
Plataformas de aprendizaje: control de un péndulo invertido

Cristian Anibal Catú Riera



UNIVERSIDAD DEL VALLE DE GUATEMALA
Facultad de Ingeniería




Plataformas de aprendizaje: control de un péndulo invertido

Trabajo de graduación en la modalidad de tesis presentado por Cristian
Anibal Catú Riera para optar al grado académico de Licenciado en
Ingeniería Mecatrónica


Guatemala,

2024

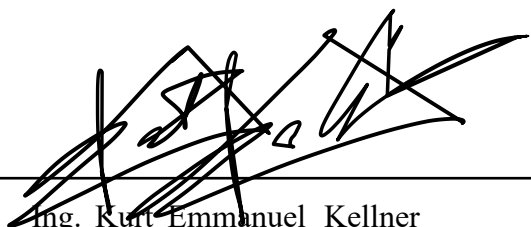
Vo.Bo.:

(f) 
M. Sc. Carlos Esquit

Tribunal Examinador:

(f) 
M.Sc. Carlos Esquit

(f) 
M. Sc. Miguel Enrique Zea Arenales

(f) 
Ing. Kurt Emmanuel Kellner

Fecha de aprobación: Guatemala, 13 de febrero de 2025.

Este trabajo representa la culminación de una etapa importante de mi vida, lo cual no se habría logrado sin el apoyo de varias personas especiales. Quiero expresar mi más profundo agradecimiento a quienes han estado a mi lado durante estos años de carrera, brindándome su apoyo y respaldo incondicional.

En primer lugar, a mis padres, por su amor y sacrificio, que han sido fundamentales para alcanzar mis metas. Su constante guía y confianza en mí han sido el pilar sobre el que he construido cada logro, y sin ellos, este proyecto no hubiera sido posible. También, agradezco a mi hermana que ha sido fuente de inspiración y ejemplo para no rendirme y saber que tengo la capacidad de dar mucho más.

Expreso mi más sincero agradecimiento a mis catedráticos Kurt Kellner, Luis Rivera, Miguel Zea y Pedro Castillo por su esfuerzo y dedicación a cada estudiante de la promoción de Ingeniería Mecatrónica.

Finalmente, y lo más importante, agradezco a Dios por ser mi fuente de esperanza y fortaleza en los momentos más difíciles de mi carrera. Él me ha dado el amor y la perseverancia necesaria para superar cada obstáculo, permitiéndome llegar a este punto con fe y humildad. Sin Su presencia en mi vida, este logro no habría sido posible.

Índice

Prefacio	III
Lista de figuras	VII
Resumen	VIII
Abstract	IX
1. Introducción	1
2. Antecedentes	2
2.1 Herramientas ejes 3D/Matlab para el control del sistema no lineal aplicado al péndulo invertido sobre carro deslizante - UPTC	2
2.2 Sistema de control PID analógico para péndulo invertido lineal - INTEC	3
2.3 Robot autónomo autobalanceado – Universidad Politécnica de Valencia	4
3. Justificación	6
4. Objetivos	7
4.0.1 Objetivo general	7
4.0.2 Objetivos específicos	7
5. Alcance	8
6. Marco teórico	9
6.1 Péndulo invertido	9
6.2 Modelo del sistema péndulo invertido	9
6.3 Sistemas de control para péndulo invertido	11
6.4 Diseño mecánico	12
6.5 Sensor Encoder magnético AS5600	14
7. Proceso de diseño mecánico y electrónico	15
7.1 Péndulo invertido - Versión 1	15
7.1.1 Diseño mecánico - Versión 1	15
7.1.2 Diseño electrónico - Versión 1	21
7.2 Péndulo invertido - Versión 2	23
7.2.1 Diseño mecánico - Versión 2	23
7.2.2 Diseño electrónico - Versión 2	28
7.3 Péndulo invertido - Versión 3	30

7.3.1	Diseño electrónico - Versión 3	30
8.	Proceso de fabricación	32
8.1	Fabricación mecánica	32
8.1.1	Fabricación desplazamiento lineal	32
8.1.2	Fabricación del péndulo - Versión 1	34
8.1.3	Fabricación del péndulo - Versión 2	37
8.2	Fabricación electrónica	39
8.2.1	Fabricación electrónica - Versión 1	39
8.2.2	Fabricación electrónica - Versión 2	40
9.	Técnicas de control para el sistema	41
9.1	Control clásico PID	41
9.1.1	Linealización del sistema	41
9.1.2	Implementación del controlador PID	43
9.2	Control moderno	44
9.2.1	Linealización del sistema	44
9.2.2	Implementación de controlador moderno	47
10.	Calibración de funcionamiento de control	49
10.1	Configuración del <i>Motor Stepper</i>	49
10.2	Configuración de sensores magnéticos	50
10.3	Calibración de control PID	50
10.4	Calibración de control con variables de estado	54
11.	Desarrollo de interfaz para control del péndulo	57
11.1	Especificaciones de interfaz	57
11.2	Desarrollo de interfaz	58
11.2.1	Pestaña de configuración	58
11.2.2	Pestaña de control PID	60
11.2.3	Pestaña de control por variables de estado	61
11.3	Retroalimentación de usuarios	62
12.	Conclusiones	64
13.	Recomendaciones	65
14.	Bibliografía	67
15.	Anexos	69

Lista de figuras

Figura 1. Variables de estado del sistema	2
Figura 2. Control PID aplicado al sistema linealizado	3
Figura 3. Péndulo invertido construido	4
Figura 4. Modelado del robot 3D	5
Figura 5. Diagrama de cuerpo libre	10
Figura 6. Diagrama de retroalimentación con PID	12
Figura 7. Diagrama de retroalimentación con variables de estado	12
Figura 8. Diagrama del proceso de diseño de Shigley	13
Figura 9. Sensor encoder magnético AS5600	14
Figura 10. Barras con cojinetes lineales	16
Figura 11. Perfil de aluminio (V-Slot)	16
Figura 12. Posicionamiento de plataforma	17
Figura 13. Acople de motor stepper	18
Figura 14. Acople polea lisa	18
Figura 15. Base para péndulo	19
Figura 16. Eje central	20
Figura 17. Masa del péndulo	21
Figura 18. Unión a faja	21
Figura 19. Tensionador de faja	24
Figura 20. Fijación de plataforma	24
Figura 21. Sujeción de sensor AS5600	25
Figura 22. Acople de motor	25
Figura 23. Final de carrera	26
Figura 24. Base del péndulo	26
Figura 25. Eje rotativo modificado	27
Figura 26. Masa modificada	27
Figura 27. Diseño de placa PCB	29
Figura 28. Placa PCB modelado 3D	30
Figura 29. Placa PCB v2 modelado 3D	31
Figura 30. Tensionador	33
Figura 31. Acople del motor en físico	33
Figura 32. Carro deslizante	34
Figura 33. Base del péndulo físico	34
Figura 34. Eje de rotación físico	35
Figura 35. Masa	36
Figura 36. Final de carrera	36
Figura 37. Primer prototipo	37
Figura 38. Base de péndulo versión final	38
Figura 39. Prototipo final	38

Figura 40.Placa PCB versión 1	39
Figura 41.Placa PCB versión final	40
Figura 42.Resultado 1 PID péndulo hacia abajo	51
Figura 43.Resultado 2 PID péndulo hacia abajo	52
Figura 44.Resultado 3 PID péndulo hacia abajo	52
Figura 45.Resultado 4 PID péndulo hacia arriba	53
Figura 46.Resultado 5 con variables de estado – péndulo hacia abajo	54
Figura 47.Resultado 6 con variables de estado – péndulo hacia abajo	55
Figura 48.Resultado 7 con variables de estado – péndulo hacia arriba	56
Figura 49.Pestaña de configuración	60
Figura 50.Pestaña control PID	61
Figura 51.Pestaña control variables de estado	62
Figura 52.Conexiones de driver TMC2130	69
Figura 53.Conexiones de sensores AS5600	70
Figura 54.Conexiones comunicación I2C	70
Figura 55.Conexiones de finales de carrera	70
Figura 56.Conexión a pin 3.3V de microcontrolador	71
Figura 57.Cálculo de ancho de pista	71
Figura 58.Cálculo de ancho de pista para alimentación de motor	71

El presente trabajo consistió en desarrollar una plataforma de aprendizaje de péndulo invertido, que permita implementar técnicas de control clásico y moderno. La implementación de esta plataforma en los cursos de Sistemas de Control 1 y Sistemas de Control 2 del Departamento de Ingeniería Electrónica, Mecatrónica y Biomédica de la Universidad del Valle de Guatemala permite un análisis y estudio más práctico de las técnicas de control en sistemas físicos, mejorando el desarrollo de las competencias sobre los principios básicos de control.

Para el desarrollo de la plataforma se seleccionaron componentes mecánicos y electrónicos que cumplen con los requerimientos para la dinámica sistema, de manera que el sistema se desempeñara de forma estable y confiable, con la menor fricción posible. El desplazamiento lineal del péndulo se basó en un perfil de aluminio con un carro deslizante, el cual se moviliza con una faja dentada conectada a un motor stepper Nema 17. Se implementó un encoder magnético AS5600 para realizar las lecturas de las variables de estado del sistema: ángulo del péndulo, velocidad angular, posición del carro y velocidad lineal. Los sistemas de control se desarrollaron en el microcontrolador NUCLEO-F446RE STM32, donde se implementó un PID y un control por variables de estado.

Se logró estabilizar el péndulo en dos posiciones distintas, colocado hacia abajo y hacia arriba. Se calibraron y modificaron los valores teóricos calculados para cada sistema de control. La calibración del control hacia arriba requirió de mayor precisión, ya que el sistema es muy inestable. Adicionalmente, se desarrolló una interfaz en Python para que el usuario pueda interactuar con el sistema, visualizar las variables de estado y modificar los parámetros del control en tiempo real.

Palabras clave: PID, variables de estado, péndulo, overshoot, desplazamiento, posición angular, velocidad angular, posición lineal, velocidad lineal.

This project involved the development of an inverted pendulum learning platform, designed to implement both classical and modern control techniques. The implementation of this platform in the Control Systems 1 and Control Systems 2 courses at the Department of Electronic, Mechatronics, and Biomedical Engineering at Universidad del Valle de Guatemala allows for a more practical analysis and study of control techniques in physical systems, enhancing the development of competencies in the fundamental principles of control.

For the platform's development, mechanical and electronic components were selected to meet the system's dynamic requirements, ensuring stable and reliable performance with minimal friction. The linear displacement of the pendulum was based on an aluminum rail with a sliding carriage, driven by a toothed belt connected to a Nema 17 stepper motor. An AS5600 magnetic encoder was implemented to measure the system's state variables: pendulum angle, angular velocity, carriage position, and linear velocity. The control systems were developed using the NUCLEO-F446RE STM32 microcontroller, where a PID and a state-variable control were implemented.

The pendulum was successfully stabilized in two different positions: facing down and facing up. The theoretical values calculated for each control system were calibrated and adjusted. The calibration for the upward control required greater precision, as the system is highly unstable. Additionally, a Python interface was developed to allow the user to interact with the system, visualize the state variables, and modify the control parameters in real-time.

Keywords: PID, state variables, pendulum, overshoot, displacement, angular position, angular velocity, linear position, linear velocity.

La aplicación y el desarrollo de los sistemas de control son fundamentales para la optimización de procesos de ingeniería. A través de la retroalimentación en tiempo real, los sistemas de control ajustan variables para estabilizar al sistema ante perturbaciones. En este contexto, el péndulo invertido es uno de los métodos más utilizados para la enseñanza y la investigación de sistemas de control, ya que es un sistema altamente inestable que requiere implementación de algoritmos de control para su estabilización. Este trabajo tiene como objetivo la implementación de una plataforma interactiva de péndulo invertido, diseñada para su uso en los laboratorios del Departamento de Ingeniería Electrónica, Mecatrónica y Biomédica de la Universidad del Valle de Guatemala.

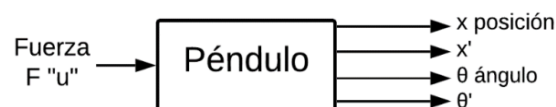
Para lograr esto se han diseñado y fabricado los componentes mecánicos y electrónicos del péndulo invertido, garantizando un sistema robusto y confiable para obtener una respuesta dinámica adecuada. Para diseñar el sistema de control fue importante analizar el modelo físico obteniendo la planta del sistema. Para el diseño del sistema de control, fue importante realizar un análisis detallado del modelo físico del péndulo invertido. Este análisis permitió obtener la planta del sistema, la cual sirvió como base para desarrollar los controladores. A partir de esta planta, se calcularon las constantes del controlador PID y las constantes del vector K , utilizando las matrices correspondientes. Estos cálculos proporcionaron un modelo de control teórico que posteriormente fue implementado y ajustado para optimizar la estabilidad del sistema.

El sistema de control fue desarrollado sobre el microcontrolador NUCLEO-F446RE STM32, utilizando control PID y por variables de estado, lo que permitió estabilizar el péndulo en dos posiciones: hacia abajo y hacia arriba. Finalmente, se desarrolló una interfaz con tkinter en Python, la cual permite una interacción en tiempo real con el péndulo invertido para modificar los parámetros de control y configuración. También facilita la visualización de las variables de estado en tiempo real. La interfaz se conforma en tres pestañas: la pestaña de configuración, control PID y control por variables de estado.

2.1. Herramientas ejes 3D/Matlab para el control del sistema no lineal aplicado al péndulo invertido sobre carro deslizante - UPTC

En la Universidad Pedagógica y Tecnológica de Colombia, se implementó una herramienta didáctica para un sistema de control de péndulo invertido de forma simulada. El problema que se buscó resolver es que el control de un modelo de péndulo invertido de forma física podría resultar un costoso y requerir mucho tiempo, por lo que el control realizado fue en un entorno de JAVA 3D. Esto ayudó a minimizar el tiempo y costos de la enseñanza en sistemas de control. Como el estudiante desea aplicar sus competencias de control, se diseñó una interfaz GUI para facilitar el aprendizaje de los conceptos básicos de un sistema de control de pregrado y posgrado. Se tomaron en cuenta dos aspectos principales, el “swing up” que consiste en llevar al péndulo en posición vertical y el “balanceo” que mantiene el péndulo en posición vertical. Se plantearon las ecuaciones y matrices no lineales con un diagrama de cuerpo libre y luego se linealizó el sistema entorno al punto de equilibrio no estable. Es importante notar que este sistema tiene cuatro variables de estado que son la posición, velocidad, ángulo del péndulo y su velocidad angular como se muestra en la Figura 1. Se hizo una realimentación por variables de estado para lograr estabilizar el sistema ubicando los polos en lazo cerrado en el lado izquierdo del plano complejo para obtener un vector de ganancias para cada variable de estado. [1]

Figura 1: Variables de estado del sistema



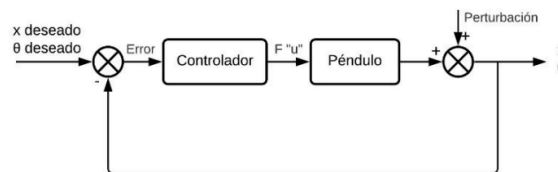
Adaptado de Osar Oswaldo (2012) [1]

Para visualizar de mejor manera el comportamiento del sistema, se muestran ventanas para las variables de estado usadas del prototipo tridimensional, inicialización y evolución del modelo, lo cual permite al usuario observar los resultados de control del péndulo simple simulado. Una de las limitaciones de este proyecto es que a pesar que la simulación e interfaz muestran de forma básica el control del péndulo invertido, las fricciones en el sistema real podrían variar. [1]

2.2. Sistema de control PID analógico para péndulo invertido lineal - INTEC

En el Instituto Tecnológico de Santo Domingo (INTEC) se implementó un modelo que no sólo presentó simulaciones matemáticas del péndulo invertido, sino también resultados físicos. A partir de las ecuaciones diferenciales del modelo no lineal del péndulo invertido se obtuvieron las matrices linealizadas evaluadas en el punto de control inestable. De esta manera se logró encontrar la planta del péndulo invertido lineal en una localidad cercana al punto inestable. Luego se aplicó un control PID, como se muestra en la Figura 2, con sus constantes proporcional, derivativa e integral respectivas que lograran estabilizar el sistema sin muchas oscilaciones y de forma eficiente.

Figura 2: Control PID aplicado al sistema linealizado



Adaptado de INTEC (2022) [2]

Para determinar las constantes de PID que estabilizan al sistema se trabajó con el *PID Tuner* de la plataforma de Matlab, el cual sugiere el control según los parámetros de respuesta y robustez deseados. Para un mejor análisis de la estabilidad del sistema, se utilizaron varias herramientas de Matlab como la *Control System Toolbox* o la *Signal Processing Toolbox*, las cuales proporcionan funciones y algoritmos para simular, analizar y ajustar sistemas de control lineal según el espacio de estados. En Simscape se simuló el péndulo invertido según los parámetros de control y los resultados también fueron exitosos.

Figura 3: *Péndulo invertido construido*



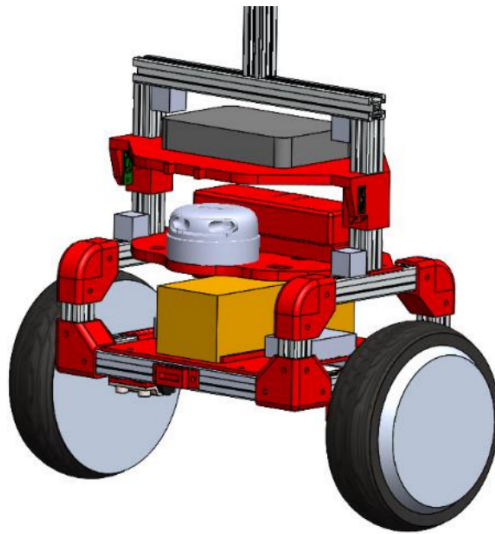
Adaptado de INTEC (2022) [2]

En el modelo físico construido que se muestra en la Figura 3, se implementaron los actuadores y sensores necesarios para poder ofrecer el control adecuado al sistema, tratando de mitigar las perturbaciones como la fricción o malas lecturas del sistema. Para poder movilizar el carro del péndulo se utilizaron cojinetes lineales, con varillas de acero, lo cual logró reducir significativamente la fricción del sistema. Para controlar la posición y avance del carro se utilizaron poleas dentadas conectadas a un motor DC que se controla con un puente H. Es importante mencionar que la lectura de la posición angular del péndulo se realizó con un potenciómetro, lo cual no permite una rotación completa del péndulo. Por lo tanto el sistema no cuenta con la parte del “swing up”, donde siempre debe comenzar cercano al punto de control deseado por las limitaciones del potenciómetro. El control realizado en este proyecto fue con un PID analógico el cual obtenía la lectura de la posición angular del péndulo en la localidad cercana al punto inestable, y luego con una retroalimentación negativa se controló el sistema. [2]

2.3. Robot autónomo autobalanceado – Universidad Politécnica de Valencia

Por otra parte, en la Universidad Politécnica de Valencia se implementó un sistema robótico tipo *segway* para un movimiento de forma autónoma. Sin embargo el problema que encontraron es que no hay una estructura y carenado para implementar los motores, placas electrónicas y sensores respectivos al sistema. Se estudió la cinemática del movimiento del robot, tomando en cuenta las velocidades de cada rueda, el retroceso, avance y giro. Se plantearon las ecuaciones respectivas para cada variable de estado, tomando el robot como un péndulo invertido. Se escogió un tipo de controlador *Brushless DC Motor* (BLDC, por sus siglas en inglés) para regular la activación y control de las bobinas de los motores, evaluando los voltajes, corrientes y costos de cada placa. En la Figura 4 se muestra el modelado 3D del robot a construir, el cual permitió una mejor visualización del acople de los componentes a utilizar.

Figura 4: *Modelado del robot 3D*



Adaptado de Francisco Quiles Bru (2021) [3]

Se utilizaron varios sensores como el LIDAR, IMU e infrarrojos. Se diseñaron varios prototipos de modelado 3D analizando las propiedades mecánicas como el módulo elástico, límite de tracción, deformación unitaria, alargamiento de rotura y resistencia al impacto. Se diseñaron de tal manera que los circuitos electrónicos, pantalla y ruedas encajaran entre sí. El resultado del proyecto fue un prototipo robótico con un centro de gravedad muy bajo lo cual daba una muy buena estabilidad al sistema. Tanto los sensores como los motores tuvieron un funcionamiento adecuado. Los componentes estructurales del sistema fueron satisfactoriamente resistentes. A través de la plataforma de Lab View, se realizó la programación necesaria para que el robot lograra mantener el equilibrio a través de un PID. La forma compacta de la estructura presentó problemas para conectar los componentes electrónicos, y además se presentó sobrecalentamiento en los drivers debido a una alimentación incorrecta. [3]

Actualmente, en el curso de Sistemas de Control 1 y Sistemas de Control 2 del Departamento de Ingeniería Electrónica, Mecatrónica y Biomédica de la Universidad del Valle de Guatemala, no se cuenta con sistemas físicos de péndulos invertidos. Una plataforma de aprendizaje de péndulo invertido bien diseñada es de beneficio como recurso educativo para el curso de Sistemas de Control, de manera que los estudiantes puedan desarrollar sus competencias de control en un prototipo real.

