoria se espera que el porcentaje de coincidencia logre alcanzar sea mayor al 90 %.

Conclusiones Generales

El desarrollo de este trabajo permitió actualizar de manera integral el modelado, la electrónica y el sistema de control de un péndulo invertido sobre base móvil. La caracterización de los motores permitió obtener una respuesta casi lineal del torque en función del ancho de pulso, contribuyendo a una operación más predecible y estable del sistema.

La actualización completa del sistema electrónico, incluyendo módulos de potencia, dispositivos de adquisición y acondicionamiento de señales, mejoró significativamente la confiabilidad en la toma de datos y la capacidad de respuesta del robot en tiempo real.

El modelo dinámico obtenido mediante la metodología de parámetros agrupados presentó una coincidencia del 79 % con los datos experimentales, a pesar de no incluir la fricción. Esta aproximación se considera muy buena y abre la posibilidad de futuras mejoras mediante la incorporación de fricción y la identificación paramétrica, lo que permitiría instrumentar un control de trayectoria más preciso.

El análisis comparativo de los controladores implementados mostró un desempeño ligeramente superior (1 %) del controlador proporcional-derivativo sobre el proporcional, lo que respalda su idoneidad para la estabilización del péndulo invertido.

En conjunto, este trabajo ofrece una plataforma experimental robusta y versátil para el estudio de sistemas dinámicos, integrando mecánica, electrónica y software. Además, sienta las bases para futuras investigaciones y aplicaciones prácticas, demostrando el valor de la integración interdisciplinaria en el desarrollo del péndulo invertido.

Bibliografía

- [1] J. L. Álvarez G y Y. Posadas V. “La obra de Galileo y la conformación del experimento en la física”. SciELO - Scientific Electronic Library Online. Accedido el 27 de noviembre de 2023. https://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S0035-001X2003000100011
- [2] Academia Lab. (2023). Reloj de péndulo. Enciclopedia. Revisado el 27 de noviembre del 2023. <https://academia-lab.com/enciclopedia/reloj-de-pendulo/>
- [3] A. Dupré, P. Janssen, An accurate determination of the acceleration of gravity g in the undergraduate laboratory. Am. J. Phys. 68 (8) August 2000, pp. 704–711.
- [4] “El péndulo cónico”. Campus de Gipuzkoa - UPV/EHU. Accedido el 27 de noviembre de 2023. <http://www.sc.ehu.es/sbweb/fisica/dinamica/trabajo/conico/conico.htm>
- [5] K. J. Åström, K. Furuta. Swinging up a pendulum by energy control. Automatica 36(2):287–295, 2000. doi:10.1016/S0005-1098(99)00140-5
- [6] L. Wang, H. Ni, W. Zhou, P. M. Pardalos, J. Fang, M. Fei. MBPOA-based LQR controller and its application to the double-parallel inverted pendulum system. Engineering Applications of Artificial Intelligence [en línea]. 36 (2014) 262–268. Disponible en: https://www.researchgate.net/deref/http://dx.doi.org/10.1016/j.engappai.2014.07.023?_tp=eyJjb250ZXh0Ijp7ImZpcnN0UGFnZSI6InB1YmxpY2F0aW9uIiwicGFnZSI6InB1YmxpY2F0aW9uIn19
- [7] “Ninebot by Segway”. amazon.com. Accedido el 27 de noviembre de 2023. https://www.amazon.com.mx/Patineta-Autoequilibrio-Bluetooth-Port%C3%A1til-silencioso/dp/B07LFD7FMF/ref=asc_df_B07LFD7FMF
- [8] Twenergy. “Tipos de patinetes eléctricos”. Twenergy. Accedido el 27 de noviembre de 2023. <https://twenergy.com/patinetes-electricos/tipos-de-patinetes-electricos/>
- [9] aliexpress. “Monociclo eléctrico inteligente para adultos, monopatín de equilibrio de una sola rueda”. aliexpress.com. Accedido el 27 de noviembre de 2023. <https://es.aliexpress.com/item/1005004491747977.html>
- [10] Otero. “Onewheel, un monopatín eléctrico, con una única gran rueda, auto-estabilizado”. Motorpasión. Accedido el 27 de noviembre de 2023. <https://www.>

motorpasion.com/coches-hibridos-alternativos/onewheel-un-monopatin-electrico-con-una-unica-gran-rueda-auto-estabilizado

