



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA



UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Escuela Técnica Superior de Ingeniería del Diseño

Diseño, implementación y control de un mecanismo de  
péndulo invertido con volante de inercia

Trabajo Fin de Máster

Máster Universitario en Ingeniería Mecatrónica

AUTOR/A: Mendoza Reyes, Lisandro Andrés

Tutor/a: Casanova Calvo, Vicente Fermín

CURSO ACADÉMICO: 2022/2023

## RESUMEN

Este trabajo abarca todos los estudios previos y el proceso a seguir para la construcción de un sistema de péndulo invertido con rueda de reacción. Se explica detalladamente cada fase del proyecto, los componentes a utilizar y las dificultades encontradas.

En la fase de diseño tridimensional, se ha desarrollado un modelo detallado del péndulo invertido con volante de inercia. Utilizando SOLIDWORKS se ha diseñado el péndulo, el volante de inercia y el soporte utilizando

Posteriormente, se ha llevado a cabo la simulación del sistema utilizando MATLAB y Simscape Multibody. Mediante la implementación de un entorno de simulación realista, se han evaluado diferentes estrategias de control y se ha analizado el comportamiento del péndulo invertido con volante de inercia en diversas condiciones de operación. Esto ha permitido optimizar los parámetros del controlador y mejorar el rendimiento general del sistema.

Finalmente, se ha procedido a la implementación prototipo. Se ha construido un prototipo físico basado en el diseño tridimensional desarrollado previamente y se ha implementado el controlador optimizado. Se han llevado a cabo pruebas exhaustivas para validar el funcionamiento del sistema y para evaluar su estabilidad y precisión.



## **ABSTRACT**

This work covers all the previous studies and the process to follow for the construction of an inverted pendulum system with a reaction wheel. Each phase of the project, the components to be used and the difficulties encountered are explained in detail along this work.

In the three-dimensional design phase, a detailed model of the inverted pendulum with reaction wheel has been developed. The pendulum, reaction wheel and the mounting have been designed with the use of SOLIDWORKS.

Subsequently, the simulation of the system has been carried out using MATLAB and Simscape Multibody. Through the implementation of a realistic simulation environment, different control strategies have been evaluated and the behavior of the inverted pendulum with reaction wheel has been analyzed under various operating conditions. This has made it possible to optimize the controller parameters and to improve overall system performance.

Finally, the physical implementation of prototype has been carried out. A physical prototype has been built based on the previously developed three-dimensional design and the optimized controller has been implemented. Extensive tests have been carried out to validate the operation of the system and to assess its stability and accuracy.



## ÍNDICE DE CONTENIDO

I. Introducción .....	8
1.1 Objetivos .....	9
1.1.1 Objetivo general.....	9
1.1.2 Objetivos específicos.....	9
1.2 Justificación .....	10
II. Marco Teórico .....	11
2.1 El péndulo invertido .....	11
2.2 El volante de inercia .....	12
2.3 Péndulo invertido con volante de inercia .....	13
2.3.1 Aplicaciones .....	14
2.4 Motores eléctricos .....	17
2.4.1 Motores de corriente alterna.....	18
2.4.2 Motores de corriente continua .....	19
2.5 Unidad de medición inercial .....	24
2.6 Controlador Proporcional-Integral-Derivativo .....	25
2.7 Modelo matemático del péndulo invertido con rueda de reacción .....	27
2.7.1 Modelo matemático del sistema mecánico .....	27
2.7.2 Modelo matemático del motor de cc.....	31
III. Modelado tridimensional y simulación.....	33
3.1 Diseño del modelo mediante SolidWorks.....	33
3.2 Simulación del modelo mediante Simulink.....	37
3.3 Control del péndulo .....	41
3.4 Resultados de las simulaciones.....	42
3.4.1 Ajuste de parámetros del controlador .....	43
3.4.2 Rechazo de perturbaciones.....	45
3.4.3 Seguimiento de referencia senoidal.....	46
3.4.4 Seguimiento de referencia triangular .....	47
3.4.5 Rechazo de perturbaciones en seguimiento de referencias .....	48
IV. Implementación real .....	50



4.1 Componentes.....	50
4.1.1 Motor eléctrico .....	50
4.1.2 Sensor de inclinación .....	51
4.1.3 Microcontrolador .....	52
4.1.4 Módulo bluetooth .....	53
4.1.5 Fuente de alimentación de corriente continua .....	54
4.1.6 Componentes 3D.....	55
4.2 Montaje.....	55
4.3 Programación del microcontrolador .....	56
4.4 Dificultades en la implementación.....	59
V. Pliego de condiciones.....	61
5.1 definición y alcance del pliego .....	61
5.2 Condiciones generales .....	61
5.3 Condiciones técnicas.....	61
5.3.1 Objeto .....	61
5.3.2 Condiciones de los materiales .....	61
5.3.3 Condiciones de la ejecución.....	62
5.3.4 Pruebas de funcionamiento .....	62
VI. Presupuesto .....	63
VII. Conclusiones .....	64
VIII. Trabajo a futuro .....	65
IX. Bibliografía .....	66
X. Planos.....	67
XI. Código de Arduino .....	71