Con este sistema mecánico se pueden implementar distintas técnicas de control, ya que la propuesta incluye los sensores y actuadores necesarios para conocer la dinámica del sistema y estabilizarlo según el método de control deseado. El péndulo invertido es una plataforma que ofrece al estudiante una experiencia de teoría y práctica, donde podrán internalizar conceptos abstractos, mejorando su comprensión sobre los principios básicos de control y dinámica. Este sistema mecánico se adapta a las técnicas de control impartidas en el curso de Sistemas de Control 1 y Sistemas de Control 2, donde incluso el estudiante podrá trabajar en la implementación de nuevos algoritmos de control innovadores.

4.0.1. Objetivo general

Implementar una plataforma interactiva y funcional de un péndulo invertido para utilizar en el laboratorio del Departamento de Ingeniería Electrónica, Mecatrónica y Biomédica de la Universidad del Valle de Guatemala.

4.0.2. Objetivos específicos

- Diseñar el sistema mecánico y electrónico de un péndulo invertido.
- Fabricar el sistema mecánico y electrónico de un péndulo invertido.
- Diseñar el sistema de control de un péndulo invertido.
- Calibrar el sistema de control con el péndulo invertido en físico.
- Desarrollar una interfaz de usuario intuitiva que permita al estudiante interactuar con el péndulo invertido.

Este proyecto se enfocó en el diseño y control de un péndulo invertido, el cual tiene como fin el desarrollo de una plataforma de aprendizaje a utilizar en el laboratorio del Departamento de Ingeniería Electrónica, Mecatrónica y Biomédica de la Universidad del Valle de Guatemala.

El desarrollo de este proyecto se conforma en distintas etapas. La primera consiste en el proceso de diseño, en donde se abarcan todos los componentes mecánicos necesarios, como la estructura del péndulo, mecanismos de actuación y soportes. También se toma en cuenta el diseño del circuito electrónico el cual controla al sistema. A partir del diseño desarrollado, se fabricó el sistema, incluyendo el ensamblaje de las piezas y construcción del circuito electrónico, asegurando un funcionamiento seguro y confiable. Luego se diseñaron los sistemas de control que permiten estabilizar el péndulo en una región cercana al punto de operación. Esto incluye el modelado del sistema linealizado y la implementación de los algoritmos de control. Se implementó el control PID, y el control por variables de estado, seleccionando las constantes del vector K con *Pole placement* y LQR.

Una vez fabricado el sistema, se procedió a la calibración del péndulo colocado hacia arriba y colocado hacia abajo. Esto incluye la calibración de constantes en el sistema de control, la configuración de los parámetros del motor y ajustes mecánicos, asegurando la estabilidad del péndulo. Por último se desarrolló una interfaz que permite al usuario interactuar fácilmente con la plataforma de aprendizaje. Esta plataforma incluye funciones para monitorear los parámetros de control y configuración del sistema.

6.1. Péndulo invertido

El péndulo invertido es un sistema que consiste en una masa conectada a una barra con libertad de oscilar alrededor de un pivote, el cual es montado sobre un carro capaz de deslizarse horizontalmente. El centro de masa del péndulo invertido está ubicado por encima del punto de suspensión, lo cual genera un comportamiento dinámico inestable. Por esta razón, el sistema requiere un control activo para mantenerlo en la posición vertical, estabilizándolo en el punto de operación en un tiempo óptimo.

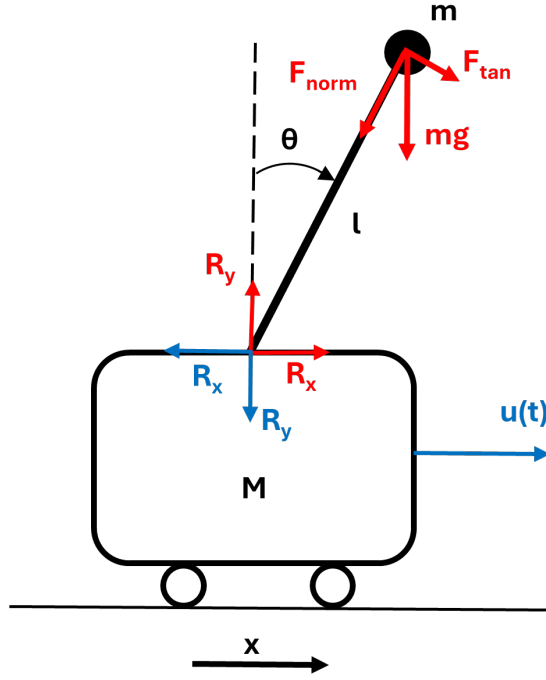
El péndulo invertido es un tema de estudio que permite analizar y comprender el comportamiento y control de un sistema dinámico no lineal, captando conceptos básicos sobre el control de sistemas en el campo de la ingeniería, física y la ciencia de control. El estudio del péndulo invertido puede abarcar varios conceptos de análisis estabilidad bajo distintas condiciones como variaciones en la longitud del péndulo, cambios en la masa o fuerzas externas.

Este sistema permite varias aplicaciones prácticas como el diseño de robots autónomos que deban mantener el equilibrio, control de brazos robóticos o sistemas de transporte auto balanceados. Es posible desarrollar distintas estrategias de control que permitan estabilizar el sistema en el punto de operación deseado.

6.2. Modelo del sistema péndulo invertido

ES importante plantear el modelo del sistema físico a partir de un diagrama de cuerpo libre como se presenta en la Figura 5, donde M es la masa del carro, l el largo de la barra, m la masa conectada al péndulo, θ el ángulo entre el eje vertical y la barra, x el desplazamiento del carro y u la fuerza aplicada al carro. Las fuerzas en color rojo actúan sobre el péndulo, mientras que las fuerzas en azul son las que actúan sobre el carro.

Figura 5: Diagrama de cuerpo libre



Adaptado de Osar Oswaldo (2022) [1]

Se puede apreciar que el péndulo experimenta en el eje vertical una fuerza de reacción R_y y otra fuerza mg debido al peso de la masa. En el eje horizontal se experimenta una fuerza de reacción R_x debido al pivote donde oscila. Además debido al movimiento oscilatorio del péndulo, se experimenta una fuerza normal y una tangencial al movimiento circular. En primer lugar, se analizarán las fuerzas que actúan sobre el carro deslizante en el eje horizontal. Se tiene una fuerza de reacción R_x , la fuerza de entrada al sistema $u(t)$ y se considera una fuerza de rozamiento con el aire que es proporcional a la velocidad del carro que viene dada por un factor β .

$$\sum F_x = M a_x \quad (1)$$

$$u(t) - F_{\text{rozamiento}} - R_x = M \ddot{x} \quad (2)$$

$$M \ddot{x} + \beta \dot{x} + R_x = u(t) \quad (3)$$

Por otra parte, se analizan las fuerzas que actúan sobre el péndulo en el eje horizontal. Es importante tomar en cuenta que la fuerza normal viene dada por la velocidad tangencial de la masa, que se puede expresar en función de la velocidad angular y largo del péndulo.

$$F_{\text{norm}} = m \frac{V^2}{l} = m \frac{(\dot{\theta}l)^2}{l} = ml\dot{\theta}^2 \quad (4)$$

La fuerza tangencial del péndulo se puede expresar en función de la aceleración angular como se muestra a continuación.

$$F_{tan} = ma_{tan} = m\ddot{\theta}l \quad (5)$$

Haciendo la sumatoria de fuerzas, se considera fuerza de reacción R_x , la componente horizontal de la fuerza normal y la componente horizontal de la fuerza tangencial. Considerando los signos respectivos de las fuerzas en el diagrama se obtiene lo siguiente.

$$\sum F_x = ma_x \quad (6)$$

$$R_x - ml\dot{\theta}^2 \sin \theta + ml\ddot{\theta} \cos \theta = m\ddot{x} \quad (7)$$

Despejando la fuerza de reacción en la Ecuación 7 y sustituyéndola en la Ecuación 3 se obtiene la primera ecuación que describe al sistema del péndulo invertido.

$$(M + m)\ddot{x} + \beta\dot{x} + ml\ddot{\theta} \cos \theta - ml\dot{\theta}^2 \sin \theta = u(t) \quad (8)$$

Por otro lado, se plantea la sumatoria de momentos en el péndulo - masa para analizar el movimiento rotacional. Se aprecia que las fuerzas que generan momento en el péndulo son la fuerza tangencial, el peso y la fuerza horizontal aplicada.

$$\sum M = I\ddot{\theta} \quad (9)$$

$$-ml\ddot{\theta}(l) - mg \sin \theta(l) - m\ddot{x} \cos \theta(l) = I\ddot{\theta} \quad (10)$$

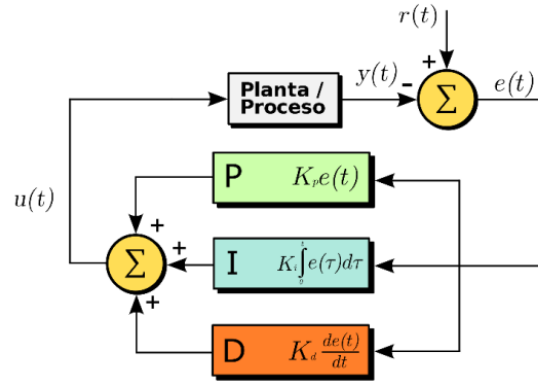
$$(I + ml^2)\ddot{\theta} + m(g \sin \theta + \ddot{x} \cos \theta) = 0 \quad (11)$$

6.3. Sistemas de control para péndulo invertido

Un sistema de control es un conjunto de elementos como sensores y actuadores, que trabajan para gestionar el comportamiento de un sistema dinámico. Tanto en la robótica, como en la automatización industrial o en la electrónica se aplican sistemas de control. El sistema de control procesa los datos de entrada y salida del sistema para modificar su comportamiento y estabilizar el sistema en un tiempo óptimo. El sistema de control puede ser analógico o digital dependiendo de los elementos involucrados para estabilizar el sistema.

Para aplicar control al péndulo invertido se es posible trabajar tanto con el control clásico como con el control moderno. En el control clásico se linealizan las ecuaciones en el punto de operación deseado, para obtener una planta del sistema y de esta manera aplicar un control proporcional, integral y derivativo (PID por sus siglas). El PID es una técnica de control que a través de un lazo de retroalimentación negativa, permite regular el comportamiento de la salida del sistema.

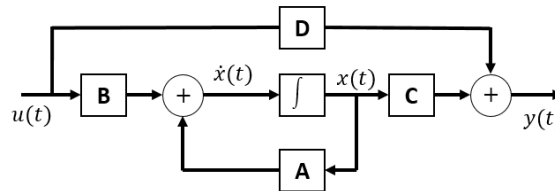
Figura 6: Diagrama de retroalimentación con PID



Adaptado de N.S. Nise (2012) [4]

Por otro lado, el control moderno no solo controla la salida del sistema, sino también toma en cuenta las variables de estado que describen completamente el comportamiento dinámico. Esta técnica permite el diseño de controladores robustos que pueden manejar parámetros del sistema y perturbaciones externas de forma más efectiva. A través de retroalimentación negativa de las variables de estado y un vector de ganancia basado en penalizaciones de las variables de, es posible controlar el sistema de forma efectiva.

Figura 7: Diagrama de retroalimentación con variables de estado

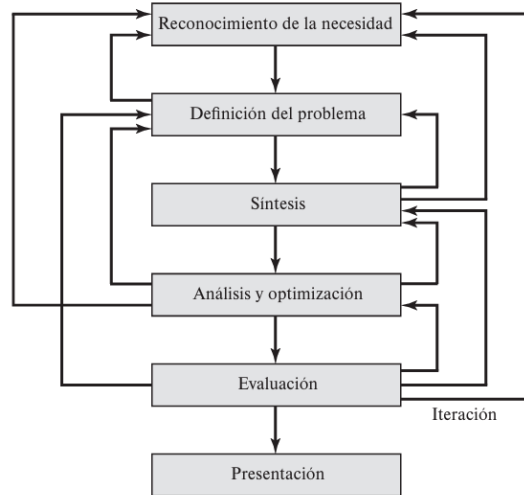


Adaptado de Osar Oswaldo (2012) [1]

6.4. Diseño mecánico

Según el libro Diseño de Ingeniería Mecánica de Shigley [5] el diseño mecánico es la planificación que busca satisfacer una necesidad o resolver un problema. El resultado del diseño debe materializarse, ser funcional, confiable y seguro. Es importante tomar en cuenta que el proceso de diseño debe ser innovador con varias iteraciones que ayuden a mejorar el producto final. El proceso de diseño se puede presentar en la Figura 8, donde algunas etapas se pueden repetir para ir actualizando el diseño de mejor manera.

Figura 8: Diagrama del proceso de diseño de Shigley



Adaptado de Shigley (2018) [5]

En primera lugar, el diseño empieza con el reconocimiento de la necesidad, donde el diseñador debe saber identificar de forma efectiva la necesidad puntual, ya que muchas veces no es del todo evidente. La necesidad puntual de este proyecto es que no se cuenta con un sistema físico de péndulo invertido en los laboratorios de Sistemas de Control 1 y 2, para poder aplicar la teoría de las técnicas de control.

Siguiendo con el diagrama de proceso de diseño de Shigley, se debe definir el problema, indicando todas las especificaciones del producto a diseñarse, como las dimensiones del espacio de trabajo en el laboratorio de Sistemas de Control, las entradas de alimentación, las salidas del actuador y las variables internas del sistema con sus respectivas limitaciones. Es importante tomar en cuenta las limitantes en la manufactura del péndulo invertido, y los recursos disponibles para fabricar el proyecto. Entre estos están la impresora 3D, cortadora láser, fresadora, torno, taladro, entre otras máquinas herramientas.

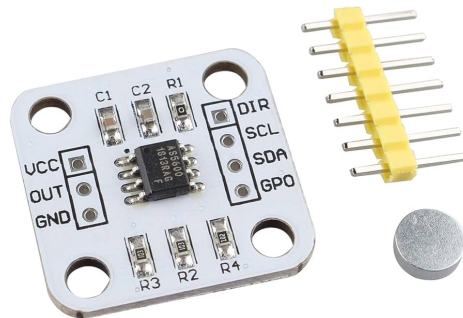
La etapa de síntesis consiste en presentar una invención o diseño del concepto, proponiendo varios esquemas de solución, con respaldo de investigación y unidades cuantificables concretas. Se realiza un análisis para evaluar si el diseño del péndulo invertido es satisfactorio y que tan bien se desempeñará para estabilizarse según las perturbaciones externas como la fricción o manipulación manual. Varios esquemas de solución se descartan o se mejoran, hasta el punto de tener distintos esquemas para seleccionar el que presente una solución más óptima.

La evaluación indica el análisis final de un diseño exitoso, de manera que si aún no se cumple con los requerimientos específicos del sistema del péndulo invertido, se repiten los pasos anteriores. Se evalúa si el producto final es confiable, funcional o si viable de manufacturar. Finalmente, se comunican los resultados finales a través de una presentación visual y descriptiva, que en este caso es a través del trabajo de graduación, indicando los beneficios del proyecto y los resultados obtenidos.

6.5. Sensor Encoder magnético AS5600

El módulo encoder magnético AS5600 [9](#) es un sensor de posición giratorio de alta resolución, el cual puede ser programable mediante el protocolo de comunicación I2C. Este sensor magnético sin contacto tiene una resolución de 12 bits, el cual puede funcionar como un encoder sin límites en el ángulo de rotación. Este tipo de sensor generalmente se utiliza para medir la velocidad y posición angular de la rotación del eje de un motor stepper. Se acopla un pequeño imán en forma de disco en el eje a medir el ángulo, y luego se posiciona el circuito a una distancia muy corta pero sin tener contacto con el imán. A medida que rota el eje, el sensor detecta la variación angular.

Figura 9: *Sensor encoder magnético AS5600*



Adaptado de INTEC (2022) [2](#)

Este encoder magnético se basa en cuatro sensores de efecto Hall colocados en círculo, los cuales detectan cambios en la dirección del campo magnético del imán. La información del campo magnético es amplificada por un amplificador integrado y luego se convierte a digital mediante un módulo ADC integrado de 12 bits. De esta forma se pueden medir 4096 posiciones distintas, es decir una resolución aproximadamente de 0.089 grados. A diferencia de encoders convencionales, este sensor tiene una resolución muy alta para detectar pequeños cambios en el ángulo. Este sensor se utilizará para medir la posición angular del péndulo al momento de oscilar en el carro deslizante.

Proceso de diseño mecánico y electrónico

En el presente capítulo se detallan los procedimientos realizados del diseño del péndulo invertido. El diseño mecánico se divide en dos partes principales, el diseño del desplazamiento lineal del carro y el diseño del péndulo. Mientras que el diseño electrónico abarca los sensores angulares y el motor paso a paso, así también como su alimentación. Se mostrará la evolución del diseño, explicando las modificaciones realizadas en cada iteración hasta llegar a la versión final.

7.1. Péndulo invertido - Versión 1

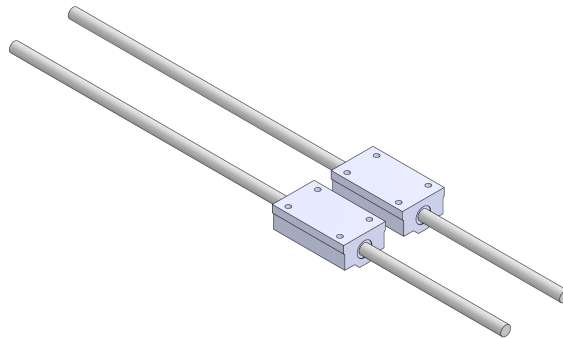
7.1.1. Diseño mecánico - Versión 1

En primer lugar, es importante notar que el diseño mecánico implica dos partes principales, el desplazamiento lineal del carro y el sistema de movimiento pendular. A continuación, se detalla el diseño del prototipo preliminar de cada uno.

Diseño desplazamiento lineal - Versión 1

Debido a que el desplazamiento lineal del sistema requiere tener la menor fricción posible, se tomaron en cuenta dos alternativas. La primera opción fue utilizar cojinetes lineales con barras de acero inoxidable de manera que dos cojinetes acoplados a un carro se desplazaran a través de las barras como se muestra en la Figura 10. Sin embargo las barras tendrían que estar perfectamente alineadas en paralelo para que no existiera fricción alguna, lo cual podría ser un problema.

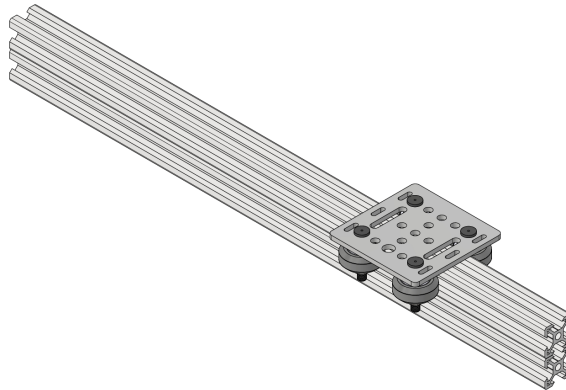
Figura 10: *Barras con cojinetes lineales*



Elaboración propia

Aunque esta opción era viable se recurrió a otra opción más segura, que fue un riel lineal de aluminio (*V-SLOT*) con dimensiones del perfil 20 x 40 mm. El riel está conformado por un carro deslizante con cuatro rodamientos V como se muestra en la Figura [11](#). De esta manera el sistema tendrá un desplazamiento lineal más confiable y sin fricción. Se escogió un largo de perfil de 1 m para que el carro tuviera suficiente desplazamiento para realizar el balanceo y control del péndulo. La ventaja de utilizar este perfil, es que se pueden utilizar tuercas T para atornillar elementos a lo largo del perfil. El riel está conformado por dos perfiles de 20 x 20 mm. En el perfil superior se movilizan los rodamientos V. Mientras que en el perfil inferior se atornillan los componentes para movilizar el carro deslizante a lo largo del riel.