- [11] LeaperKim. “Monociclo eléctrico Veteran Sherman S”. aliexpress.com. Accedido el 27 de noviembre de 2023. <https://es.aliexpress.com/item/1005005296837148.html>
- [12] “MONOPATÍN ELÉCTRICO MONORUEDA - VEHÍCULOS CON INGENIO”. Accedido el 27 de noviembre de 2023. <https://vehiculosconingenio.com/producto/monopatin-electrico-monorueda/>
- [13] H. Muhammad et al. Real-Time Control of an Inverted Pendulum: A Comparative Study. Proceedings - 2011 9th International Conference on Frontiers of Information Technology, FIT 2011. 183–188. 10.1109/FIT.2011.41
- [14] B. Dayakar, “Stabilization of Inverted pendulum Using Sliding Mode control”, tesis, 2016. Accedido el 27 de noviembre de 2023. http://ethesis.nitrkl.ac.in/9114/1/2016_MT_BDayakar.pdf
- [15] M. Ramos Ortega, “Desarrollo de un sistema mecatrónico tipo péndulo invertido sobre base móvil.”, Tesis de maestría, BUAP, Puebla, 2015. <https://hdl.handle.net/20.500.12371/12537>
- [16] Autonics, Datasheet encoder incremental series E50S, <http://www.farnell.com/datasheets/1673624.pdf>. Consultado en noviembre de 2023
- [17] R. Kelly, V. Santibañez, Robótica Automática. PEARSON EDUCACION. (2003)
- [18] B. Fabien, Analytical System Dynamics. Boston, MA: Springer US, 2009. <https://doi.org/10.1007/978-0-387-85605-6>
- [19] F. Coluccio Leskow. (2021) Fricción. <https://concepto.de/friccion/>
- [20] E. Pastor Torres, “Implementación y control de un emulador de cuello humano”, Tesis de maestría, BUAP, Puebla, 2020
- [21] L. A. Alcantar Vergara, “Modelado y control de un robot cartesiano de dos grados de libertad realizando estudios de jerk”, Tesis de maestría, BUAP, Puebla, 2023
- [22] C. K. Alexander, M. N. O. Sadiku. “Fundamentos de circuitos eléctricos 3 Edición”. Mc Graw Hill, 2006
- [23] D. M. González Arriaga et al., “Lenguaje D” Programas de computación, Registro Público del Derecho de Autor, n^o 03-2021-120110454200-01, 2021
- [24] N. F. Figueroa-Mosquera, «Android Application USB-Serial datalogger for embedded systems», Interfaces, Universidad de Lima, vol. 9, pp. 209–220, 2016

- [25] A. Choudhary, «FPGA Based Solution For Ethernet Controller As Alternative For TCP/UDP Software Stack,» IEEE Xplore Digital Library, 2018
- [26] A. Soetedjo, Y. I. Nakhoda y C. Saleh, «An Embedded Platform for Test-bed Implementation of Multi-Agent System in Building Energy Management System,» *Energies*, vol. 12, p. 3665, 2019
- [27] Intel, Altera, noviembre 2023. <https://www.intel.la/content/www/xl/es/products/programmable.html>. Último acceso: 27 noviembre de 2023
- [28] J. G. Proakis, D. G. Manolakis, “Tratamiento digital de señales 4 Edición”, Pearson Prentice Hall, 2007
- [29] C. O. J. Alexander et al., “Fundamentos y aplicación del muestreo en señales ubicadas en las bandas altas del espectro”, *Scientia Et Technica*, 2008, XIV(39), pp. 37–42. ISSN: 0122-1701. <https://www.redalyc.org/articulo.oa?id=84920503008>
- [30] López Gómez Jesús et al., (2020). Influence of PWM Torque Control Frequency in DC Motors by Means of an Optimum Design Method. *IEEE Access*. <https://doi.org/10.1109/ACCESS.2020.2990158>
- [31] R. L. Boylestad, *Dispositivos Electrónicos y teoría de Circuitos*. Pearson, 2013
- [32] “Fuentes conmutadas”. *Circuitos y Esquemas Electrónicos - Electrónica Fácil*. Accedido el 27 de noviembre de 2023. <https://www.electronicafacil.net/tutoriales/Fuentes-conmutadas.html>
- [33] B. E. Babcock, «The Design, Construction, and Use of a SMALL PRONY BRAKE,» *Gas Engine*, vol. 35, n^o 7, 2000
- [34] M. Chiumenti, *Momentos de Inercia: Problemas Resueltos*. Barcelona, 2012
- [35] Shigley, *Mechanical Engineering Design*, 6a ed. McGraw-Hill Publ. Co., 2001
- [36] E. F. Erazo Muñoz, “Diseño y simulación de un sistema mecánico subactuado: péndulo invertido traslacional configurable”, *Tesis de Licenciatura*, UNIV. AUTON. OCCIDENTE, Santiago de Cali, 2006
- [37] J. Herrmann Mendoza, “Control Robusto de Sistemas Sub-actuados”, *Tesis de maestría*, Centro Investig. Estud. Av. del Inst. Politéc. Nac., CDMX, 2021
- [38] F. R. Cortés, *Robótica: Control de Robots Manipuladores*, 1^a ed., Puebla: Alfaomega, 2011
- [39] M. G. Marin, “Desarrollo de una red neuronal convolucional para realizar la identificación paramétrica de cuatro parámetros de un robot cartesiano de tres grados de libertad”, *Tesis de maestría*, BUAP, Puebla, 2024
- [40] J. F. P. Michimani, “Desarrollo e instrumentación de un escáner 3D”, *Tesis de maestría*, BUAP, Puebla, 2024