## ÍNDICE DE FIGURAS

Figura 1. Péndulo invertido simple .....	11
Figura 2. Volante de inercia.....	13
Figura 3. Péndulo invertido con volante de inercia.....	14
Figura 4. Segway.....	15
Figura 5. Sistema de posicionamiento de un satélite .....	16
Figura 6. Robot unicycle con volante de inercia .....	17
Figura 7. Motor de corriente alterna .....	19
Figura 8. Motor de corriente continua.....	20
Figura 9. Motor sin escobillas .....	22
Figura 10. Motor paso a paso .....	24
Figura 11. IMU.....	25
Figura 12. Control PID .....	26
Figura 13. Diagrama del péndulo invertido simple .....	27
Figura 14. Circuito eléctrico de un motor cc .....	32
Figura 15. Base del péndulo.....	33
Figura 16. Volante de inercia.....	34
Figura 17. Soporte del péndulo.....	34
Figura 18. Ensamblaje final del péndulo invertido.....	35
Figura 19. Análisis de masa del péndulo invertido .....	36
Figura 20. Montaje final del péndulo en simulink .....	37
Figura 21. Bucle de control PID .....	41
Figura 22. Selección de referencias.....	42
Figura 23. Respuesta con $k_p$ elevada .....	43
Figura 24. Respuesta con $k_i$ elevada .....	44
Figura 25. Rechazo de perturbaciones en posición vertical.....	45
Figura 26. Seguimiento de referencia senoidal.....	46
Figura 27. Seguimiento de referencia triangular.....	47
Figura 28. Seguimiento de ref. senoidal con perturbaciones.....	48
Figura 29. Seguimiento de ref. triangular con perturbaciones .....	49
Figura 30. Motor Nidec 24H677 .....	50
Figura 31. Sensor MPU6050.....	52
Figura 32. Arduino DUE.....	53
Figura 33. Modulo Bluetooth HM-10.....	54
Figura 34. Componentes 3D.....	55
Figura 35. Montaje final .....	56
Figura 36. Definición de librerías .....	56
Figura 37. Definición de pines.....	57
Figura 38. Configuración PWM .....	57
Figura 39. Configuración inicial .....	57
Figura 40. Controlador PID.....	58



## ÍNDICE DE TABLAS

Tabla 1. Variables de interés .....	28
Tabla 2. Especificaciones técnicas del motor Nidec24H677 .....	51
Tabla 3. Especificaciones técnicas del sensor MPU6050 .....	52
Tabla 4. Especificaciones técnicas del Arduino DUE .....	53
<i>Tabla 5. Especificaciones técnicas del módulo bluetooth HM-10.....</i>	<i>54</i>
Tabla 6. Presupuesto .....	63



## LISTA DE SIGLAS

CA	Corriente Alterna
CAD	Diseño asistido por ordenador
CC	Corriente Continua
IMU	Unidad de Medición Inercial
PID	Proporcional-Integral-Derivativo



## I. INTRODUCCIÓN

Un péndulo invertido consiste en un barra acoplada a un soporte, en estos sistemas de control, el objetivo es estabilizar la barra en posición vertical controlando el movimiento de la barra. Una forma de lograr esto es mediante el uso de un volante de inercia o rueda de reacción, que es una masa giratoria que permite generar un par de torsión para contrarrestar el movimiento del péndulo.

Un control de estabilidad fiable es un requisito fundamental para lograr el rendimiento deseado del sistema. Un sistema de control estable asegura que la respuesta del sistema se establezca dentro de límites aceptables, proporcionando acciones de control precisas y oportunas. Además, este tipo de control mejora la robustez de los sistemas, esto se refiere a la capacidad de un sistema de mantener un funcionamiento estable incluso cuando se enfrenta a condiciones inciertas o variables, como cambios en la dinámica o perturbaciones externas.

El presente trabajo tiene como finalidad mostrar el proceso de diseño e implementación de un prototipo de péndulo invertido con volante de inercia. A lo largo del desarrollo de este prototipo se pondrán en prueba los conocimientos adquiridos sobre electrónica, mecánica y control.

Este trabajo está compuesto por las secciones que se detallan a continuación:

Planteamiento del problema: se identifican los objetivos del proyecto, así como la justificación de este.

Marco teórico: se exponen los fundamentos técnicos y teóricos necesarios para el desarrollo del proyecto.

Modelado tridimensional y simulación: se detalla todo el proceso de diseño mecánico de las piezas en SolidWorks, también se especifican las simulaciones de funcionamiento realizadas en Simscape Multibody.

Implementación real: se detalla todo el proceso que conlleva la implementación física del prototipo, de la misma forma se especifican los componentes electrónicos y software a utilizar.

Conclusiones: se presentan las conclusiones obtenidas del desarrollo del proyecto.