Figura 11: *Perfil de aluminio (V-Slot)*

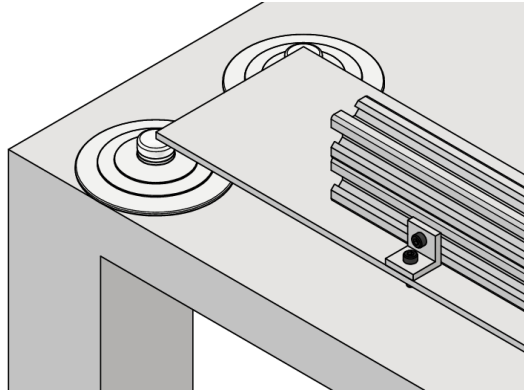


Elaboración propia

El perfil de aluminio se fija a una plataforma MDF de 60 x 1500 mm mediante cuatro piezas que se fabricarán con impresión 3D. Estas piezas se atornillan con tornillos allen M4 a la plancha de MDF y al perfil de aluminio con tuercas T, las cuales periten una mejor unión. De esta forma el usuario podrá transportar el sistema completo a cualquier mesa de trabajo. Fue importante colocar el perfil de aluminio cerca del costado de la plataforma MDF para que el péndulo pudiera oscilar correctamente. La persona que utilice este sistema igual debe

colocar la plataforma en la orilla de la mesa para que el péndulo funcione de forma correcta. Debajo de la plataforma MDF se colocaron 4 ventosas para que el movimiento brusco del carro no provoque un deslizamiento de todo el sistema en la mesa de trabajo. El perfil de aluminio se fija al MDF tal como se muestra en la Figura [12](#).

Figura 12: *Posicionamiento de plataforma*

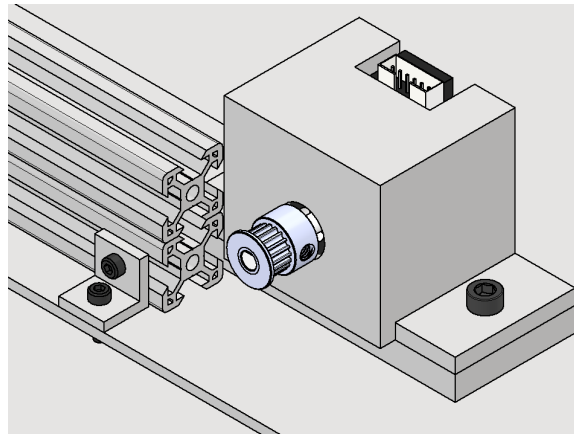


Elaboración propia

El carro del riel cuenta con tuercas excéntricas las cuales ayudan a ajustar los ejes de los rodamientos V, y que el sistema tenga un desplazamiento correcto. Se consiguieron los modelos CAD de la plataforma de desplazamiento y del perfil de aluminio para poder diseñar el ensamble del prototipo.

Para poder diseñar el sistema de actuación que moviliza la plataforma a lo largo del riel se decidió colocar una faja dentada de 6 mm. Esta faja se conectó en los extremos de la plataforma móvil de manera que en un extremo del riel se coloca el motor que moviliza la faja a través de una polea dentada como se muestra en la Figura [13](#). Para fijar el motor en la posición correcta, se diseñó una pieza en 3D que lo sujeta, alineando el eje del motor en el centro del perfil superior. La pieza que se muestra en la Figura [13](#) ayuda a darle la elevación correcta al motor para que el eje quede correctamente alineado para mover la faja dentada. La polea dentada tiene un diámetro de 15mm, con 16 dientes, la cual se atornilla al eje del motor stepper con un tornillo allen. A la pieza que sostiene al motor se le diseñó un espacio para poder conectar los cables de las bobinas del motor. Fue importante tomar las medidas exactas del motor físicamente para poder diseñar la pieza y que encaje correctamente.

Figura 13: *Acople de motor stepper*

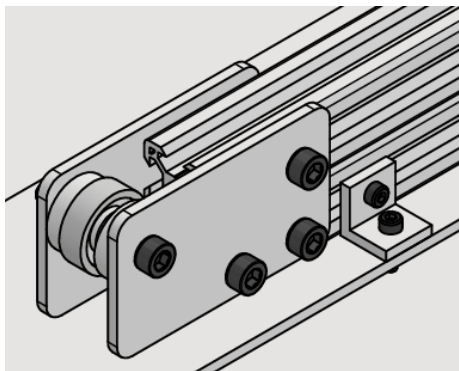


Elaboración propia

En el otro extremo se colocó una polea lisa, de manera que la faja pasara en medio del perfil de aluminio para movilizar la plataforma como se presenta en la Figura 14. Para fijar la polea lisa se utilizaron dos placas de MDF atornillándolas al perfil de aluminio, y luego colocando un tornillo M5 que funciona como eje de la polea. Dado que la polea lisa no es del mismo ancho que el perfil de aluminio, se colocaron dos piezas de impresión 3D como se muestra en la Figura 14 para que se mantenga centrado al perfil y no se mueva a lo largo del eje que lo sostiene. A este tornillo se le colocó una tuerca de seguridad para que no apriete a la polea y permita moverse libremente.

Fue importante escoger un diámetro de 12 mm el cual es el mismo diámetro de la polea dentada para que la faja se mantuviera alineada con el eje horizontal. También se diseñaron las piezas de MDF de manera que lograran alinear el centro de la polea lisa con el centro del perfil superior para que la faja fluyera libremente tanto en la parte superior como en la parte inferior del primer perfil.

Figura 14: *Acople polea lisa*



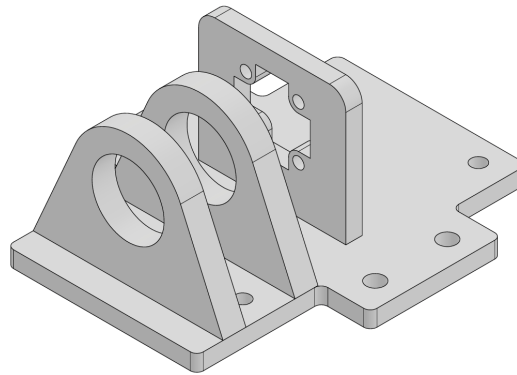
Elaboración propia

Diseño péndulo - Versión 1

Para el diseño del péndulo se diseñó una plataforma que se fabrica en impresión 3D. Esta plataforma tiene la funcionalidad de posicionar el sensor AS5600 para medir el ángulo, sujetar la faja dentada para movilizarse en el riel y sujetar el péndulo para que oscile libremente. En la Figura 15 se presenta la base diseñada del péndulo, la cual va acoplada con el carro deslizante mostrado anteriormente. Se detalla cada una de sus características y funcionalidades para el sistema.

Para posicionar el sensor angular AS5600 correctamente centrado con el eje central se diseñó la primera pared que se muestra en la Figura 15, la cual consta de 4 agujeros para atornillar el sensor y un espacio central que permite la lectura magnética del eje de rotación. Se utilizan tornillos M3 para sujetar el sensor AS5600, y se colocan tuercas incrustadas en el material para poder atornillar con mayor facilidad. Los agujeros del sensor AS5600 son de 4 mm pero se utilizaron tornillos M3 para que se pudiera ajustar su posición al momento de hacer la calibración de las lecturas con el eje de rotación. Un leve desfase entre el sensor magnético y el imán podría afectar significativamente las lecturas del sensor. Las mediciones angulares deben ser lo más precisas posibles para lograr un buen control.

Figura 15: Base para péndulo



Elaboración propia

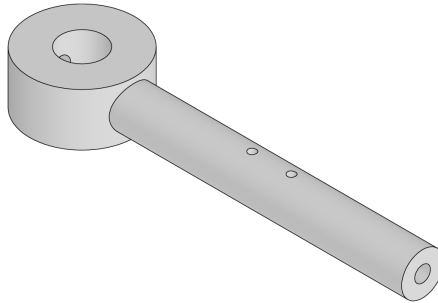
Es importante mencionar que el péndulo debe oscilar con la menor fricción posible, para que el control no se vea afectado. Por lo tanto, se decidió utilizar dos cojinetes para sostener el eje de rotación del péndulo, de manera que al tener dos puntos de sujeción, el torque generado por el peso y oscilación del péndulo no descentraliza el eje de rotación. Estos cojinetes se colocan a presión en las dos paredes que se muestran en la pieza de la Figura 15. El diámetro externo del cojinete es de 8mm, por lo que en el diseño de la pieza se dejaron agujeros de las mismas dimensiones.

Para conectar la base del péndulo al carro deslizante se colocan cuatro tornillos M5 en los orificios diseñados, los cuales son los mismos tornillos que sujetan a los cojinetes V. De esta manera se pueden reutilizar para hacer la unión de todos los componentes al mismo tiempo.

Otra pieza importante en el diseño del péndulo es el eje de rotación que se muestra en la Figura 16, el cual se fabrica con impresión 3D. Este eje tiene la funcionalidad de acoplar la barra oscilante del péndulo con la plataforma deslizante, transmitiendo la medición de la posición angular al sensor magnético AS5600. En el extremo del eje de rotación se colocó un imán de 4 mm de diámetro, el cual estará a 5 mm de distancia del sensor angular. Si el imán se encuentra a una distancia mayor, puede que el sensor no haga la medición de forma correcta. En el otro extremo del eje se diseñó un orificio de 8 mm, donde se acopla la barra oscilante del péndulo. Para atornillarla se diseñó axialmente un agujero de 3 mm, donde se coloca un tornillo de 6 mm de longitud.

Como se muestra en la Figura 16, el eje central tiene dos agujeros en el centro, separados 7 mm. Estos agujeros se diseñaron con la finalidad de colocar dos pines que limitarán el movimiento axial del eje cuando el péndulo oscile. Están separadas a esta distancia, dado que el ancho del cojinete es de 7 mm.

Figura 16: *Eje central*



Elaboración propia

La barra oscilante se diseñó con un diámetro de 8 mm y con una longitud de 600 mm, ya que el control del péndulo invertido es más eficiente con barras de mayor longitud. Para que el tornillo M3 se acoplara mejor a la barra oscilante, se diseñó un agujero de 3.2 mm con una pequeña profundidad en el extremo inferior. El material de fabricación propuesto fue de acero inoxidable para una mejor resistencia.

Para la masa en el extremo del péndulo se diseñó un cilindro como se muestra en la Figura 17, el cual se atornilla al péndulo con un tornillo M4. Esta pieza fue diseñada para maquinarla con acero inoxidable. La masa tiene la ventaja de poder desplazarla a lo largo del péndulo, lo cual permite hacer pruebas del rendimiento de control aplicado, e ir modificando las constantes del control para analizar su comportamiento.

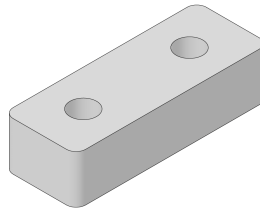
Figura 17: *Masa del péndulo*



Elaboración propia

Para conectar la faja dentada al carro deslizante se diseñaron dos piezas que se muestran en la Figura 18, de forma que presen la faja con dos tornillos. Con estas piezas se podrá tensar la faja dentada de mejor manera. Es importante que la faja esté lo suficientemente tensa para que el contacto con la polea dentada sea adecuado y logre responder ante cambios bruscos de velocidad.

Figura 18: *Unión a faja*



Elaboración propia

7.1.2. Diseño electrónico - Versión 1

En el diseño electrónico se toman en cuenta los dos sensores angulares que medirán los ángulos y velocidades, y el motor stepper. Se toma en cuenta la alimentación de los dispositivos y la comunicación con el microcontrolador. Primero se comienza con un prototipo preliminar para realizar todas las conexiones en un protoboard y verificar el funcionamiento.

Motor Stepper - Versión 1

Para el Motor Stepper primero fue importante considerar la alimentación. Según la hoja técnica de datos, la corriente nominal del motor es de 2A. Sin embargo se utilizará un driver para el motor, el cual es el TMC2130. Este controlador tiene la capacidad de regular la corriente que pasa por el motor, ajustándolo para la aplicación según se requiera en el torque. Aunque el motor está diseñado para 2 amperios por fase, el controlador puede operar a corrientes más bajas como 800 mili amperios. Para alimentar el motor a través de este driver se pueden implementar fuentes de alimentación de 5 a 46V. Por esta razón, se

seleccionó una fuente de 12V que provee 2A como máximo, el cual es suficiente para el motor que se está utilizando. Para eliminar las fluctuaciones de corriente, los ruidos y picos de voltaje se colocó un capacitor de 100 μ F en paralelo.

Luego, fue necesario alimentar el driver desde el microcontrolador para que pueda operar de forma correcta. Este driver opera desde 3.3V a 5V, lo cual debe ser independiente a la alimentación del motor. El STM proporciona 3.3V, por lo que se alimentó con este voltaje al TMC2130.

La comunicación entre el driver TMC2130 y el microcontrolador es a través de SPI. Por lo tanto se toman en cuenta las señales *Serial data out* y (SDO) *Serial data in* (SDI) para el envío de datos y comunicación entre el esclavo y el maestro. En la comunicación se reciben instrucciones desde el microcontrolador, como el ajuste de corriente o el control del micro step, mientras que el driver también puede proporcionar información al microcontrolador como el torque que se mide en el registro *Stall Guard*. Luego se toma en cuenta señal *Serial clock* (SCK) que proporciona el pulso de reloj para sincronizar la comunicación entre ambos dispositivos. También se utilizan los pines *Chip Select* (CS) y *Enable* (EN) para seleccionar y activar el driver, que en este caso solo se tiene un driver. [6](#)

Luego se tiene el pin de STEP que recibe los pulsos desde el microcontrolador para mover al motor los micropasos según se realizó en la configuración. Por último se toma en cuenta el pin de DIR, el cual indica al driver si el motor debe girar en sentido horario o antihorario. Con 1 lógico, el driver interpreta como sentido horario, mientras que el 0 lógico lo interpreta como sentido antihorario. En la Figura [52](#) de los anexos se muestra el esquemático de las conexiones del driver. Luego de hacer todas estas conexiones se verificó la funcionalidad del motor.

Sensores - Versión 1

Para la conexión del sensor angular AS5600, se toma en cuenta la alimentación, la cual es 3.3V proporcionado por el microcontrolador. En el sensor del motor se realizó la conexión con *jumpers*, mientras que en el sensor del péndulo, fue necesario optar por cables más largos, ya que se mantiene en movimiento a lo largo de todo el riel. Se tiene la precaución que los cables estén debidamente conectados para no provocar desconexiones durante el movimiento. [7](#)

En el aspecto de la comunicación de los sensores, se trabajó con el protocolo de comunicación I2C, el cual cuenta con dos hilos. El primero es el *Serial Clock Line* (SCL), el cual proporciona el reloj de sincronización entre el esclavo y el maestro. Luego se conecta el *Serial Data Line* (SDA), el cual consiste una línea bidireccional que sirve para enviar y recibir datos al esclavo. Sin embargo, solo un dispositivo puede transmitir datos en un momento dado según los comandos que se envíen.

Tanto el pin SDA como el SCL necesitan estar conectados a una resistencia pull-up, generalmente de 4.7k Ω a 10k Ω asegurando que el pin esté en un estado lógico alto cuando no se accione. El pin realiza la conmutación de la línea enviando un estado lógico 0, o un estado lógico 1 a través de la resistencia. Esto asegura que los cambios entre niveles lógicos sean estables cuando ningún dispositivo este transmitiendo datos. En las Figuras [53](#) y [54](#)

de los anexos se muestran los esquemáticos de las conexiones. Luego de hacer todas las conexiones, se verificó que las lecturas de los sensores fueran correctas.

Por otra parte fue necesario conectar las señales digitales de los finales de carrera, las cuales indican si el carro deslizante llegó al límite del riel. En cada final de carrera se conectan tres hilos, dos de alimentación, donde se trabaja con 3.3V, y un hilo que es la señal de lectura digital tal como se muestra en la Figura 55 de los anexos. Se procuró utilizar cables largos para el final de carrera en el extremo izquierdo, asegurando que no interfiera con el movimiento lineal del carro deslizante.

7.2. Péndulo invertido - Versión 2

7.2.1. Diseño mecánico - Versión 2

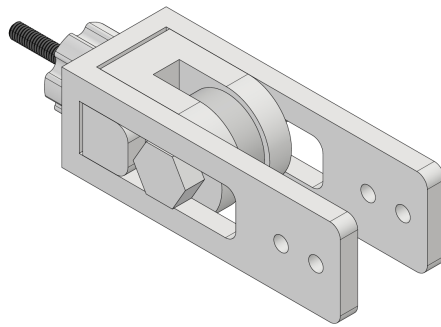
El primer prototipo presento muchas áreas de mejora, tanto mecánicas como electrónicas. Se sustituyeron varios elementos, y se presentaron nuevos diseños que ayudaran al usuario a configurar el péndulo de manera más eficiente.

Diseño desplazamiento lineal - Versión 2

En el primer prototipo se identificó una mejora para poder unir la faja al carro deslizante. En los extremos del carro deslizante ya existen orificios donde entra la faja dentada de 6 mm. Se utilizaron cinchos de plástico para prensar la faja dentada al carro deslizante. Por lo tanto ya no fue necesario colocar las piezas de la Figura 18 para sujetar las fajas al carro deslizante. Sin embargo fue necesario diseñar otro sistema para tensar la faja dentada.

Es muy importante que la faja este tensa para que la respuesta brusca del motor no haga que la faja se deslice en la polea dentada, y luego se pierda velocidad o posición en el sistema de control. Por tal razón se diseñaron las piezas de la Figura 19, la cual tendrá un tornillo M4 que al girarlo desplaza la polea lisa para tensar la faja. Este sistema es más práctico para ajustar o aflojar la faja si se desea desarmar el sistema o realizar algún cambio de polea. El ajuste de tensión es mucho más preciso que el prototipo anterior, en donde se tensaba jalando la faja manualmente y atornillando la pieza de la Figura 18.

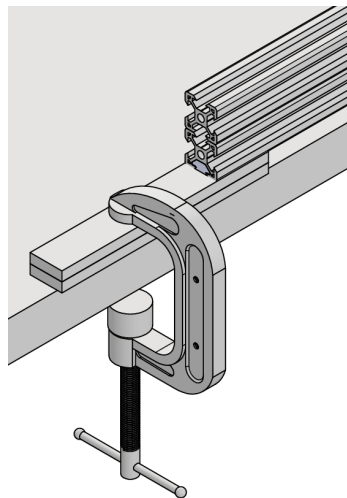
Figura 19: *Tensionador de faja*



Elaboración propia

Por otra parte, se tiene el riesgo que el movimiento brusco del péndulo cuando hace el swing up o desplaza el carro, podría deslizar la plataforma entera de MDF. Las ventosas que se colocaron le dan poca estabilidad a la plataforma, ya que tiende a inclinarse hacia los lados. Entonces mejor se diseñaron dos piezas de MDF de 6 mm como se muestra en la Figura 20, las cuales se atornillan en los extremos del riel, y con un sargento prensar estas piezas a la mesa de trabajo. Se atornilla el primer MDF al riel y luego se pega con cola blanca otro MDF para que la superficie con la mesa sea con el MDF y no con el tornillo.

Figura 20: *Fijación de plataforma*



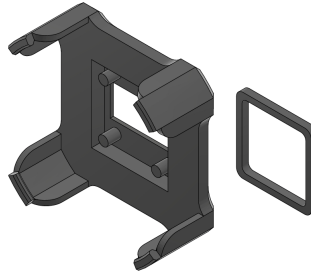
Elaboración propia

De esta forma ya no hay riesgo de que el sistema se deslice por cambios bruscos del péndulo. Para el usuario también es práctico mover el sistema completo a cualquier mesa de trabajo y fijarlo.

Para posicionar el sensor AS5600 que medirá la posición del motor stepper se buscó un modelo que convencionalmente se utiliza para acoplar este encoder al motor. Este modelo consiste en dos piezas que sujetan al sensor magnético y lo centra al eje del motor como

se muestra en la Figura 21. Se coloca una arandela de soporte para que el sensor quede acoplado a la pieza de sujeción. En la parte trasera tiene un espacio rectangular para que se puedan conectar los pines al microcontrolador.