Anexos



Actualización de la Electrónica de un Robot Subactuado Tipo Péndulo Invertido Instrumentado en un Microprocesador en Plataforma FPGA

Ing. Luis Francisco de la Rosa Pineda¹, Dr. Sergio Vergara Limon²,
Dra. María Aurora Diozcora Vargas Treviño³, Dr. Jesús López Gomez⁴ y Dr. Daniel Marcelo González Arriaga⁵

Resumen— En este artículo se presenta la actualización de la electrónica desarrollada para un robot subactuado del tipo péndulo invertido sobre base móvil. Se detalla el diseño de la etapa de potencia, destinada a la activación del actuador principal, así como la integración y procesamiento de las señales provenientes de los sensores utilizados en el sistema de control. La solución electrónica ha sido diseñada para su implementación en una FPGA Cyclone V, previamente configurada con un microprocesador integrado para ejecutar las tareas de control requeridas por el robot.

Palabras clave— Péndulo Invertido, Robot Subactuado, Microprocesador, FPGA.

Introducción

En la literatura científica se muestran diversos trabajos relacionados con el modelo Péndulo Invertido, y se usa ampliamente hoy en día, por ello se ha convertido en un desafío de ingeniería común para los investigadores debido a la importante de aplicación en la vida práctica. Algunas de las aplicaciones actuales de los Péndulos Invertidos se presentan en los cohetes al lanzar, en el brazo robótico cuando se comporta como un péndulo invertido controlado, los sismómetros, robots humanoides, entre otros [L. Wang, H. Ni].

El sistema mecatrónico tipo péndulo invertido está compuesto por una serie de elementos como lo son una estructura mecánica, un sistema sensorial, un sistema de adquisición de datos y un ordenador, además de esto tiene que contar con un software que gestione el área sensorial y el movimiento de la estructura mecánica. El propósito final es el de conservar el péndulo perpendicular ante la presencia de perturbaciones, así el péndulo inclinado regresa a la posición vertical cuando al carro se le aplica una fuerza de control apropiada. Para mover el carro este se conecta mediante una cita de transmisión al motor que proporcionará un par proporcional a la tensión aplicada. El lazo de control se cierra mediante la lectura de la posición del carro y del péndulo con encoder. Al final se regresa el carro a una posición inicial de referencia.