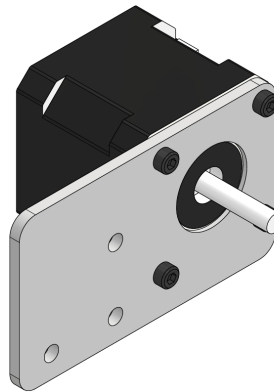
Figura 21: *Sujeción de sensor AS5600*



Elaboración propia

Por otro lado, para posicionar de mejor manera el motor, sin tener que usar material PLA y estar ajustando la altura del motor con la pieza de la 13, mejor se usó una pieza de MDF que se atornille al perfil de aluminio y acople al motor como se muestra en la Figura 22. De esta forma es más fácil centrar el eje del motor al perfil superior y que la faja no roce tanto en la parte inferior como en la parte superior del perfil. Se escogió un grosor de 3 mm para esta pieza.

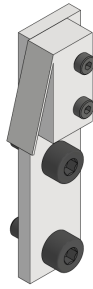
Figura 22: *Acople de motor*



Elaboración propia

Por otra parte, para evitar que el carro deslizante no topara en los extremos con el motor y el tensionador, se decidió implementar dos finales de carrera de manera que activen una señal para detener el motor. De esta forma se evita forzar el motor y que la faja dentada no se dañe con la polea del motor. Se diseñó una pieza en MDF la cual sostiene al final de carrera como se muestra en la Figura 23. Fue importante diseñar esta pieza de manera que el final de carrera esté a una altura adecuada para detectar al carro deslizante.

Figura 23: *Final de carrera*



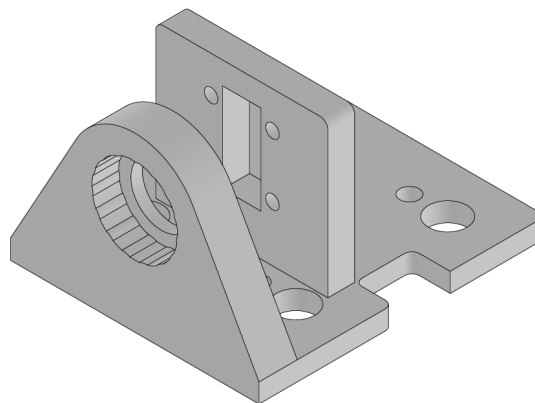
Elaboración propia

Diseño péndulo - Versión 2

Se realizaron varias mejoras a la base que sostiene al péndulo llegando a la pieza que se muestra en la Figura 24. En el primer prototipo, para unir la plataforma del péndulo al carro deslizante se utilizaron los mismos tornillos que sujetan a los cojinetes V. Sin embargo, se consideró que no es práctico aflojar todos los cojinetes si se desea cambiar la plataforma del péndulo o realizar algún ajuste. Por eso se diseñaron otros agujeros en la plataforma para que se atornille al carro deslizante independientemente de la unión con los rodamientos.

Se analizó el diseño, y se llegó a la conclusión que no hace falta colocar dos cojinetes, solo basta con uno, pero que esté bien colocado y apretado. Para que el cojinete entre a la plataforma de mejor manera, se diseñaron pequeños arcos a lo largo de la circunferencia donde entra el cojinete. Esta técnica ayuda que el cojinete esté más apretado y que no se logre salir fácilmente del orificio. También, en el primero prototipo al momento de insertar el cojinete, podría quedar torcido. Para esto se diseñó un tope como se muestra en la Figura 24, el cual rectifica al cojinete, asegurando que el eje de rotación este correctamente alineado.

Figura 24: *Base del péndulo*

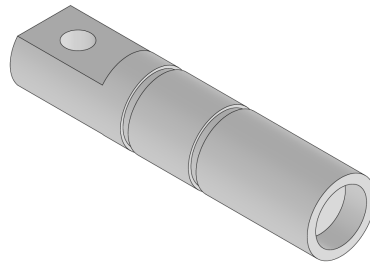


Elaboración propia

Además, se diseñó otro eje de rotación más resistente para el péndulo. El diseño anterior estaba pensado para fabricarlo en impresión 3D, sin embargo la pieza debe ser perfectamente cilíndrica, y con la impresión 3D no se puede lograr tal exactitud. Además, las capas de fabricación del PLA pueden ser vulnerables ante los esfuerzos de flexión y corte que experimenta el eje. Por lo tanto, mejor se recurrió al maquinado de una barra de acero inoxidable.

Se realizaron modificaciones en el eje de rotación para que no se desplazara axialmente con el cojinete. Se diseñaron dos ranuras para colocar anillos de retención de 7 mm de diámetro interno a los lados del cojinete. Las ranuras están separadas la misma distancia que el grosor del cojinete, que es 8 mm. También se colocaron dos secciones rectas en el extremo del eje para que al momento de atornillar el péndulo, la barra y el tornillo queden correctamente perpendiculares como se muestra en la [Figura 25](#). De esta manera, el péndulo queda con un mejor acople con el eje de rotación.

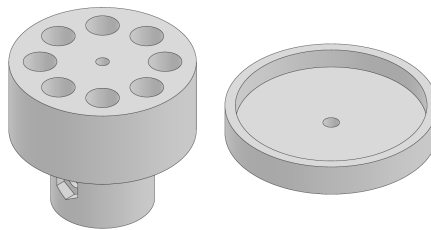
Figura 25: *Eje rotativo modificado*



Elaboración propia

Para la masa del péndulo, mejor se decidió implementar una pieza que aloje pequeñas masas de acero inoxidable para modificar la masa total y hacer distintas pruebas con el sistema de control. Como se muestra en la [Figura 26](#), la pieza puede alojar 8 cilindros de 8 mm diámetro por 25 mm de largo. Además se tiene la posibilidad de desacoplarse del péndulo en dado caso se quiera utilizar otra pieza. Para que los cilindros no se salieran de la pieza se diseñó una tapa como se muestra en la [Figura 26](#). A esta tapa se le diseñó un agujero para atornillarla contra la pieza principal.

Figura 26: *Masa modificada*



Elaboración propia

7.2.2. Diseño electrónico - Versión 2

Para asegurar una mejor conexión de los componentes y eliminar errores comunes que pueden surgir en las conexiones con protoboards, se optó por diseñar un circuito PCB en la plataforma de Easy Eda [8]. Esto garantiza que las conexiones estén definidas de forma clara y precisa, evitando que el usuario interprete de manera errónea el cableado para la implementación. La disposición de los componentes en el PCB se basa en la distribución física de los dispositivos en la plataforma del péndulo. Otra ventaja de utilizar una placa PCB es que se pueden implementar otros controladores que tengan comunicación I2C y SPI, donde el usuario pueda identificar las conexiones necesarias.

Motor Stepper - Versión 2

Para la conexión del motor stepper se incluyeron headers para colocar el driver TMC2130 en la placa, de manera que si se desea implementar en otra aplicación, se pueda retirar del PCB sin ningún problema. Se colocaron pines para conectar desde el microcontrolador las señales necesarias de SPI mencionadas anteriormente, y también se colocaron otros pines que alimentan a las bobinas del motor stepper. Con un *terminal block* atornillable se conectan las líneas de del transformador de 12V que alimentan al motor stepper como se muestra en la Figura 28.

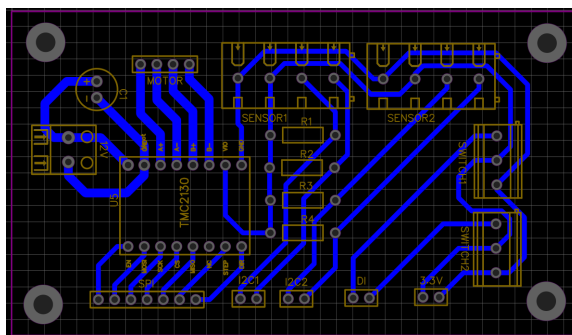
Se incluye el capacitor de 100 μF , el cual regula el ruido del voltaje y los picos de corriente que puedan llegar desde la fuente de alimentación. También se agregaron dos terminales para conectar el voltaje 3.3V de alimentación interna del circuito. Este voltaje no solo sirve para alimentar el TMC2130, sino también para alimentar los sensores magnéticos AS5600 y los finales de carrera.

Sensores - Versión 2

Para conectar los sensores a la placa se colocaron headers de manera que se pudieran identificar en el PCB por grupos de pines como se muestra en la Figura 28. En los sensores magnéticos se utilizan cuatro hilos, dos para alimentación y otros dos para la comunicación. Por eso, se colocaron dos grupos de cuatro pines en el borde de la placa.

Mientras que para los finales de carrera, se utilizan tres hilos, dos de alimentación y uno para la señal digital, colocando dos grupos de tres pines en el borde de la placa. Agrupar los pines ayuda a que el usuario pueda identificar las conexiones con mayor facilidad. Luego se implementaron los pines que van conectados a la comunicación I2C del microcontrolador, con sus respectivas resistencias en pull-up como se muestran en la Figura 27. Y por último se implementaron los pines que transmiten al controlador las señales digitales de los finales de carrera.

Figura 27: *Diseño de placa PCB*



Elaboración propia

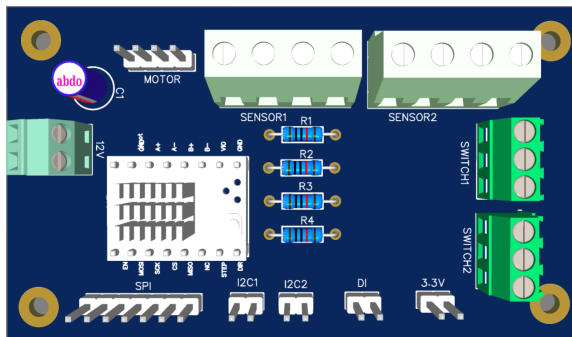
Debido a que son pocas pistas, todo se trabajó en una sola capa, en el *bottom layer*. Considerando que el microcontrolador maneja una corriente máxima en sus pines de control de 50 mili amperios con 3.3V y que la ruta más larga es de 100 milímetros, se determinó que el grosor adecuado para las rutas es de 2 mil. Esto se determinó con la plataforma de *ANSI PCB Trace Width Calculator* [9], en donde se especifican las condiciones de la pista para determinar un grosor adecuado. Es importante considerar que la placa tiene un grosor de 1oz de cobre, y que se permite un aumento de temperatura de 1 °C como máximo, con una temperatura ambiente de 20°C

En la Figura [57] de los anexos se muestran los datos proporcionados en la plataforma para determinar el ancho de pista. Sin embargo, el ancho de pista mínimo es de 20 mil en el servicio de fresado de la Universidad del Valle, por lo que se decidió escoger un grosor mayor, el cual fue de 30 mil. Según la plataforma [9], es importante respetar un espaciado mínimo entre las pistas de 23.66 mil para la prevención de cortocircuitos y manejo de altas corrientes.

Por otra parte es importante considerar que el motor consume más corriente, por lo que el grosor de las rutas que alimentan al motor deben ser de mayor grosor. De la misma manera, como se muestra en la Figura [58] de los anexos se determinó un ancho adecuado de pista de 54 mil para la alimentación de 12V considerando que la fuente y el motor manejan una corriente máxima de 2 amperios. En este caso la ruta más larga no pasa de 25 mm, y se considera un aumento de temperatura máxima de 20°C. Se obtiene un espaciado mínimo de 25.4 mil, lo cual nos da mayor seguridad de no generar un cortocircuito según la norma IPC-2221A que maneja la plataforma *ANSI PCB Trace Width Calculator*.

Para los pads se escogieron agujeros de 36 mil de diámetro los cuales son adecuados para insertar los pines en la placa PCB. Para tener un buen espacio de soldadura, se seleccionaron grosores de 65 mil de diámetro para los pads. [10] Se colocaron agujeros de 3 mm en las esquinas para atornillar la placa a una base. Finalmente se obtuvo el diseño de una placa PCB de 3500 x 2000 mil con un grosor de 1 oz de cobre. Con todas estas dimensiones también se cumplen las especificaciones mínimas de fabricación en el Maker Lab de la Universidad del Valle. Se colocaron los modelos 3D de los componentes a soldar en la placa PCB y se obtuvo el prototipo que se muestra en la Figura [28].

Figura 28: Placa PCB modelado 3D



Elaboración propia

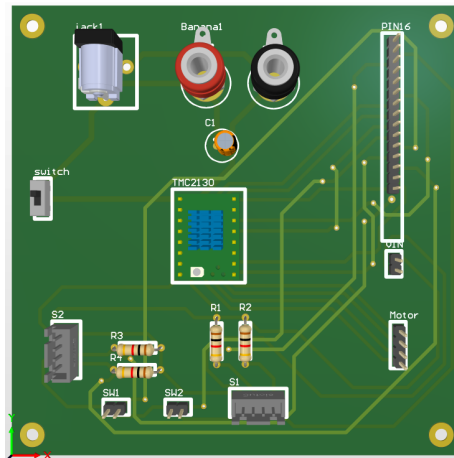
Es importante mencionar que para los cables que van hacia el sistema del péndulo invertido, se colocaron bloques terminales con tornillo para que el cable este sujetado correctamente y no hubieran problemas ante movimientos bruscos. Se identifican las terminales con etiquetas para que el estudiante pueda reconocer de forma efectiva las conexiones. Además se diseñaron agujeros de 3 milímetros para atornillar la placa PCB a una base con tornillos M3. Esta base se acopla al perfil de aluminio para que de esta manera se pueda fijar la placa al sistema completo.

7.3. Péndulo invertido - Versión 3

7.3.1. Diseño electrónico - Versión 3

Luego de haber diseñado y fabricado la primera placa PCB, se identificó la necesidad de que el sistema de conexión debía ser más práctico para el estudiante. Por esta razón se diseñó una segunda placa donde las conexiones se harían a través de pines con orientación para que el usuario no se confunda con el orden de conexión. Se diseñaron conectores con latch en 3D, los cuales tienen la funcionalidad de darle orientación a la conexión y hacer un click para que no se desprenda fácilmente el macho con la hembra. Para el sistema de conexión con el microcontrolador, se buscó poner todos los pines de comunicación en una sola fila de manera que el estudiante solo conecte el microcontrolador sin necesidad de estar verificando los pines y causar posibles confusiones como se muestra en la Figura 29. Esto ayudará a que no haya problema de comunicación y que no se confundan las conexiones entre los sensores, y que no se pierda la finalidad del laboratorio en hacer conexiones que lleven tiempo para el estudiante.

Figura 29: *Placa PCB v2 modelado 3D*



Elaboración propia

Con respecto a la alimentación de la placa, se colocó un *Barrel Jack* para que solo se conecte y desconecte de forma más práctica sin necesidad de usar un destornillador como se hacía con los bloques terminales. Como alternativa de alimentación, se implementaron conectores hembra banana que van hacia la fuente del laboratorio. En caso de requerir más potencia en el motor, el estudiante puede conectar un mayor voltaje, ya que el TMC2130 soporta hasta 46 voltios. Luego, para la seguridad del motor y en caso de emergencia, se colocó un switch que desconecta la alimentación de potencia con el resto del sistema. Esto es por si el estudiante llega a tener problemas con el motor o el driver al momento de querer realizar nuevas pruebas de control con el péndulo.

En este capítulo se detalla el procedimiento realizado para la fabricación y ensamble de los componentes que conforman a la plataforma del péndulo invertido. Se realizaron varios ajustes y modificaciones a lo largo del proceso de fabricación para que el sistema de control funcionara de la mejor manera. A continuación, se presenta el procedimiento realizado tanto para la fabricación mecánica, como para la fabricación electrónica.

8.1. Fabricación mecánica

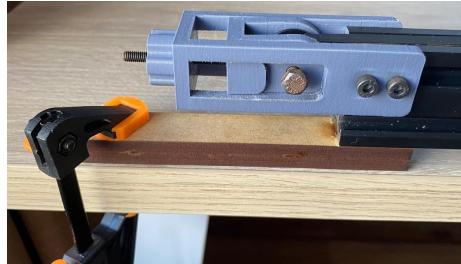
Para la fabricación mecánica, primero se trabajó con el desplazamiento lineal, tomando en cuenta el motor, el tensionador y el carro deslizante. Luego se trabajó con el sistema del péndulo, que consiste en el acople del eje de rotación, el péndulo y la masa. A continuación, se describe el procedimiento realizado para el ensamble y fabricación del desplazamiento lineal.

8.1.1. Fabricación desplazamiento lineal

En primer lugar, se consiguió el perfil de aluminio 20 x 40 mm de un metro de largo, junto con el carro de desplazamiento. Se verificó que el perfil tuviera la ranura correcta para que los cojinetes pudieran desplazarse y no se dañaran ya que existen otros perfiles que no tienen la ranura en V. Con las tuercas excéntricas se ajustaron los ejes de los cojinetes para que el desplazamiento sea suave, pero que tengan un buen ajuste y no quede flojo. Un mal ajuste podría afectar al desplazamiento del carro, ya que a lo largo del recorrido puede fluctuar. Para determinar que los cojinetes no hayan quedado muy apretados, se colocó el perfil a 45 grados y se verificó que el carro se desplazara sin fricción.

Luego, se fabricaron las piezas del tensionador con impresión 3D, las cuales consisten en la tuerca de ajuste, la base que se desplaza a la polea lisa, y la base fija que se atornilla al perfil de aluminio. Se utilizó un tornillo con cabeza hexagonal de 40 mm, de manera que tuviera el suficiente largo para desplazar la polea lisa a lo largo de la base. Ya instalado en la plataforma quedó como se muestra en la Figura 30. Al girar el tornillo, la polea se desplaza para adelante o para atrás con gran precisión, lo cual ayuda a tensar o aflojar la faja dentada de forma práctica.

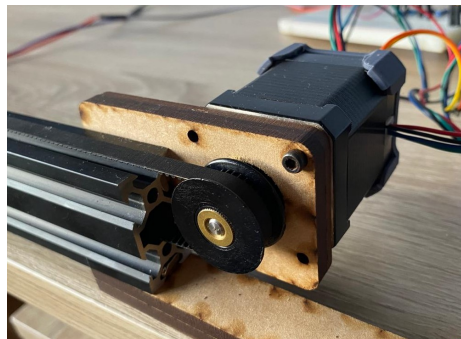
Figura 30: *Tensionador*



Elaboración propia

Por otra parte, se fabricó la base que acopla al motor con el perfil de aluminio. Esta pieza se fabricó en la cortadora láser con un MDF de 3 mm de grosor. Al colocar la pieza en el perfil de aluminio, se tuvo el problema que al momento de tensar la faja dentada, se deformaba la pieza. Entonces para evitar este problema, se agregó otra pieza idéntica pero de mayor grosor, con 6 mm de ancho. De esta forma se obtiene un ancho total de 9 mm, el cual es más resistente. En la Figura 31 se muestra el resultado del acople del motor en el perfil de aluminio.

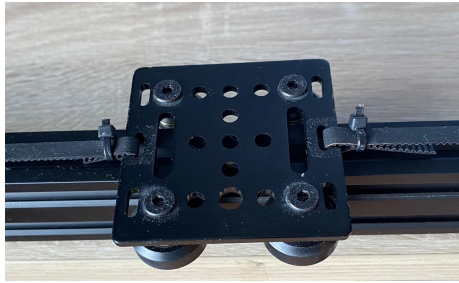
Figura 31: *Acople del motor en físico*



Elaboración propia

En los extremos del carro deslizante se colocó la faja dentada de 6 mm con cinchos plásticos como se muestra en la Figura 32. La faja dentada pasa sobre la superficie superior y rodea la polea lisa y luego se introduce en medio de los perfiles para poder movilizar el carro desde el extremo derecho con el motor stepper. De esta manera se puede movilizar el carro desde el extremo con el motor stepper.

Figura 32: *Carro deslizante*



Elaboración propia

Al momento de hacer pruebas con la movilidad del péndulo, se tuvo el problema que la polea lisa hacía que la faja topara con las paredes del tensionador, y el movimiento del péndulo se interrumpía por instantes. Por lo tanto, en su lugar se utilizó una polea lisa con paredes como se muestra en la Figura 30, la cual permite que la faja no esté en contacto con la pared del tensionador, sino en contacto con las paredes de la polea lisa. De esta manera ya no existe un roze que interrumpa el movimiento del carro deslizante.

8.1.2. Fabricación del péndulo - Versión 1

En primer lugar se fabricó la base del péndulo que va colocada sobre el carro deslizante con impresión 3D, el cual se muestra en la Figura 33. Fue importante verificar que la pared que sostiene al péndulo fuera resistente, ya que está sometida a un esfuerzo de flexión. El grosor de esta pared es de 10 mm con un relleno de 20 % lo cual es suficiente para soportar el esfuerzo de flexión. En esta plataforma se insertó el cojinete en el agujero hasta que topara en la pared y quedara alineado de forma correcta. Se verificó que el cojinete se ajustara suficientemente a presión en el agujero para evitar cualquier riesgo de aflojamiento ante movimientos bruscos.

Figura 33: *Base del péndulo físico*



Elaboración propia

Para maquinar el eje de rotación, se trabajó con una barra de acero inoxidable. En primer lugar, se cortó con una sierra a una medida de aproximadamente 37 mm, luego con una fresadora se fresaron los extremos para que quedaran totalmente rectos. Con la misma fresadora se fabricaron dos cuñas, cada una de 2 mm de profundidad en el extremo derecho como se muestra en la Figura ?? . Estas cuñas tienen la función de dejar recta la varilla del péndulo al momento de atornillarla al eje de rotación. Con el taladro se perforó un agujero de 4.2 mm, donde pasa el tornillo de 4 mm, el cual une el eje de rotación con el péndulo tal como se muestra en la Figura 33.

Figura 34: *Eje de rotación físico*

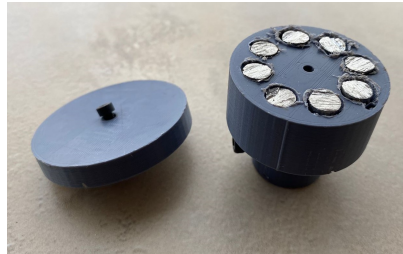


Elaboración propia

En el extremo izquierdo, se perforó un agujero de 4.9 mm para que el imán de 5 mm logre acoplarse a presión. Es importante mencionar que el maquinado de esta pieza debe ser muy precisa, ya que es responsable de la rotación del péndulo y de transmitir la posición angular al sensor. El imán debe estar entre un rango de distancia de 0.5 mm a 3 mm del sensor para que pueda hacer una lectura correcta. Esta pieza está diseñada para que el imán esté a una distancia de 2 mm del sensor. Luego, con el torno se fabricaron dos ranuras de 7 mm separadas a 8 mm para colocar los anillos de retención, los cuales impiden que el eje salga del eje de rotación cuando el péndulo este en funcionamiento.

Luego, con impresión 3D, se fabricaron las dos piezas que conforman la masa en el extremo del péndulo, las cuales son la tapa, y el cuerpo que aloja los cilindros de acero inoxidable. Se tuvo el problema que los agujeros para los cilindros del acero inoxidable eran muy pequeños. Por lo que se tuvo que utilizar una lija para agrandar los agujeros y que entraran libremente a la pieza como se muestra en la Figura 35. Para conseguir los cilindros de acero inoxidable, se tomó una varilla de repuesto de una impresora 3D y se cortaron 8 unidades con una longitud de 25 mm. Por último se colocó la tapa en la parte superior de la pieza para que los cilindros no se salieran cuando el péndulo esté en funcionamiento.

Figura 35: *Masa*



Elaboración propia

Al fabricar la barra de acero inoxidable del péndulo, se perforó un agujero de 3.8 mm en el extremo del péndulo para hacer una rosca para un tornillo de M4. Sin embargo al generar la rosca, el material era muy resistente que se rompió el machuelo. Luego de varios intentos de generar la rosca en el material, se decidió utilizar aluminio, que es más blando y maleable para generar la rosca sin ningún problema. Sin embargo, por motivos al momento de hacer el control, se determinó que una varilla de menor peso tendrá mejor rendimiento en la estabilización del sistema, por lo que se optó por una varilla de madera. En el otro extremo de la varilla se perforó otro agujero transversalmente para unir la masa al péndulo.

Al acoplar el sensor magnético que mide el ángulo del péndulo, no se tuvo ningún problema. Se colocaron los cuatro tornillos allen M3 en las esquinas del sensor y se atornilló a la pared de la base del péndulo. Se ajustó su posición de manera que estuviera centrado con el imán del eje de rotación. Se verificó que la distancia al imán fuera de 2 mm para hacer una medición correcta de la posición angular.

Para acoplar los finales de carrera al perfil de aluminio, se fabricó la pieza MDF que se muestra en la Figura 36. Se atornillaron los finales de carrera con tornillos M2, de manera que quedaran a una altura adecuada para que detecten al carro deslizante al momento de topar en los extremos. Se soldaron los cables a las terminales del switch para poder conectarlos al microcontrolador.

Figura 36: *Final de carrera*

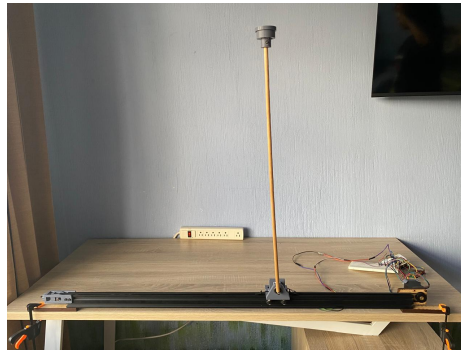


Elaboración propia

Por otra parte se fabricó con impresión 3D la pieza que sostiene el sensor magnético que mide la rotación en el eje del motor stepper. Esta pieza se conforma de una washa que centra el sensor en el soporte, el cual se acopla directamente en la parte trasera del motor. Se coloca el imán de 5 mm en la parte trasera del eje de manera que el sensor pueda detectar el campo magnético y medir la posición y velocidad angular del motor.

Al realizar todas estas correcciones se obtuvo el resultado del primer prototipo físico, el cual realizaba un movimiento suave y sin fricción tanto en el movimiento lineal del carro deslizante como en el movimiento oscilante del péndulo. En la Figura 37 se muestra el resultado del primer prototipo.

Figura 37: *Primer prototipo*

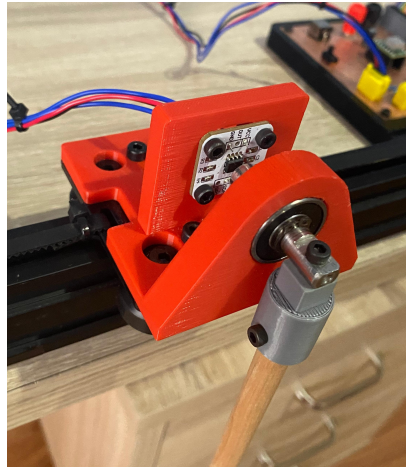


Elaboración propia

8.1.3. Fabricación del péndulo - Versión 2

Se identificaron distintas mejoras para la resistencia del sistema. Todo lo fabricado en MDF se imprimió en 3D con PLA, ya que luego de varias semanas el MDF se veía desgastado. Al utilizar PLA se mejora la resistencia de todo el sistema ya que en el laboratorio muchos estudiantes manipulan el péndulo y es propenso a dañarse fácilmente. Además se fabricó una pieza extra que sostiene al péndulo con el carro deslizante de mejor manera, el cual le da rigidez y seguridad al movimiento oscilante tal como se muestra en la 38. Se refabricó la base del carro deslizante ya que el cojinete se aflojaba constantemente. Se trabajó con un ajuste más cerrado, el cual proporciona más rigidez y seguridad ante movimientos bruscos del péndulo.

Figura 38: *Base de péndulo versión final*



Elaboración propia

Con los sargentos que se tenían en el primer prototipo, el péndulo topaba en ocasiones, lo cual afectaba seriamente al control. Entonces, se fabricaron dos sargentos con PLA de forma que permiten el movimiento completo del péndulo si en algún futuro se implementa *Swing up*. Por último, se fabricó una masa rectangular, la cual ayuda a hacer el movimiento completo, ya que la masa anterior también topaba con los sargentos. En la Figura 39 se muestra el prototipo final del péndulo invertido, el cual es totalmente funcional.

Figura 39: *Prototipo final*



Elaboración propia

8.2. Fabricación electrónica

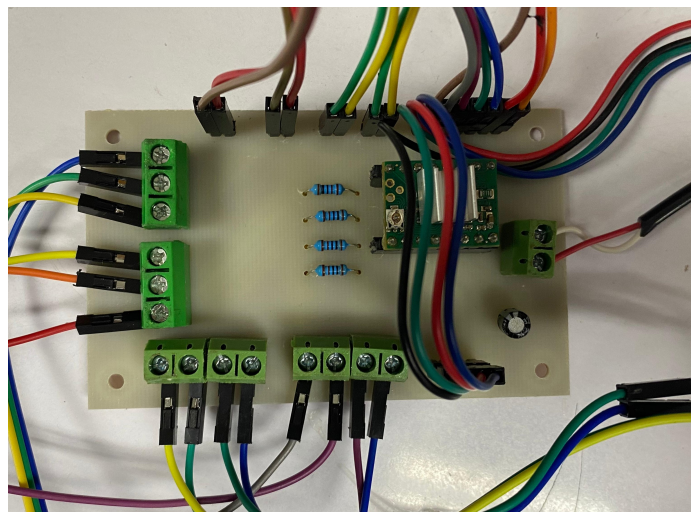
8.2.1. Fabricación electrónica - Versión 1

En primer lugar para probar la funcionalidad del sistema se realizaron todas las conexiones en un protoboard. Se conectaron las bobinas del motor stepper al driver TMC2130, asegurando que las terminales A1, A2, B1 y B2 estuvieran correctamente en su conexión. Luego se optó por alimentar el motor con una fuente de 12V que entrega máximo 2 amperios. Esto se realizó con un bloque terminal para garantizar que los cables nunca se toquen y hagan cortocircuito. Fue importante colocar el capacitor de 100 micro faradios el cual ayuda a proveer un voltaje con mayor estabilidad ante los picos y ruidos no deseados. Luego se realizaron las conexiones respectivas del SPI y la alimentación del driver. Se realizaron pruebas con la librería del TMC2130 para verificar que el motor trabajara correctamente, y respondiera ante distintos comandos de configuración con SPI.

Por otro punto, se conectaron los sensores magnéticos AS5600 por medio de *jumpers* largos dado que el carro deslizante se mueve a lo largo de todo el riel, lo cual debe tener la suficiente holgura para que no se desconecte el cable. Para la comunicación I2C se colocaron todas las resistencias respectivas en el protoboard. Se realizaron las pruebas respectivas con la librería de AS5600 para verificar que se obtuvieran lecturas correctas del ángulo y la velocidad angular de cada sensor. En el caso de los finales de carrera se colocaron *jumpers* largos en el extremo más alejado del circuito, verificando que la lectura digital se captara correctamente.

Luego de haber hecho pruebas con el control y calibración que se describen en los siguientes dos capítulos, se optó por fabricar la placa PCB. Se mandaron los archivos gerber del *bottom layer* y de los orificios al Maker Lab de la Universidad del Valle, y se obtuvo la placa que se muestra en la [40](#).

Figura 40: Placa PCB versión 1



Elaboración propia

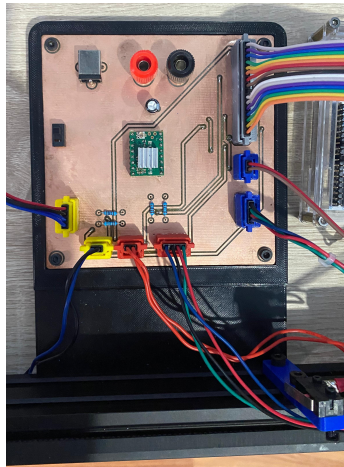
Se soldaron los *headers* y los bloques terminales, las cuatro resistencias y el capacitor en la placa PCB fabricada. Para identificar de mejor manera las conexiones en la placa se colocaron etiquetas para cada grupo de *headers* y bloques terminales. Para el driver TMC2130 se colocaron *headers* hembra para no soldarlo y poder utilizarlo en otras aplicaciones.

Para verificar el funcionamiento de la placa se realizaron las mismas pruebas con los ejemplos de las librerías del sensor AS5600 y el driver TMC2130. Para evitar que el estudiante confunda las terminales del voltaje de alimentación de 3.3V que viene del microcontrolador, se colocó una pieza en 3D, la cual evita cambiar la polaridad y dañar algún componente. En las esquinas de la placa se diseñaron orificios para tornillos M3, los cuales sujetarán a la placa con una base que se acopla al perfil de aluminio. De esta manera se podrá transportar todo el sistema con mayor facilidad.

8.2.2. Fabricación electrónica - Versión 2

Ya fabricada la placa PCB, se realizaron las mismas pruebas para el funcionamiento de la placa, midiendo continuidad y asegurando las conexiones correctas entre los sensores y el microcontrolador. Luego se fabricaron los conectores para cada pin, los cuales ayudan a darle orientación a la conexión para que no hayan confusiones en el orden de los pines. Se diseñaron conectores de 2, 4 y 16 pines. Para identificar de mejor manera cual conector corresponde al de la placa, se imprimieron en distintos colores, lo cual es más intuitivo para el estudiante ubicar la conexión, tal como se muestra en la Figura [41](#).

Figura 41: *Placa PCB versión final*



Elaboración propia

Se puede apreciar que es más intuitiva la conexión de cada sensor y los componentes de manera que el estudiante no se confunda en el laboratorio. También se imprimieron etiquetas en papel que se colocaron en la placa identificando cada conexión. Además se fabricó una pieza con PLA que fija la placa PCB con el riel del péndulo, de manera que sea más práctico transportar el sistema.

Técnicas de control para el sistema

En este capítulo se describe como se implementaron las técnicas de control en el micro controlador para hacer al sistema estable. Se implementaron dos técnicas, control clásico PID y control moderno con variables de estado. Es importante tomar en cuenta que el controlador a implementar es discretos, el cual tiene un periodo de muestreo, lo cual se debe programar en el microcontrolador adecuadamente.

9.1. Control clásico PID

Para implementar el control PID, primero es importante linealizar el sistema en el punto de operación deseado ya que el modelo del sistema no es lineal. Luego, se aplicará el sistema de control PID en el microcontrolador según las constantes determinadas.

9.1.1. Linealización del sistema

En primer lugar se consideran las ecuaciones planteadas según el diagrama de cuerpo libre presentada en la sección del marco teórico en la Figura 5. Dado que la varilla de madera tiene un peso despreciable, se considera que el centro de gravedad del sistema está sobre la masa, es decir en el extremo del péndulo. Esto hace que la inercia I sea insignificante, la cual se considera cero. Además, la fuerza de rozamiento del carro será tan pequeña que también se puede descartar, por lo que nos quedan las siguientes dos ecuaciones que describen a nuestro sistema físico. [11]

$$ml\ddot{\theta} + mg \sin \theta + m\ddot{x} \cos \theta = 0 \quad (12)$$

$$(M + m)\ddot{x} + ml\ddot{\theta} \cos \theta - ml\dot{\theta}^2 \sin \theta = u(t) \quad (13)$$

Para determinar la planta del sistema, es importante hacer una linealización sobre el punto de control deseado. En este caso será con θ igual a cero. En las ecuaciones [12](#) y [13](#) que describen al sistema, se aprecia que hay tres términos no lineales. Las partes no lineales del sistema de ecuaciones diferenciales deben linealizarse sobre el punto de operación, lo cual se realiza con la siguiente ecuación. **Control_System**

$$f(\theta) \approx f'(\theta_o)\delta\theta + f(\theta_o) \quad (14)$$

Cada expresión no lineal debe derivarse, y evaluarse en $\theta = 0$, de manera que se sustituye el resultado en las ecuaciones planteadas inicialmente. En las ecuaciones A continuación se presenta el procedimiento de la linealización para cada expresión.

$$f(\theta) = \dot{\theta}^2 \sin \theta \quad (15)$$

$$f(0) = \dot{\theta}^2 \sin \theta \Big|_{\theta=0} = 0^2 \sin 0 = 0 \quad (16)$$

$$f'(0) = 2\dot{\theta}\ddot{\theta} + \dot{\theta}^2 \cos \theta \Big|_{\theta=0} = 0 \quad (17)$$

$$\dot{\theta}^2 \sin \theta \approx 0 \quad (18)$$

$$f(\theta) = \cos \theta \quad (19)$$

$$f(0) = \cos 0 = 1 \quad (20)$$

$$f'(0) = -\sin 0 = 0 \quad (21)$$

$$\cos \theta \approx 1 \quad (22)$$

$$f(\theta) = \sin \theta \quad (23)$$

$$f(0) = \sin 0 = 0 \quad (24)$$

$$f'(0) = \cos 0 = 1 \quad (25)$$

$$\sin \theta \approx \delta\theta \quad (26)$$

De las expresiones linealizadas anteriormente se sustituyen en las ecuaciones [12](#) y [13](#), obteniendo el siguiente sistema de ecuaciones diferenciales ya linealizado en el punto de operación $\theta = 0$

$$(M + m)\ddot{x} + ml\delta\ddot{\theta} = u(t) \quad (27)$$

$$m\ddot{x} + ml\delta\ddot{\theta} - mg\delta\theta = 0 \quad (28)$$

De esta manera ya es posible plantear una función de transferencia de la planta. Despejando \ddot{x} en la ecuación 27 y sustituyendo en la ecuación 28 se obtiene lo siguiente.

$$Ml\delta\ddot{\theta} - g\delta\theta(M + m) + u(t) = 0 \quad (29)$$

Aplicando la transformada de La Place se obtiene la siguiente expresión para determinar la función de transferencia del sistema ya linealizado. 12

$$Mls^2\Delta\theta(s) + F(s) - g(M + m)\Delta\theta(s) = 0 \quad (30)$$

$$\frac{\Delta\theta(s)}{F(s)} = \frac{1}{-Mls^2 + (M + m)g} \quad (31)$$

9.1.2. Implementación del controlador PID

El sistema de control con PID toma en cuenta tres aspectos, el control proporcional, integral y derivativo. La salida del controlador PID viene dada por la siguiente ecuación:

$$u(t) = k_P e(t) + k_I \int_0^t e(\tau) d\tau + k_D \dot{e}(t) \quad (32)$$

Para obtener las constantes k_P , k_I y k_D del PID se trabajó con el *PID Tuner* de la herramienta de Matlab. Para esto es necesario conocer la planta del sistema. Tomando en cuenta que la masa del carro es de 0.1 kg, la masa del extremo es de 0.05 kg, y el largo de la varilla es de 0.65 m, se determinó que la planta es la siguiente:

$$G(s) = \frac{1}{-0.065s^2 + 1.472} \quad (33)$$

En esta herramienta fue modificando la robustez y velocidad del controlador para obtener una respuesta estable y eficiente. Las constantes k_P , k_I y k_D se van modificando a manera que el controlador tenga un funcionamiento exitoso. Las constantes seleccionadas con ayuda del *PID Tuner* fueron las siguientes:

$$k_P = -452 \quad (34)$$

$$k_I = -5254 \quad (35)$$

$$k_D = -9.74 \quad (36)$$

Sin embargo, se debe tomar en cuenta que hay que implementar un controlador digital, tomando en cuenta un periodo de muestreo de 1 milisegundo. Para implementar el control proporcional digital, se mide el error actual con respecto al ángulo deseado y se multiplica por una constante k_P . Este procedimiento se hace cada periodo de muestreo. El error actual

viene dado por la siguiente ecuación, donde r_k es la referencia, es decir cero, y y_k es la variable de salida, es decir el ángulo medido.

$$e(k\Delta t) = e_k = r_k - y_k \quad (37)$$

Luego, el controlador integral toma en cuenta el error acumulado a lo largo del tiempo, y lo multiplica por una constante k_I . Por lo tanto se trabaja con la siguiente ecuación en cada iteración para almacenar el error, donde E_k corresponde al error almacenado y e_k al error actual.

$$E_k = E_{k-1} + e_k \quad (38)$$

En el controlador derivativo se toma en cuenta la diferencia entre el error actual y el error anterior, y luego se multiplica por una constante k_D . La salida de este controlador viene dada por la siguiente ecuación, donde e_{k-1} corresponde al error de la iteración anterior, la cual es importante almacenarla.

$$\dot{e}(k\Delta t) \approx \frac{e_D}{\Delta t} = \frac{e_k - e_{k-1}}{\Delta t} \quad (39)$$

Una vez implementadas las expresiones anteriores en el microcontrolador, se ejecuta la siguiente ecuación con las constantes del PID.

$$u(t) \approx u_k = k_P e_k + k_I E_k \Delta t + k_D \frac{e_D}{\Delta t} \quad (40)$$

Para este controlador se implementó con un periodo de muestreo de 1 milisegundo para lograr estabilizar al péndulo. Se obtuvo un resultado coherente pero fue necesario modificar las constantes para mejorar la estabilización del sistema. Esto se describirá más adelante en el capítulo 10.

9.2. Control moderno

Para el control moderno se toman en cuenta las variables de estado del sistema. Para plantear el sistema en el espacio de estados, se toma en cuenta el desplazamiento angular del péndulo, la velocidad angular, el desplazamiento del carro y la velocidad de desplazamiento. Antes de aplicar el control es importante linealizar el sistema según el punto de operación deseado. Se trabajará el control del péndulo posicionado hacia arriba y posicionado hacia abajo.