Descripción del Método

Diseño mecánico del robot

El péndulo invertido sobre base móvil consta de un carro, el cual está formado por una caja hueca fabricado de aluminio con 4 llantas de Nylamid SL que reposan sobre el riel y cuatro llantas de Nylamid SL que lo sujetan por debajo del riel, cuenta con dos barras de aluminio, una a cada lado del carro con el propósito de mantener el equilibrio y estos puedan girar libremente alrededor del eje que las une al carro, al final de cada barra se encuentra una barra más delgada de aluminio con un peso de bronce en las puntas las cuales forman al péndulo, el carro se encuentra sobre una estructura fija construida en perfil de aluminio #21223 de 40x40 mm 6051, la base consta de dos barras, las cuales también funcionan como riel para el carro, y dos patas en los extremos que las sostienen, bajo el riel se encuentran dos bases donde se sitúan dos poleas que son parte de la transmisión que conectan el carro con el motor, una de las bases es fija y la otra es ajustable para poder tensar la banda dentada que se sitúa entre ellas y la cual está conectada al carro, el motor se sitúa a un costado junto con una caja de aluminio en la cual se encuentra un mecanismo de tornillo sinfín y corona que conecta al eje motor con la polea fija, el robot cuenta con dos encoder, uno de ellos se encuentra dentro de la base móvil, para poder medir la posición del péndulo, el otro se encuentra conectado al eje de la polea fija, este se utiliza para medir la posición del motor, también cuenta con dos finales de carrera, los cuales se sitúan

¹ Ing. Luis Francisco de la Rosa Pineda es alumno de la Maestría en Ciencias de la Electrónica Opción en Automatización en la Benemérita Universidad Autónoma de Puebla, Puebla, México. rp223470434@alm.buap.mx (autor corresponsal)

² El Dr. Sergio Vergara Limon es Profesor de la Maestría en Ciencias de la Electrónica Opción en Automatización en la Benemérita Universidad Autónoma de Puebla, Puebla, México. sergio.vergara@correo.buap.mx

³ La Dra. María Aurora Diozcora Vargas Treviño es Profesora de la Maestría en Ciencias de la Electrónica Opción en Automatización en la Benemérita Universidad Autónoma de Puebla, Puebla, México. aurora.vargas@correo.buap.mx

⁴ El Dr. Jesús López Gomez es Profesor de la Universidad Juárez Autónoma de Tabasco, Tabasco, México. jl06599@docente.ujat.mx

⁵ El Dr. Daniel Marcelo González Arriaga es Profesor de la Benemérita Universidad Autónoma de Puebla, Puebla, México. daniel.gonzalezarr@correo.buap.mx

debajo del riel para limitar la zona en la que el carro puede moverse libremente, como se muestra en la Figura 1 [M. RAMOS ORTEGA].

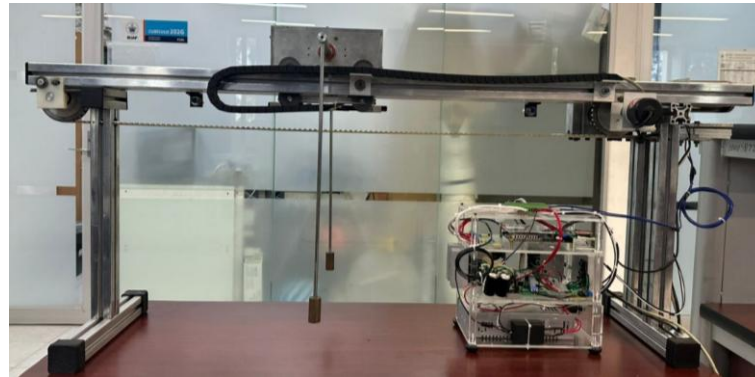


Figura 1. péndulo invertido sobre base móvil.

Actualización de la electrónica

El robot cuenta con sensores y actuadores que deben ser alimentados de manera adecuada para su correcto funcionamiento, así como es necesario realizar el acondicionamiento de las señales que nos entregan los sensores para acoplarlos a las terminales del FPGA que realizara su procesamiento.

Sistema embebido a implementar como lazo de control

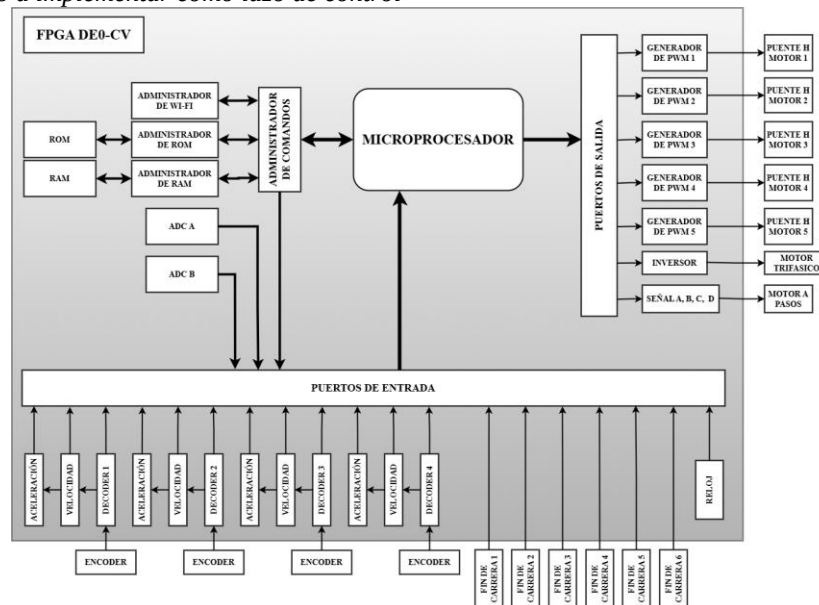


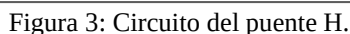
Figura 2: Diagrama general del FPGA.

El sistema del péndulo invertido se pretende trasladar a un nuevo procesador desarrollado en el laboratorio de Robótica y control de la MCEA. Este microprocesador está interconectado con “núcleos de propiedad intelectual” (IP Core, por sus siglas en inglés) a través del puerto de entrada y salida, con tareas para leer y escribir datos de propósito específico, sin embargo, como se ilustra en el diagrama de la Figura 2, el dispositivo recibido tiene recursos limitados, pero suficientes para implementar nuestro sistema embebido [D. M. González Arriaga].

Para el control del robot es necesario decodificar la señal de posición del motor y el péndulo, saber si el carro ha llegado a su fin de carrera, además de mandar la acción de control a través del motor con la señal PWM, para ello, de los recursos del sistema se utilizarán 2 Decoder, un generador de PWM y dos finales de carrera.

Puente H

Anteriormente el sistema contaba con un puente H construido con transistores MOSFET, el cual fue diseñado a partir de las características de la fuente de alimentación anterior, se decidió reemplazar todo el circuito por uno nuevo, el cual se diseñó utilizando transistores BJT, el diagrama de conexiones se presenta en la Figura 3.



Fines de carrera

Figura 4: Sensor inductivo y circuito de acoplamiento.

Estos sensores son de tipo PNP, entre sus características se encuentran que:

- Voltaje de alimentación es entre 5vdc-36vdc.
- Distancia de detección 4mm.

Debido a que su alimentación mínima es de 5vdc y los puertos del FPGA soportan solo 3.3vdc, se prosiguió a realizar un circuito divisor de voltaje para acoplar la señal de salida del sensor a la entrada de nuestra tarjeta.

Encoder

Para conocer algunos datos del sistema se utilizan sensores, los cuales son dispositivos que miden una variable física y la transforman en una señal eléctrica que es procesada por un sistema de adquisición de datos, el péndulo cuenta con dos encoder, el encoder es un transductor rotativo que transforma un movimiento angular en una serie de pulsos digitales, se pueden usar para controlar desplazamientos de tipo angular o de tipo lineal, uno de ellos se encuentra dentro del carro, el cual está conectado a las barras de aluminio de los lados para poder medir la posición del péndulo, el otro se encuentra conectado al eje de la polea fija, este se utiliza para medir la posición del motor, ambos sensores tienen la misma precisión.

El encoder del motor es de la marca a Autonics modelo E50S6-2500-6-I-5 ver Figura 10, el cual cuenta con las siguientes características [Autonics]:

- Voltaje de alimentación 5 Vdc.
- 6 señales de salida $A, \bar{A}, B, \bar{B}, Z, \bar{Z}$.
- 2500 pulsos por revolución.
- Salida Line Driver.
- 5 V de voltaje de alimentación



Figura 5. Encoder modelo E50S6-2500-6-1-5 [3].

Para el péndulo se utiliza un encoder de eje hueco de la empresa SICK modelo DFS60E-BHC02000 (ver Figura 6) con las siguientes características:

- Voltaje de alimentación 5 Vdc.
- 6 señales de salida $A, \bar{A}, B, \bar{B}, Z, \bar{Z}$.
- 2000 pulsos por revolución.
- Salida Line Driver.