9.2.1. Linealización del sistema

En primer lugar, es necesario plantear las cuatro variables de estado de la siguiente manera.

$$x_1 = \theta \quad (41)$$

$$x_2 = \dot{\theta} \quad (42)$$

$$x_3 = x \quad (43)$$

$$x_4 = \dot{x} \quad (44)$$

Para determinar la función del $f(x,u)$ se despejan las variables \ddot{x} y $\ddot{\theta}$ de las ecuaciones [12](#) y [13](#). Luego se plantean en función de las variables de estado para obtener las funciones respectivas.

$$\ddot{x} = \frac{mg \sin \theta \cos \theta - ml\dot{\theta}^2 \sin \theta - u}{m \cos^2 \theta - (M + m)} \quad (45)$$

$$\ddot{\theta} = \frac{u \cos \theta + ml\dot{\theta}^2 \sin \theta \cos \theta - (M + m)g \sin \theta}{ml \cos^2 \theta - (M + m)l} \quad (46)$$

$$\dot{x} = f(x, u) = \begin{bmatrix} x_2 \\ \frac{u \cos x_1 + mlx_2^2 \sin x_1 \cos x_1 - (M+m)g \sin x_1}{ml \cos^2 x_1 - (M+m)l} \\ x_4 \\ \frac{mg \sin x_1 \cos x_1 - mlx_2^2 \sin x_1 - u}{m \cos^2 x_1 - (M+m)} \end{bmatrix} \quad (47)$$

Es importante tomar en cuenta que antes de linealizar, se debe encontrar el punto de operación deseado x_{ss} y u_{ss} que hagan que la función $f(x,u)$ sea cero. [13](#) Tomando en cuenta que se desea que el ángulo $\theta = 0$, se obtiene el siguiente resultado.

$$\dot{x} = f(x_{ss}, u_{ss}) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (48)$$

$$x_{ss} = \begin{bmatrix} 0 \\ 0 \\ x_f \\ 0 \end{bmatrix} \quad (49)$$

$$u_{ss} = [0] \quad (50)$$

Se aprecia que x_3 no se encuentra en $f(x,u)$, lo cual tiene sentido ya que la posición final del carro se puede establecer en cualquier punto deseado. Para determinar las matrices linealizadas del sistema se debe derivar cada función respecto a cada variable de estado

y entrada del sistema. A continuación se presentan las expresiones derivadas respecto las variables de estado.

$$\begin{aligned} \frac{\delta f_2}{\delta x_1} &= \frac{(-u \sin x_1 + mlx_2^2 \cos x_1 - (M+m)g \cos x_1)(ml \cos^2 x_1 - (M+m)l)}{(ml \cos^2 x_1 - (M+m)l)^2} \\ &+ \frac{ml \sin 2x_1(u \cos x_1 + mlx_2^2 \sin x_1 \cos x_1 - (M+m)g \sin x_1)}{(ml \cos^2 x_1 - (M+m)l)^2} \end{aligned} \quad (51)$$

$$\frac{\delta f_2}{\delta x_2} = \frac{2x_2 m \sin x_1 \cos x_1}{m \cos^2 x_1 - (M+m)} \quad (52)$$

$$\begin{aligned} \frac{\delta f_4}{\delta x_1} &= \frac{(mg \cos 2x_1 - mlx_2^2 \cos x_1)(m \cos^2 x_1 - (M+m))}{(m \cos^2 x_1 - (M+m))^2} \\ &+ \frac{m \sin 2x_1(mg \sin x_1 \cos x_1 - mlx_2^2 \sin x_1 - u)}{(m \cos^2 x_1 - (M+m))^2} \end{aligned} \quad (53)$$

$$\frac{\delta f_4}{\delta x_2} = -\frac{2x_2 ml \sin x_1}{m \cos^2 x_1 - (M+m)} \quad (54)$$

Se plantean las matrices en función de las expresiones derivadas y se evalúan en el punto de operación x_{ss} y u_{ss} que se determinó anteriormente en las ecuaciones 49 y 50. Las matrices derivadas quedan de la siguiente manera. 14

$$A = \frac{\delta f}{\delta x} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{\delta f_2}{\delta x_1} & \frac{\delta f_2}{\delta x_2} & 0 & 0 \\ 0 & 0 & 0 & 1 \\ \frac{\delta f_4}{\delta x_1} & \frac{\delta f_4}{\delta x_2} & 0 & 0 \end{bmatrix} \Bigg|_{(x,u)=(x_{ss},u_{ss})}$$

$$B = \frac{\delta f}{\delta u} = \begin{bmatrix} 0 \\ \frac{\cos x_1}{ml \cos^2 x_1 - (M+m)L} \\ 0 \\ -\frac{1}{m \cos^2 x_1 - (M+m)} \end{bmatrix} \Bigg|_{(x,u)=(x_{ss},u_{ss})}$$

A continuación se presentan los resultados de las matrices linealizadas en función de los parámetros del sistema físico. De esta manera ya es posible aplicar un LQR que determine un vector de ganancia K para la retroalimentación por variables de estado en el sistema y poder estabilizarlo en el punto de operación. 15

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{(M+m)g}{Ml} & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -\frac{mg}{M} & 0 & 0 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 \\ -\frac{1}{Ml} \\ 0 \\ -\frac{1}{M} \end{bmatrix}$$

$$C = [1 \ 0 \ 0 \ 0]$$

9.2.2. Implementación de controlador moderno

Para el control por variables de estado, se hace una retroalimentación negativa del error en cada variable de estado. Este error es multiplicado por una constante, logrando la estabilización de todas las variables de estado. La salida del controlador u viene dada por la siguiente ecuación:

$$u = -K(x_{ss} - x) \quad (55)$$

Para determinar una correcta selección del vector K es posible utilizar dos métodos: pole placement y LQR [16]. A continuación se describen las dos metodologías realizadas.

Pole placement

Para el pole placement es importante analizar los polos del sistema, ya que es un sistema inestable. El objetivo de este método es establecer los polos del sistema al lado izquierdo del plano imaginario. Con los eigen valores de la matriz se obtienen los polos del sistema:

$$\begin{bmatrix} 4.7580 \\ -4.7580 \\ 0 \\ 0 \end{bmatrix}$$

Se aprecia que hay un polo positivo, lo cual indica que el sistema es inestable, por lo tanto la retroalimentación debe hacer que los polos sean negativos. Por lo tanto se aplica el pole placement utilizando la herramienta de Matlab para colocar los polos en el lado negativo del plano. Se escogieron los polos -5, -4, -1 y -1.5. Utilizando la herramienta de Matlab se obtiene el siguiente resultado del vector K :

$$K_1 = [-4.3961 \ -0.8842 \ 0.0994 \ 0.2104]$$

Se hace el mismo procedimiento para el péndulo posicionado hacia abajo, y se obtiene el siguiente vector K .

$$K_2 = [1.2593 \ 0.4740 \ -0.1988 \ -0.4207]$$

LQR

El método de LQR, se enfoca en obtener un vector K óptimo que logre estabilizar las variables de estado. Para esto es importante plantear una matriz de penalización que prioriza las variables de estado en el sistema. La matriz de penalización que se decidió implementar fue la que se presenta a continuación:

$$Q = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$R = [1]$$

Con estas matrices de penalización, se trabajó con la función de Matlab LQR y se obtuvieron los siguientes vectores para controlar las variables de estado posicionando el péndulo hacia arriba y hacia abajo respectivamente.

$$K_1 = [-13.5022 \quad -2.7286 \quad 1.0000 \quad 1.4921]$$

$$K_2 = [3.0979 \quad 0.4786 \quad -1.0000 \quad -1.3898]$$

Para implementarlo en el microcontrolador, se realizaron las lecturas de las variables de estado a 1 milisegundo como periodo de muestreo, luego se calculó el error actual de la variable de estado respecto a X_{ss} , se multiplicó por las constantes del vector K y se envió a la salida del controlador. Dado que el sistema físico se puede comportar de forma distinta, en el siguiente capítulo se describe el procedimiento de la calibración de estas constantes.

10.1. Configuración del *Motor Stepper*

Como primer punto, fue importante configurar el motor stepper con el driver TMC2130. Este driver del motor se comunica a través de SPI con el microcontrolador, el cual tiene la capacidad proveer distintas configuraciones al motor. Por lo tanto fue necesario conectar el SDI (*serial clock*), SDI (*serial data in*), SDO (*serial data out*) y CS (*chip select*) Se instalaron las librerías del TMC2130 las cuales permiten un mejor manejo del driver.

Primero, existe la opción de configurar la corriente que utiliza el motor, el cual utiliza de forma estándar 600 mili amperios. El motor utilizado es un NEMA 17HS4023, el cual presenta una curva de velocidad torque. La corriente nominal para el motor es de 2A a 12V, sin embargo se configuró con 800 mili amperios para no saturar la fuente de voltaje. La fuente de voltaje utilizada provee 12V con 2 amperios como máximo.

Se configuró el motor para que actuara en modo silencioso, con una configuración de 16 micro steps, lo cual indica que cada 16 microsteps, el eje del motor avanza 1.8°. También se configuró el (*Stall Guard*), para que el mismo driver indicara el torque que presenta el motor al momento de estar funcionando.

Para mover el motor en una dirección, la librería tiene la función de establecer la velocidad del eje, las cuales se establecen en steps por milímetros. Se empezó probando el rango de velocidades que soporta el motor sin que entre en falla. Se logró determinar que el motor soporta hasta una velocidad de 45,000 steps/s, sin embargo cuando se conecta el motor para movilizar la faja, la velocidad máxima que soporta es de 32,000 steps/s. Es necesario mandar el comando `stepper.run()` al motor para que ejecute los pulsos al motor. Realizando varias pruebas para poner en funcionamiento al motor, se determinó que el motor tiene un mejor funcionamiento en una interrupción que se ejecuta cada 2 micro segundos, ya que al

momento de querer implementar otras funciones en el loop, se interrumpía el funcionamiento del motor, y ya no funcionaba correctamente.

Ya configurado de esta manera, es posible mandarle una velocidad al motor desde -32,000 a 32,000 steps/s. Este será el rango de salida del controlador PID, por lo que fue necesario acotarlo para que no sobrepasara estos límites. Por otra parte, es posible configurar la aceleración del motor. Se determinó que una aceleración de 150,000 pasos/s² presentaba una respuesta adecuada para actuar el motor, ya que para estabilizar el péndulo se necesita una respuesta inmediata en los cambios de velocidad.

10.2. Configuración de sensores magnéticos

Como segundo punto fue necesario configurar el sensor magnético AS5600 para que pueda leer el ángulo del péndulo. Este sensor se comunica al microcontrolador con el protocolo de comunicación I2C. Se instalaron las librerías respectivas desde la plataforma de Arduino para poder inicializar y configurar los sensores. Se logró obtener la lectura angular del péndulo con un rango de 0 a 360°. El péndulo colocado hacia arriba tiene un desfase de aproximadamente 96°. Se creó una función para poder ingresar el desfase angular de la posición deseada considerándolo como ángulo 0° y que haga lecturas de -180° a 180°. De esta forma, se podrá realizar la retroalimentación negativa del PID, para estabilizar el sistema a un ángulo de 0°. Es importante tomar en cuenta que si se desea estabilizar el péndulo hacia abajo, se debe colocar el desfase respectivo, que es aproximadamente 276°. Al momento de hacer el control, pequeñas variaciones en el desfase puede afectar significativamente a la estabilización del péndulo. Por esta razón es importante calibrar el offset angular correctamente antes de comenzar el control.

Luego, para obtener las lecturas de las velocidades angulares de cada sensor, se tuvo el problema que no realizaba lecturas correctas de velocidades altas. Esto se debe a que el sensor realiza una comparación entre el ángulo actual respecto al ángulo anterior, y con esto determina una velocidad angular. Si la frecuencia de muestreo es demasiado corta, entonces a velocidades altas ya no realiza lecturas correctas. Para solucionar esto, se implementó una segunda interrupción únicamente para realizar las mediciones de los sensores. Se implementó un periodo para la interrupción de 1 milisegundo, lo cual no da problemas para velocidades altas. Esta interrupción debe ser independiente a la del motor, ya que las lecturas tienen una latencia relativamente alta, lo cual afecta a la actuación del motor.

Fue importante establecer niveles de prioridad en las interrupciones para obtener un funcionamiento exitoso del control. La interrupción con mayor prioridad es el envío de steps al motor, y luego es la interrupción de las lecturas de los sensores. De esta forma el motor tiene un movimiento más suave y sin pausas que afecten al sistema.

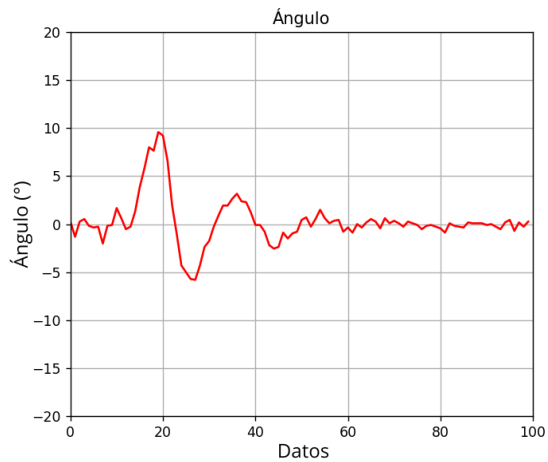
10.3. Calibración de control PID

Luego de configurar el motor, y poder obtener lecturas correctas del ángulo, se implementó el PID discreto. Para las constantes kP, kD y kI, se toma en cuenta el periodo de control de 1

milisegundo. Dado que el sistema no es lineal, el PID únicamente funciona para la localidad cercana al punto de operación, es decir para ángulos muy pequeños. Por lo tanto al comenzar el control PID se debe colocar el péndulo hacia arriba o hacia abajo, dependiendo del control que se desea.

En la primera prueba se trabajó con el péndulo hacia abajo, para mayor seguridad, de manera que se estabilizara con una oscilación inicial pequeña, cercano al punto de operación. Se colocaron las constantes del control PID calculadas anteriormente y se obtuvo el resultado que se muestra en la Figura 42.

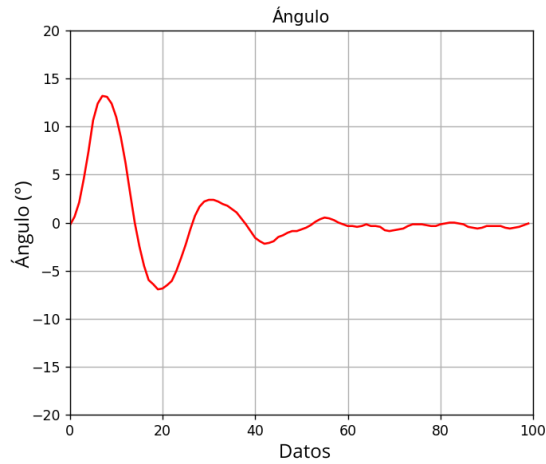
Figura 42: Resultado 1 PID péndulo hacia abajo



Elaboración propia

Se puede observar que en el comportamiento transitorio se exhiben oscilaciones iniciales que luego tienden a estabilizarse alrededor del punto de operación. A pesar que lograba estabilizar el péndulo a un ángulo de cero, el sistema presentaba movimientos muy bruscos. Por lo tanto se fueron modificando las constantes del PID hasta encontrar una respuesta más suave, con un movimiento transitorio similar. Se comenzó con una reducción de las ganancias de las tres variables, lo cual reduce el sobre impulso inicial de querer estabilizar el sistema. Se escogieron las constantes de 100, 0.1 y 0.1, para el controlador PID respectivamente, obteniendo el resultado que se observa en la Figura 43.

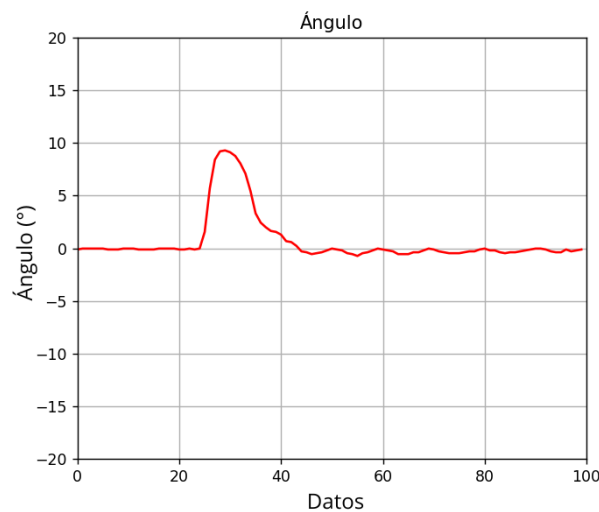
Figura 43: Resultado 2 PID péndulo hacia abajo



Elaboración propia

En la gráfica anterior se aprecia que hay una mejora significativa en la estabilización del sistema, realizando un movimiento menos brusco. Sin embargo, se aprecia que sigue presentando *overshoot*, lo cual no es una respuesta adecuada para la estabilización del péndulo. Se modificaron varias veces las constantes hasta encontrar una respuesta exitosa, la cual se presenta en la Figura 44. Se fue probando aumentar levemente el control proporcional para encontrar una respuesta más rápida ante el error actual. Las constantes derivativa e integral no contribuyeron significativamente a mejorar el control. Se seleccionaron las constantes de 400, 0.1 y 0.1 para el controlador PID respectivamente, lo cual presente el siguiente resultado.

Figura 44: Resultado 3 PID péndulo hacia abajo



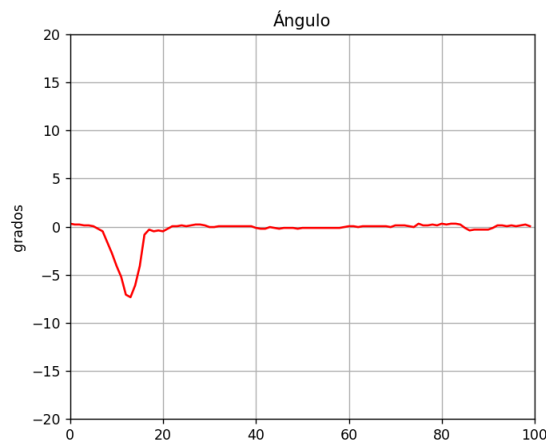
Elaboración propia

En este último resultado se puede observar que el sistema logró una estabilización exitosa, sin *overshoot*, en un tiempo óptimo. Este control es menos propenso a las perturbaciones externas y más robusto. Aunque el PID estabiliza el péndulo, el carro deslizante se desplaza hacia los extremos. Esto se debe a que no se estaban usando las variables de estado, que toman en cuenta la posición del carro deslizante. El PID únicamente tiene la funcionalidad de estabilizar el ángulo del péndulo, más no estabiliza la posición del carro deslizante. Como resultado de esto, el carro puede seguir desplazándose hasta topar con el extremo y hacer que el péndulo se salga de su rango de estabilización. Por esto fue importante trabajar con variables de estado, lo cual se describe más adelante.

En el segundo caso, se trabajó con el péndulo invertido hacia arriba, con las mismas constantes. El péndulo parecía estabilizarse pero un problema en el movimiento no permitía actuar correctamente el sistema. Habían varios factores que afectaban el movimiento fluido del péndulo. Primero, la polea del motor no estaba correctamente centrada, lo cual hacía que la faja topara con el perfil de aluminio. Luego los cojinetes V no estaban correctamente apretados para que fluyera libremente el péndulo. Se verificó que el carro estuviera correctamente ajustado colocando el perfil a 45° , permitiendo que se deslizara libremente a lo largo del riel.

Después se analizó que la barra de aluminio era muy pesada para que el motor pudiera mover con facilidad el sistema, por lo que mejor se utilizó una varilla de madera del mismo diámetro y largo, lo cual mejoró considerablemente el comportamiento del sistema. Se pudo apreciar que entre menos pesada es la varilla del péndulo, mejor control tiene el sistema. Luego se probaron las constantes teóricas, pero el movimiento era demasiado brusco, por eso se optó por modificar las variables del PID, lo cual fue mejorando el funcionamiento significativamente. El resultado obtenido con las constantes 15000, 0, 0 fue el de la Figura 45.