Figura 6: Encoder modelo DFS60- BHC02000.

Debido a que la señal de salida de los encoders es igual a la de su alimentación y la alimentación mínima para su funcionamiento es de 5V entonces es necesario realizar un acoplamiento de la señal para reducir la señal de 5V a 3.3V por lo cual se tiene el siguiente circuito.

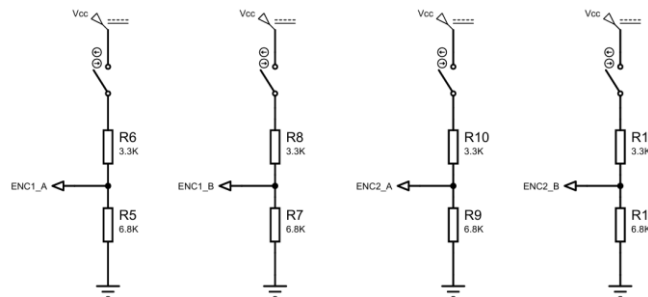


Figura 7: Acoplamiento de los encoders.

Resultados

Para verificar el funcionamiento del puente H se conectaron las terminales del motor al osciloscopio para observar el comportamiento de la conmutación de los transistores. En la figura 8, se puede observar cómo al aplicar una señal PWM a las entradas de control, estas se activan y desactivan de manera correcta entregando la energía requerida al motor.

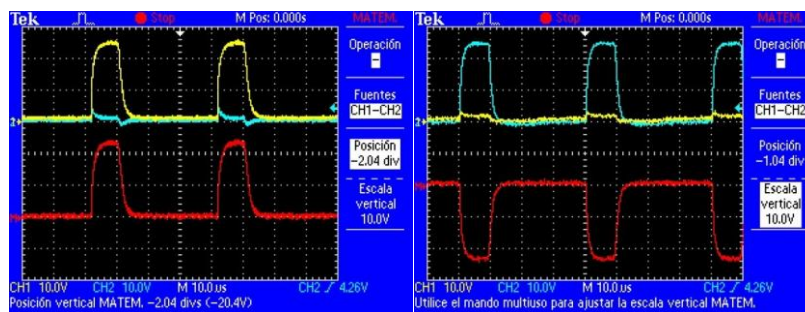


Figura 8: comportamiento del puente H.

Para los fines de carrera se realizó el impreso del circuito en una placa fenólica para después realizar las pruebas de su funcionamiento y asegurarnos que las salidas que irán conectadas al FPGA sean de 3.3 Vdc. En la figura 9, se pueden observar los fines de carrera de ambos lados ya colocados en el robot.

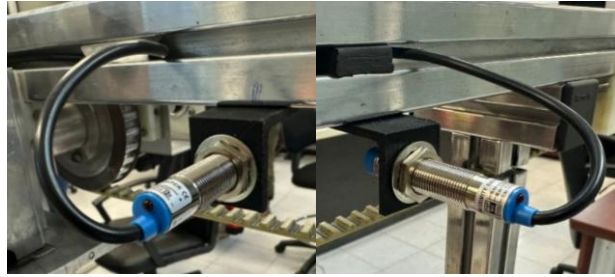


Figura 9: Fines de carrera inductivos.

Ahora para verificar las señales del encoder se realizó un código en lenguaje D, con el cual se observaron las cuentas por vuelta de cada encoder. En la figura 10, se puede observar como el encoder genera las dos señales de cuadratura y esas son las que ingresan al FPGA y en una interfaz de LabVIEW nos muestra las cuentas que lleva del encoder.

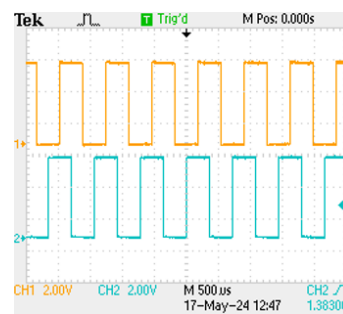


Figura 10: Lectura del encoder con el microprocesador.

Por último se concentró toda la parte electrónica en una estructura que fuera fácil de transportar, como se puede apreciar en la Figura 11.



Figura 11: Electrónica.

Para verificar el funcionamiento de la electrónica, se realizó como prueba un control de posición de la base móvil, el controlador utilizado tiene la siguiente forma:

$$\tau = k_p \tanh(k_s e)$$

donde:

τ es el torque entregado por el controlador

k_p es una ganancia proporcional

k_s es una ganancia de ajuste para el error

e es el error de posicion definido como $e = x_d - x_c$

En este caso se fijó una $x_d = 300mm$.

Una vez definido el algoritmo de control, se realizó la implementación en lenguaje D posteriormente se compiló para generar el archivo Hexadecimal que se carga al microcontrolador para obtener los datos experimentales.

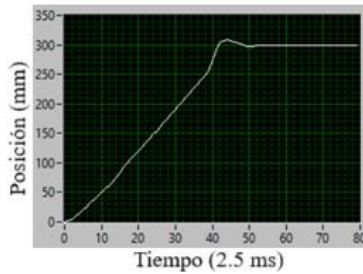


Figura 12: Posición de la base móvil.

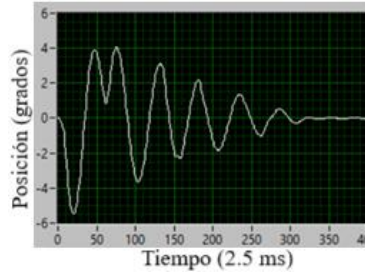


Figura 13: Posición del péndulo.

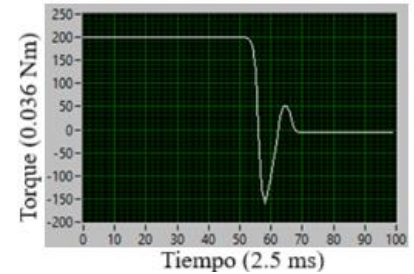


Figura 14: Torque entregado por el controlador.

En la figura 12 se puede observar la gráfica de la posición de la base móvil, en donde se aprecia que el controlador logra llevar a la base a la posición deseada. En la figura 13 se observa la posición del péndulo, la cual no se considera en el control, sin embargo, se obtuvieron las mediciones para verificar el funcionamiento del encoder. En la figura 14, se puede observar el torque entregado por el controlador al sistema, la cual esta presentada en el porcentaje del torque nominal del motor.

Conclusiones

La estructura mecánica del robot, en conjunto con el diseño de la nueva electrónica, garantiza una operación segura y altamente confiable del sistema. La electrónica de adquisición permite una medición precisa y continua de las variables del sistema, lo que nos permite un control eficiente. Por otro lado, la electrónica de potencia proporciona la energía necesaria para la operación del motor, asegurando un desempeño adecuado frente a las demandas operativas. Este sistema se integra con un firmware optimizado para el procesamiento de señales y la ejecución de algoritmos de control en tiempo real, lo que maximiza la autonomía y la precisión del sistema.

Referencias bibliográficas.

L. Wang, H. Ni, W. Zhou, P. M. Pardalos, J. Fang, M. Fei. MBPOA-based LQR controller and its application to the double-parallel inverted pendulum system. *Engineering Applications of Artificial Intelligence* [en línea]. 36 (2014) 262–268. Disponible en: https://www.researchgate.net/deref/http%3A%2F%2Fdx.doi.org%2F10.1016%2Fj.engappai.2014.07.023?_tp=eyJjb250ZXh0Ijp7ImZpcnN0UGFnZSI6InB1YmtpY2F0aW9uIiwicGFhZSI6InB1YmtpY2F0aW9uIn19

M. RAMOS ORTEGA, “DESARROLLO DE UN SISTEMA MECATRÓNICO TIPO PÉNDULO INVERTIDO SOBRE BASE MÓVIL.”, Tesis de maestría, Benemérita Universidad Autónoma de Puebla, Puebla, 2015. Accedido el 27 de noviembre de 2023. [En línea]. Disponible: <https://hdl.handle.net/20.500.12371/12537>

D. M. González Arriaga, M.A.D. Vargas Treviño, S. Vergara Limón “Lenguaje D” Programas de computación, Registro Público del Derecho de Autor, número de registro: 03-2021-120110454200-01, 2021.

Autonics, Datasheet encoder incremental series E50S <http://www.farnell.com/datasheets/1673624.pdf>. Consultado en noviembre de 2023.