Figura 45: Resultado 4 PID péndulo hacia arriba



Elaboración propia

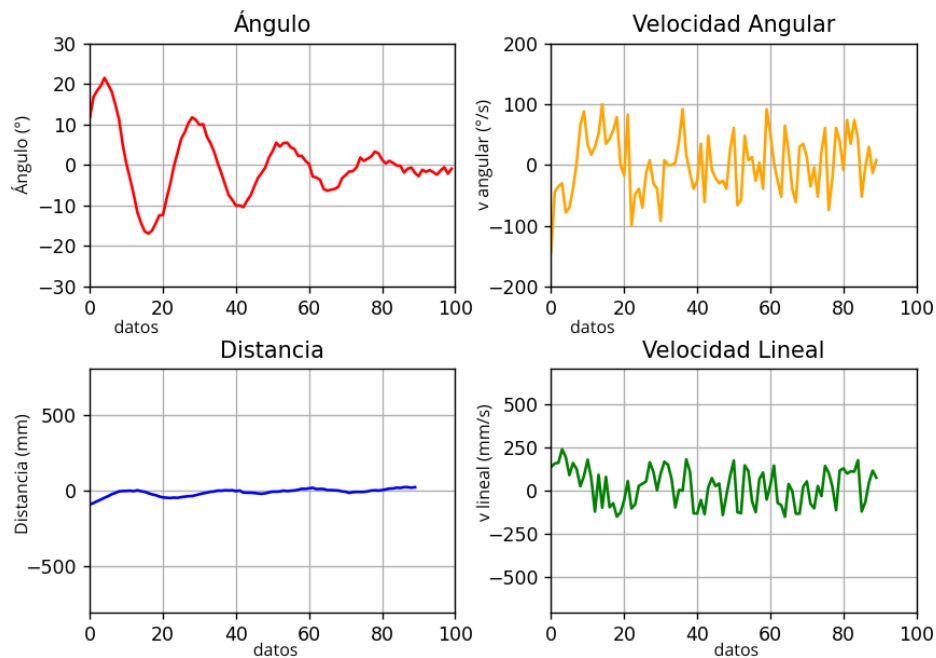
En comparación al sistema donde el péndulo está hacia abajo, este sistema es más vulnerable ante las perturbaciones externas. Se puede apreciar que el control PID tiende a estabilizar el péndulo en el punto de operación deseado, sin embargo ante pequeñas

perturbaciones, el sistema busca estabilizar el péndulo pero se desplaza bruscamente hacia los extremos. Esto se debe a que no se han implementado las variables de estado, en donde se toma en cuenta estabilizar la posición del carro deslizante. En la siguiente sección se estabiliza el sistema con variables de estado, en donde el control es más eficiente.

10.4. Calibración de control con variables de estado

En este caso se implementó el control por variables de estado, tomando en cuenta el ángulo, velocidad angular, desplazamiento y la velocidad lineal. En primer lugar, se estabilizó el péndulo hacia abajo, utilizando el vector K calculado con LQR anteriormente. En la Figura 46 se muestran los resultados con el valor teórico del vector K . Se puede apreciar que no se logró una correcta estabilización de las variables de estado. Tanto el ángulo como el desplazamiento lineal se lograron estabilizar en el punto de operación, pero la velocidad angular y lineal tienen una respuesta con bastantes perturbaciones.

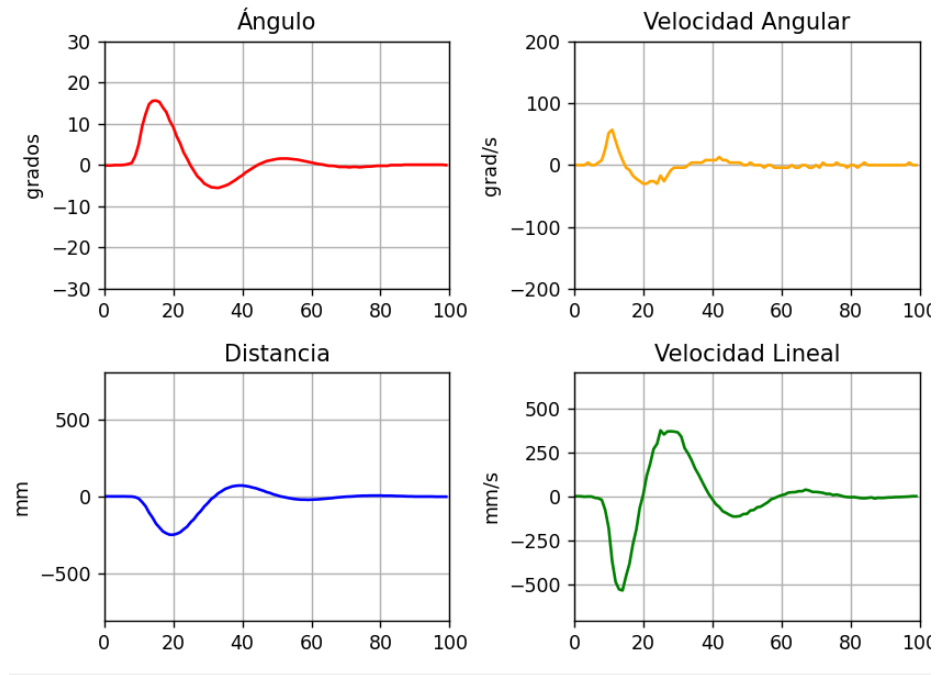
Figura 46: Resultado 1 con variables de estado – péndulo hacia abajo



Elaboración propia

A lo largo de la respuesta transitoria se pudo observar un movimiento muy brusco del sistema por lo que se optó por disminuir los valores del vector K para obtener una respuesta más suave como se muestra en la Figura 47. Se seleccionaron las constantes 2000, 20, 80 y 0.4 respectivamente del vector K para las variables de estado, las cuales proporcionaron un comportamiento adecuado del sistema.

Figura 47: Resultado 2 con variables de estado – péndulo hacia abajo

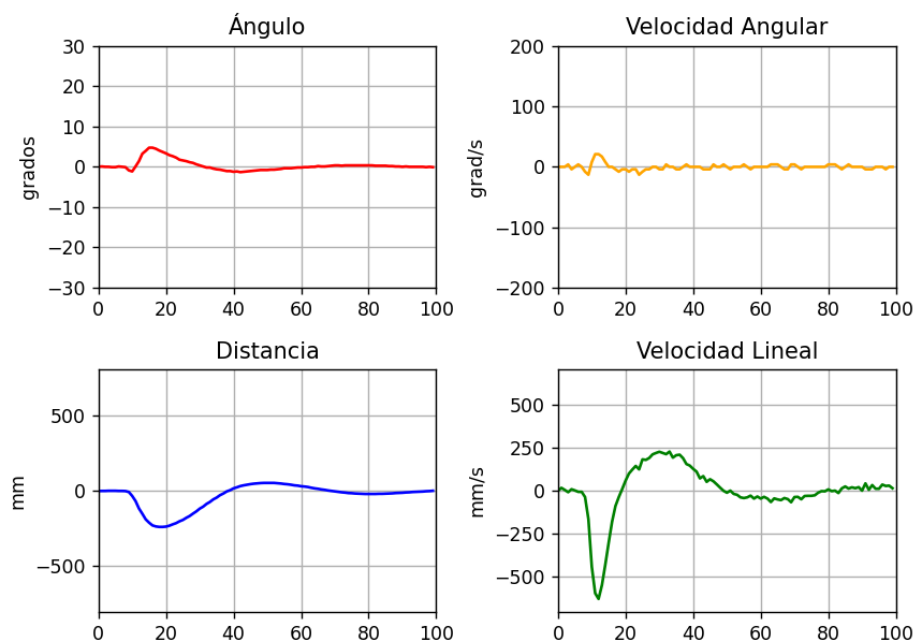


Elaboración propia

En los gráficos anteriores se puede apreciar que el sistema de control por variables de estado tiene un comportamiento exitoso, en donde estabiliza las cuatro variables en un tiempo óptimo. El sistema es capaz de estabilizar el ángulo del péndulo al mismo momento que estabiliza la posición del carro, donde es posible establecer la posición final del carro en cualquier punto a lo largo del riel. En el vector de ganancia se buscó un equilibrio entre la estabilización del ángulo y el desplazamiento lineal, dando mayor prioridad al ángulo del péndulo, ya que si se da prioridad a la posición, el ángulo no se logra estabilizar de manera correcta.

Por otro punto, para estabilizar el péndulo hacia arriba fue más complejo. En un principio se utilizaron los valores teóricos del vector K para controlar el sistema pero se obtuvo un sistema divergente, en donde los movimientos bruscos hacían que el péndulo no lograra estabilizarse adecuadamente. Se fueron modificando los valores del control a manera que el péndulo se mantuviera estable y se logró obtener los resultados de la Figura 48. Las constantes calibradas del vector K para lograr los siguientes resultados fueron 1500, 200, -15 y -50 respectivamente.

Figura 48: Resultado 1 con variables de estado – péndulo hacia arriba



Elaboración propia

En los resultados anteriores, se puede observar que el sistema de control es exitoso, el comportamiento del sistema ya no diverge como normalmente lo hacía sin el control de forma natural, o con el control PID, donde el carro se desplazaba hacia los extremos. Las cuatro variables de estado convergen al punto de operación deseado a pesar de las perturbaciones externas que se pueden realizar al péndulo invertido. Analizando el comportamiento del ángulo se aprecia que el ángulo de atracción puede llegar a ser de hasta 10 grados, donde converge de forma rápida. El sistema de control primero da prioridad a la estabilización del ángulo y luego busca estabilizar la posición del carro en el punto seleccionado.

A pesar de las perturbaciones externas, se aprecia que la velocidad angular del sistema siempre se mantiene muy cercana a cero. Además la velocidad lineal del carro se estabiliza de forma exitosa. Se observa que debido al movimiento constante que debe realizar el carro para estabilizar el péndulo, se presenta un poco de ruido en las velocidades angular y lineal.

Es importante tomar en cuenta que el tiempo de muestreo debe ser lo más mínimo posible para reducir la latencia en la toma de decisiones del sistema, lo cual mejor el rendimiento general del sistema. El control preciso del sistema depende de muchos factores, como la calibración exacta del ángulo, el tiempo de muestreo, el microstep con que realiza la actuación el motor, entre otros. Esta plataforma permite modificar distintas variables y configuraciones para que el estudiante pueda visualizar y experimentar como se ve afectado el sistema de control el tiempo real.

Desarrollo de interfaz para control del péndulo

En el presente capítulo se describe el proceso de desarrollo de la interfaz que permite al usuario configurar y controlar el péndulo de forma más eficiente y comprensible. Se utilizó la herramienta de Python, junto con la biblioteca gráfica Tkinter [17] para realizar la interfaz. Tkinter es la interfaz que trae por defecto Python para el grupo de herramientas GUI (*Graphical User Interface*), el cual ofrece un conjunto amplio de widgets, como etiquetas, gráficas, botones y pestañas para realizar interfaces personalizadas y complejas.

11.1. Especificaciones de interfaz

En primer lugar, se definieron especificaciones de diseño antes de iniciar el desarrollo de la interfaz. Esto es importante para poder definir cuál será su funcionalidad, interacciones del usuario esperado y requisitos principales a cumplir. De esta forma se puede tener claridad de los objetivos a alcanzar, comprendiendo las necesidades y expectativas del usuario al momento de querer controlar el péndulo.

- **Compatibilidad de comunicación:** La interfaz debe ser compatible para comunicarse a través del puerto serial con el microcontrolador para mandar o recibir los parámetros necesarios para el control.
- **Actualización en tiempo real:** La interfaz debe ser capaz de mostrar en tiempo real a través de gráficas las cuatro variables de estado del sistema: ángulo del péndulo, velocidad angular, distancia recorrida y velocidad lineal, con un intervalo de actualización que permita una respuesta ágil y precisa.
- **Control de parámetros del sistema:** Los usuarios deben poder modificar los parámetros del sistema en tiempo real como las constantes del k_P , k_I , k_D para el controlador PID,

así como las constantes K1, K2, K3, K4 del controlador por variables de estado.

- Personalización en la configuración: La interfaz debe ser capaz de permitir la personalización de los parámetros de configuración como el amperaje del sistema, el microstep del motor, la aceleración, el tiempo de muestreo, entre otros. Esto también incluye la capacidad de almacenar las configuraciones actuales para que se muestren en otras pestañas.
- Diseño intuitivo y coherente: La interfaz debe ser fácil de usar, con una apariencia ordenada y limpia, colocando los elementos de forma lógica. Se debe priorizar la facilidad de acceso a las funciones importantes, minimizando la necesidad de consulta y entrenamiento. La interfaz debe contar con distintas pestañas para poder interactuar de manera distinta con el péndulo, según sea la necesidad del usuario.m

11.2. Desarrollo de interfaz

Como primer punto se implementó la comunicación serial entre el microcontrolador y la interfaz, en donde se enviaron las cuatro variables de estado a través del formato texto JSON (*Java Script Object Notation*) [18]. Este formato consiste en cadenas de datos estructurados para intercambiar información entre distintos lenguajes de programación. Este formato se serializa y deserializa tanto en el microcontrolador como en Python, de manera que es más eficiente comunicarse entre sí.

Se enviaron los datos de las variables de estado para decodificarlas en Python y almacenarlas en variables. Se crearon gráficas para cada variable, que al momento de recibir los datos desde el microcontrolador, se van actualizando en tiempo real. Se buscó que el tiempo de transmisión fuera adecuado para que hubiera suficiente tiempo de actualización de la animación en la interfaz. Se determinó que 70 mili segundos era suficiente para mostrar una animación de la gráfica de forma suave como se muestra en la Figura 49. Si se coloca una frecuencia de envío mayor, la visualización de las variables se va atrasando en el tiempo, dado que Python llega a un límite para poder recibir los datos y actualizarlos de forma inmediata.

Luego se crearon tres pestañas: configuración, controlador PID y controlador por variables de estado. Cada vez que se cambia de pestaña, la interfaz envía un comando a través del puerto serial para indicar en que modo debe estar el microcontrolador, ya sea en configuración o en control PID o variables de estado. Dependiendo del modo, el microcontrolador actúa el motor de distinta forma, según las variables que se envían. A continuación se explican cada uno de las pestañas desarrolladas.

11.2.1. Pestaña de configuración

En esta pestaña, el usuario es capaz de modificar parámetros que afectan al rendimiento del péndulo, tanto en la actuación del motor, como en las lecturas de los sensores. El objetivo de esta pestaña es que el usuario pueda calibrar el péndulo para que las lecturas del ángulo

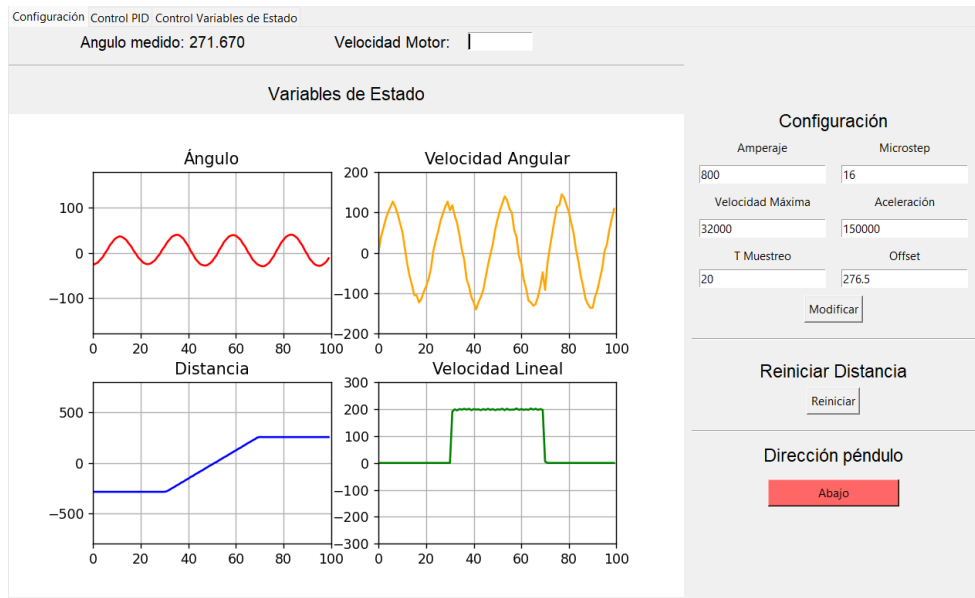
y la posición inicial sean adecuadas con el control a realizar. A continuación se describen las variables que permite modificar la interfaz en la pestaña de configuración.

- Amperaje: el usuario es capaz de configurar el amperaje del motor, dependiendo si quiere utilizar otros modelos. La corriente que está por defecto es 800mA, la cual proporciona buen rendimiento al motor NEMA 17HS19-2004S1.
- Microstep: el usuario puede de modificar el microstep del motor, el cual se encuentra en valores posibles de 2, 4, 8, 16 y 32. Este valor indica cuantos microsteps tiene que hacer el motor para que avance 1.8° . Es decir que tiene una resolución de hasta $1.8/32$ grados que equivale aproximadamente a 0.056° , lo cual es muy preciso. Esto ayuda a que el motor pueda brindar al péndulo un control más exacto.
- Tiempo de muestreo: es posible modificar el tiempo de muestreo de las lecturas del ángulo y de la ejecución del PID. De esta manera el usuario será capaz de visualizar como se comporta el sistema con distintas frecuencias de muestreo.
- Velocidad máxima: el usuario tendrá la posibilidad de modificar la velocidad máxima que soporta el motor, ya que el motor tiene un limite en donde el driver manda los pasos, y ya no gira de forma correcta.
- Aceleración: el driver también tiene la opción de modificar la aceleración con que responde le motor. En este caso la aceleración por defecto es de 150000 microsteps por segundo al cuadrado, lo cual es una respuesta inmediata del motor para cambiar a velocidades abruptas.
- Offset: el sensor como tal lee ángulos de 0 a 360° , sin embargo se desean mediciones de -180° a 180° , definiendo un offset angular que se toma como ángulo cero. La interfaz también permite modificar el offset del ángulo medido en tiempo real, ya que cualquier alteración física en el eje del péndulo afecta considerablemente la medición del ángulo. Como esta medida debe ser muy precisa para realizar el control, entonces se empleó esta configuración, en donde el usuario podrá visualizar el ángulo medido por el sensor y calibrarlo de forma práctica.

Por otra parte se tiene un botón de arriba/abajo, el cual permite establece en que posición se desea hacer el control, ya sea con el péndulo invertido o con el péndulo hacia abajo. Además se agregó otro botón que resetea la posición inicial del carro en dado caso se quiera establecer el origen de la posición en otro punto. Por ejemplo, si actualmente se inició el carro en el extremo izquierdo y se quiere que el cero sea el centro del riel que se encuentra a 200 mm, se mueve el carro y se resetea la posición para que sea su nueva referencia cero.

Otra funcionalidad importante de esta pestaña es el control del carro deslizante. A través de una entrada de texto se puede definir la velocidad del motor para mover el carro deslizante. Esta función es muy importante para realizar pruebas del movimiento y velocidad del carro, así también como establecer posiciones iniciales en distintos puntos a lo largo del riel.

Figura 49: Pestaña de configuración



Elaboración propia

11.2.2. Pestaña de control PID

En esta pestaña el usuario tiene la posibilidad de controlar el péndulo a través de un PID. El controlador PID (*Proporcional-Integral-Derivativo*) es una de las técnicas más utilizadas, el cual se basa en ajustar la salida del sistema según el error entre el valor deseado y el valor medido. La salida viene dada por un conjunto de constantes k_P , k_I , k_D , las cuales logran una salida estable. Esta es una funcionalidad esencial para optimizar el rendimiento del sistema en la estabilización del ángulo del péndulo.

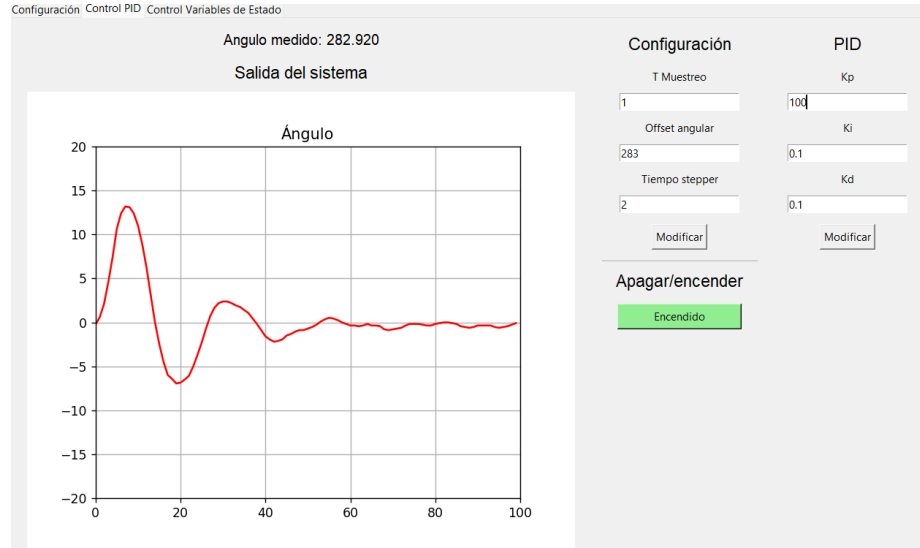
Esta pestaña se divide principalmente en dos secciones. La primera sección del lado izquierdo muestra la gráfica en tiempo real del ángulo del péndulo, lo cual permite monitorear como responde la salida ante el control aplicado. La segunda sección ubicada en el lado derecho corresponde a los campos de entrada k_P , k_I , k_D , los cuales se pueden modificar en tiempo real mediante un botón de modificar situado en la parte inferior como se muestra en la Figura 50.

Por otra parte, para que sea más seguro y práctico el uso de esta pestaña, se utilizó un botón de encendido/apagado, el cual facilita al usuario pausar el motor de forma inmediata si existe algún problema y no tener que desconectar el microcontrolador. En dado caso este funcionando el control, y se cambia de pestaña, se apaga inmediatamente el control.

Para que el usuario no tenga que regresar a la pestaña de configuración, se colocaron dos parámetros importantes de la configuración: el tiempo de muestreo y el offset angular. Esto es esencial dado que el usuario podrá modificar los parámetros en tiempo real para calibrar el sistema. El offset angular debe ser muy preciso, y por motivos de movimientos bruscos, la lectura se altera significativamente, entonces se calibra con el offset. El tiempo de

muestreo cambia el comportamiento del sistema, lo cual también es importante ir calibrando mientras se ejecuta el control. Todos los datos se mandaron mediante el formato JSON para una mejor eficiencia de comunicación.

Figura 50: Pestaña control PID



Elaboración propia

11.2.3. Pestaña de control por variables de estado

En esta pestaña el usuario es capaz de controlar el péndulo invertido con las cuatro variables de estado, las cuales son ángulo del péndulo, velocidad angular, posición y velocidad lineal. El objetivo es poder estabilizar estas variables en el menor tiempo posible. El usuario podrá modificar las constantes del control por variables de estado en tiempo real para ver el comportamiento de todo el sistema.

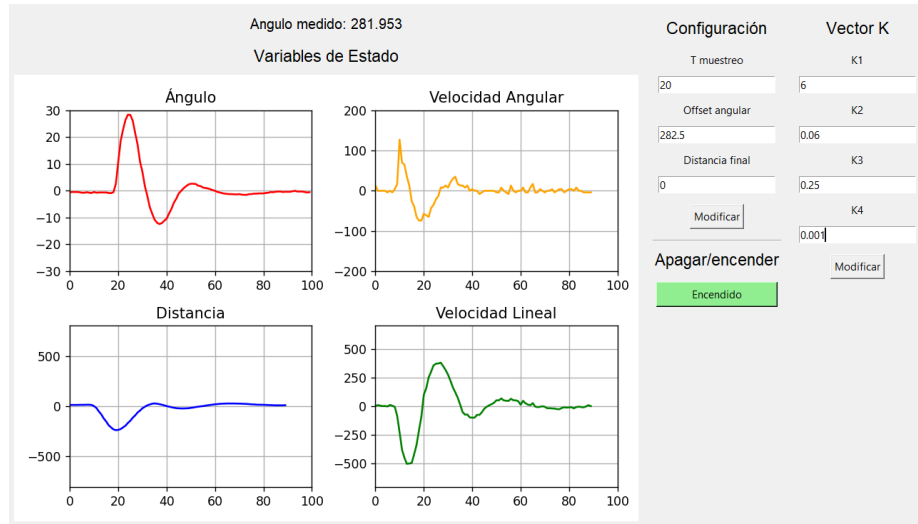
Para tener una mejor visualización del comportamiento del sistema a lo largo del tiempo, se emplean cuatro gráficas similares a la pestaña 1. En este caso como el ángulo busca estabilizarse a una localidad más cerrada, se grafica en un rango de -20° a 20° .

Para la posición del carro deslizante se graficó en un rango de -600 mm a 600 mm, ya que el usuario tiene la posibilidad de configurar los extremos como posición inicial. Dado que es un control por variables de estado, es posible indicar la posición final deseada para el carro deslizante, por lo tanto se colocó una entrada de texto para que el estudiante pueda controlar las posición del carro en tiempo real.

Para la variable de velocidad angular, se realizaron pruebas girando el péndulo rápidamente, y se determinó que alcanza entre un rango de $200^\circ/\text{s}$ a $-200^\circ/\text{s}$, por lo que se colocó de esta forma en la gráfica. Mientras que en la velocidad lineal, se desplazó el carro deslizante a grandes velocidades para verificar el rango de velocidad. Se determinó un rango de velocidad para graficar entre $-600\text{mm}/\text{s}$ a $600\text{mm}/\text{s}$.

De la misma manera que la pestaña anterior se colocaron entradas de texto para que el estudiante pueda modificar las constantes del vector K que estabiliza las variables de estado en el sistema. Estas variables de estado se pueden modificar en tiempo real para que se vea el cambio del comportamiento en la estabilización del péndulo. También se coloca un botón de encendido y apagado en dado caso resulte algún problema. Dado que la lectura del ángulo debe ser muy precisa, se colocó una entrada de texto para que el usuario pueda modificar el offset angular en tiempo real y se estabilice el péndulo. 51

Figura 51: Pestaña control variables de estado



Elaboración propia

11.3. Retroalimentación de usuarios

Para obtener una interfaz optimizada, se realizaron demostraciones a distintos usuarios de la carrera de Ingeniería Mecatrónica de la Universidad del Valle, y se obtuvo la retroalimentación sobre aspectos a mejorar. Esta retroalimentación ayuda a alcanzar los requerimientos establecidos inicialmente, y poder brindar al usuario una experiencia más intuitiva con la interfaz.

Como primer punto, se identificó el problema que al momento de modificar las variables en los campos de texto, no se distingue si ya se modificaron en el microcontrolador o solo en la interfaz. Por lo tanto, se empleó un código de color que cuando el usuario modifique un valor en el campo de texto, cambie a color amarillo y al presionar modificar, regrese a su valor original. De esta manera, el usuario podrá distinguir si ya se hizo el cambio.

Además se solicitó por los usuarios colocar un botón de ayuda, que muestre una pestaña en donde se indiquen instrucciones de como utilizar cada pestaña. Esto se hizo con el objetivo de garantizar un uso adecuado de la interfaz y mejorar la comprensión del usuario sobre su uso. Se describen una serie de pasos en cada pestaña para que se comprenda en que consiste cada entrada de texto y como operar los distintos elementos de la pestaña.

Además, con el fin de garantizar la integridad y la coherencia de los datos ingresados

por los usuarios en la interfaz de control, se implementó la funcionalidad de validación de caracteres en los campos de entrada. Esto restringe que el usuario pueda ingresar caracteres en los campos destinados a configuraciones con valores numéricos. Si el microcontrolador recibe caracteres en los parámetros de entrada, podrían haber serios errores con el funcionamiento. Esta funcionalidad mejora la robustez del sistema y reduce la posibilidad de errores por parte del usuario, asegurando que siempre se manden valores numéricos a través de la comunicación serial.

Por otro lado, se tenía el problema que al iniciar la interfaz, si no había un dispositivo conectado, se generaban errores de lectura en el serial. Para solucionar este problema y facilitar la conexión entre la interfaz de usuario con el microcontrolador, se agregó una funcionalidad que permite una selección dinámica del puerto de comunicación serial (COM). Se da la opción de escoger a través de un menú desplegable el puerto COM correspondiente al dispositivo conectado. Una vez seleccionado el puerto COM y establecida la conexión, la interfaz recibe los datos y automáticamente se actualizan los gráficos.

Finalmente, un punto muy importante a agregar en la interfaz, fue incluir dimensionales. Se implementaron unidades dimensionales asociadas a cada uno de los campos de entrada en las pestañas. También se incluyeron en las dimensionales en los ejes verticales de las gráficas. Estas unidades permiten que los datos sean interpretados correctamente por el usuario, asegurando que los valores ingresados sean coherentes con el sistema físico.

- Se diseñó y fabricó un sistema de péndulo invertido que cumple con los requisitos mecánicos y electrónicos, integrando componentes que permiten el control para estabilizar el péndulo.
- Se logró estabilizar el péndulo invertido a través de un controlador PID, tomando en cuenta la respuesta proporcional, integral y derivativa, sin embargo el carro deslizante diverge.
- El péndulo invertido fue estabilizado utilizando un sistema de control con variables de estado, logrando estabilizar el ángulo, la velocidad angular, la posición del carro y su velocidad lineal en una región cercana al punto de operación.
- Se desarrolló una interfaz que facilita la interacción del sistema, permitiendo al usuario ajustar los parámetros y monitorear el comportamiento del péndulo en tiempo real.
- Las constantes teóricas según el modelo del sistema no fueron óptimas en el modelo físico, por lo que fueron calibradas para obtener respuestas estables.

Aunque se logró implementar el control del péndulo invertido tanto con PID como con variables de estado, se puede explorar e implementar algoritmos de control más avanzados como el control predictivo o el control adaptativo que pueda mejorar la estabilización ante perturbaciones externas. Se puede implementar un observador de variables de estado con el Filtro de Kalman para ponerlo en práctica en tiempo real, facilitando el rendimiento de control al estimar las variables de estado con mayor precisión.

Para una fase futura se recomienda implementar el *Swing up* el cual consiste en estabilizar el sistema a partir de una posición de reposo, balanceándose hasta llegar a la localidad del punto de operación. Implementar este enfoque mejora la capacidad del sistema para estabilizar el péndulo a partir de una variedad de posiciones iniciales. Esta funcionalidad se sugiere implementarla en la interfaz de usuario para conocer la respuesta de la trayectoria en tiempo real. Esta técnica de control puede ser esencial para comprender la optimización de trayectorias que se implementan en el curso de Control de Sistemas 2.

Aunque en la interfaz de usuario se implementaron aspectos esenciales para la calibración y control del péndulo, se debe continuar con el desarrollo para incorporar funciones adicionales que permitan una mejor interacción con el sistema. Se podría implementar un registro histórico, en donde se almacene la respuesta del comportamiento del péndulo y así comparar las gráficas con distintas configuraciones y tener un mejor análisis de las trayectorias del péndulo. En el análisis de la trayectoria del péndulo se podrían obtener los porcentajes de overshoot, el tiempo de subida y tiempo de estabilización, lo cual ayuda a comparar de forma cuantitativa entre distintas respuestas y configuraciones.

Se podría realiza una búsqueda de nuevos componentes para optimizar el diseño mecánico de la plataforma para implementar materiales más ligeros y resistentes que ofrezcan una mejor estabilidad del sistema. Adicionalmente se debe implementar componentes electrónicos que mejoren la potencia del sistema como un motor con más torque y velocidad, o bien mejorar la precisión de lectura en las variables de estado del sistema con sensores más confiables y precisos.

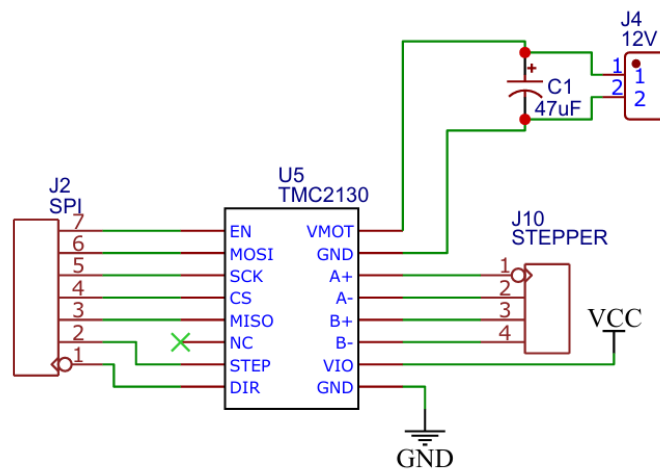
Por otro lado, se recomienda optimizar el procesamiento de datos y actuación del motor para gestionar de mejor manera los recursos del microcontrolador. Esto podría reducir la latencia en la toma de decisiones y mejorar el rendimiento general del sistema, aumentando su velocidad y precisión de control. Se debe utilizar la plataforma correcta del microcontrolador y evitar usar las librerías del driver y los sensores, ya que las librerías podrían tener funciones innecesarias que brinden latencia en la comunicación.

- [1] Oscar Rodríguez, Ronald Pineda, Pedro Cárdenas. Herramientas ejes 3D/Matlab para el control del sistema no lineal aplicado al péndulo invertido sobre carro deslizante [Artículo] Universidad Pedagógica y Tecnológica de Colombia, Revista Colombiana de Tecnologías de Avanzada, vol. 1 n.19, paginas 28-34, 2012.
- [2] Victor Pérez, Elvys Henríquez. Sistemas de control PID analógico para Péndulo Invertido Lineal [Trabajo fin de grado], Instituto Tecnológico de Santo Domingo (INTEC), 2022.
- [3] Francisco Quiles. Robot Autónomo Balanceado, Escuela Politécnica de Valencia Campus de Alcoy, 2021.
- [4] Norman Nise. Control systems engineering [Sistemas de control de ingeniería] [Libro] Pompona, California State Polytechnic University, Wiley, edición 7, 2015.
- [5] Richard Budynas, Keith Nisbett. Diseño de Ingeniería Mecánica de Shigley [Libro] Edificio Punta Santa Fe, Prolongación Paseo de la Reforma 1015, Torre A, Piso 16, Col. Desarrollo Santa Fe, Del. Álvaro Obregón, CP 01376, Ciudad de México, MCGraw-Hill edición 10, 2018.
- [6] Trinamic Motion. TMC2100 Stepper motor driver hoja de datos [TMC2130 Motor stepper driver hoja de datos] [Libro] 2018 [8/09/2024], dirección: <https://www.trinamic.com>.
- [7] OsramGroup. AS5600 12-Bit programmable contactless cagnetic position sensor [AS5600 Sensor de posición magnético sin contacto programable de 12 bits] [Hoja de datos] 2020 [8/09/2024], dirección: <https://ams.com/as5600>.
- [8] EasyEDA: Free easy online PCB design tool [Herramienta gratuita y fácil en línea para diseño de PCB] [En línea] s.f. [8/09/2024], dirección: <https://easyeda.com/es>.
- [9] Nick Smith. PCB trace width calculator [Calculadora de ancho de pista para PCB] [En línea] s.f. [8/09/2024], dirección: <https://nick.desmith.net/Electronics/TraceWidth.html>.
- [10] Ramesh Garg, Inder Bahl, Prakash Bhartia. Microstrip antenna design handbook [Manual de diseño de antenas de microcinta][Libro] Artech House, 2001.
- [11] Mark Spong, Seth Hutchinson, Michael Vidyasagar. Robot modeling and control [Modelado y control de robots] [Libro] Wiley, edición 2, 2020.

- [12] Chi Chen. Theory and design [Teoría y diseño] [Libro] 198 Madison Avenue, New York 10016, Oxford University Press, Inc, edición 3, 1999.
- [13] Hassan Khalil. Nonlinear control [Control no lineal] [Libro] Saffron House, 6-10 Kirby Street, London, England, Pearson Education Limited, edición 1, 2015.
- [14] Joao Hespanha. Linear systems theory [Teoría de sistemas lineales] [Libro] 6 Oxford Street, Woodstock, Oxfordshire, United Kingdom, Princeton University Press, edición 1, 2009.
- [15] Karl Astrom, Richard Murray. Feedback systems [Sistemas de retroalimentación] [Libro] 41 William Street, Princeton, New Jersey, Princeton University Press, edición 2, 2012.
- [16] Gene Franklin, David Powell, Abbas Emami-Naeini. Control de sistemas dinámicos [Libro] México, Pearson, edición 8, 2015.
- [17] John Grayson. Python and tkinter programming [Programación en python y tkinter] [Libro] Shelter Island, NY, Manning Publications, 1999.
- [18] Lindsay Bassett. Introducing JSON [Introduciendo a JSON] [Libro] Sebastopol, California, Estados Unidos, O'Reilly Media, edición 1, 2015.

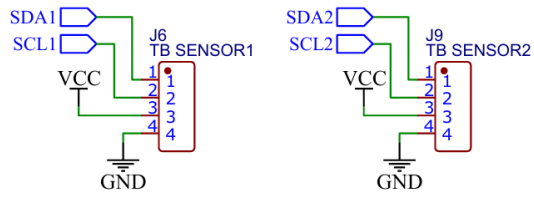
A continuación, se presentan los diagramas que se realizaron para las conexiones de los sensores y drivers a utilizar en la plataforma. Estos diagramas también ayudaron al momento de hacer el diseño preliminar de la placa PCB. Se presentan las conexiones del driver TMC2130 en el Anexo 52, conexiones de sensores AS5600 en el Anexo 53, I2C en el Anexo 54, finales de carrera en el Anexo 55 y alimentación para el microcontrolador en el Anexo 56.

Anexo 52: *Conexiones de driver TMC2130*



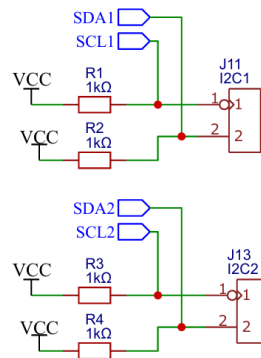
Elaboración propia

Anexo 53: Conexiones de sensores AS5600



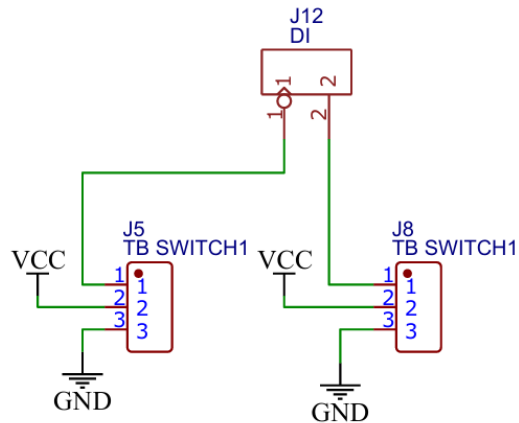
Elaboración propia

Anexo 54: Conexiones comunicación I2C



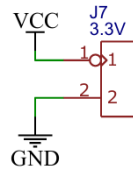
Elaboración propia

Anexo 55: Conexiones de finales de carrera



Elaboración propia

Anexo 56: Conexión a pin 3.3V de microcontrolador



Elaboración propia

Para poder determinar el ancho de pista mínimo a utilizar en el diseño de la placa PCB de la plataforma se trabajó con *PCB Trace Width* que utiliza fórmulas de la norma IPC-2221, tomando en cuenta varios factores, como el amperaje máximo, la temperatura, el grosor de la placa y largo de la pista. En el Anexo [57](#) se muestra el cálculo del ancho de pista para las conexiones de control, mientras que en el Anexo [58](#) se muestra el cálculo del ancho de pista para la alimentación del motor.

Anexo 57: Cálculo de ancho de pista

ANSI PCB TRACE WIDTH CALCULATOR							
Input Data			Results Data				
Field	Value	Units	Trace Data	Internal Traces		External Traces	
				Value	Units	Value	Units
Current (max. 35A)	50	mA	Required Trace Width	2.05	mil	0.79	mil
Temperature Rise (max. 100°C)	1	°C	Cross-section Area	2.75	mil ²	1.06	mil ²
Cu thickness	1	oz/ft ²	Resistance	0.98	Ω Ohms	2.54	Ω Ohms
Ambient Temperature	20	°C	Voltage Drop	0.05	Volts	0.13	Volts
Conductor Length	100	mm	Loss	0	Watts	0.01	Watts
Peak Voltage	3.3	Volts	Required Track Clearance	23.66	mil		

Adaptado de PCB Trace Width (2024) [9]

Anexo 58: Cálculo de ancho de pista para alimentación de motor

ANSI PCB TRACE WIDTH CALCULATOR							
Input Data			Results Data				
Field	Value	Units	Trace Data	Internal Traces		External Traces	
				Value	Units	Value	Units
Current (max. 35A)	2	Amps	Required Trace Width	53.89	mil	20.71	mil
Temperature Rise (max. 100°C)	20	°C	Cross-section Area	72.41	mil ²	27.84	mil ²
Cu thickness	1	oz/ft ²	Resistance	0.01	Ω Ohms	0.03	Ω Ohms
Ambient Temperature	20	°C	Voltage Drop	0.02	Volts	0.05	Volts
Conductor Length	25	mm	Loss	0.04	Watts	0.1	Watts
Peak Voltage	12	Volts	Required Track Clearance	25.4	mil		

Adaptado de PCB Trace Width (2024) [9]