Las demás secciones presentan las dificultades encontradas, el presupuesto y recomendaciones propuestas para el desarrollo de este prototipo.

## 1.1 OBJETIVOS

### 1.1.1 OBJETIVO GENERAL

El objetivo general de este Trabajo Final de Máster es el modelado, diseño del control, simulación e implementación real de un péndulo invertido con volante de inercia., se busca estabilizar el péndulo en su posición vertical con la ayuda de un volante de inercia conectado a un motor.

### 1.1.2 OBJETIVOS ESPECÍFICOS

1. Realizar un estudio previo sobre el comportamiento, aplicaciones y componentes utilizados para adquirir un mejor conocimiento del funcionamiento del sistema.
2. Formular un modelo matemático del sistema.
3. Diseñar los componentes mecánicos del péndulo utilizando un programa de diseño computarizado.
4. Crear un modelo del prototipo en Simulink que permita el estudio del comportamiento del péndulo en un entorno simulado.
5. Efectuar las simulaciones de funcionamiento mediante el uso de Simulink para analizar el comportamiento del sistema.
6. Construir el prototipo final del péndulo mediante la impresión 3D y ensamblaje de los componentes mecánicos diseñados.
7. Realizar la programación del microcontrolador que se encarga de controlar el comportamiento del sistema y verificar el correcto funcionamiento del prototipo.



## 1.2 JUSTIFICACIÓN

De acuerdo con la normativa que regula las enseñanzas universitarias oficiales, los estudios conducentes a la obtención del Título Oficial de Máster, se deben concluir con un trabajo fin de máster. Por lo que realizar un trabajo fin de máster es un requerimiento de carácter obligatorio para la obtención del Título de “Máster en Ingeniería Mecatrónica”.

El diseño y desarrollo de un péndulo invertido con volante de inercia ofrece una oportunidad para investigar y explorar los principios fundamentales de la física y el control. Este proyecto permite adquirir conocimientos prácticos sobre la estabilización y control de sistemas no lineales, lo cual es relevante en diversas áreas de la ingeniería y la robótica.

Además, la construcción de un péndulo invertido implica la aplicación de habilidades técnicas en áreas como la mecánica, la electrónica y el control. Este proyecto brinda la oportunidad de desarrollar competencias en diseño mecánico, montaje de sistemas electrónicos, programación de controladores y análisis de datos, lo cual es valioso para una carrera en ingeniería mecatrónica.

## II. MARCO TEÓRICO

### 2.1 EL PÉNDULO INVERTIDO

Un péndulo invertido es un sistema de control dinámico en el que un objeto con forma de barra se equilibra verticalmente sobre un punto de pivote, sin soporte externo. El problema del péndulo invertido es un tema fundamental en teoría de control y robótica, y sirve como punto de referencia para diseñar y probar algoritmos de control.

La configuración básica de un péndulo invertido, también conocida como el péndulo invertido simple, consiste en una barra de péndulo con una masa en la parte superior y un pivote en la parte inferior. La masa del péndulo puede moverse libremente en el plano horizontal, y el objetivo es controlar el péndulo de tal manera que permanezca en posición vertical sin caerse.

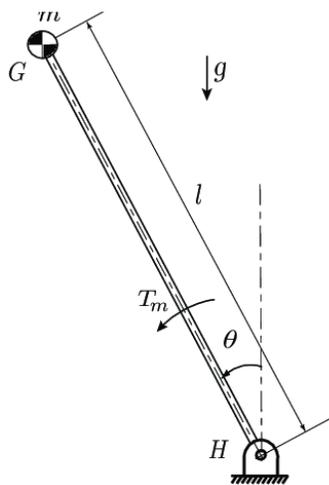


Figura 1. Péndulo invertido simple

El péndulo invertido simple es inherentemente inestable porque el péndulo se encuentra en una posición de equilibrio inestable cuando está perfectamente vertical. Incluso una pequeña perturbación puede hacer que la barra se caiga. Por lo tanto, controlar este tipo de sistema requiere ajustes continuos de la posición de la barra para mantener el péndulo en la posición deseada.



Existen varios enfoques para resolver este problema de control, entre los principales métodos de control están, el control proporcional-integral-derivativo (PID) y el control basado en modelos físicos. El objetivo es diseñar un algoritmo de control que pueda detectar el estado del péndulo y aplicar las entradas de control adecuadas para mantenerlo en equilibrio.

El problema del péndulo invertido proporciona un contexto interesante y práctico para estudiar los conceptos fundamentales de la teoría de control, incluidos el modelado de sistemas, la estabilidad, la controlabilidad y la observabilidad. También sirve como plataforma para explorar técnicas de control avanzadas y sus aplicaciones en sistemas del mundo real.

## 2.2 EL VOLANTE DE INERCIA

La volante de inercia, también conocido como rueda de reacción, consiste en una masa montada sobre un eje motorizado. La masa almacena energía cinética al girar, de acuerdo con el principio de conservación del momento angular, si la rueda de reacción cambia su velocidad o dirección de rotación, debe ocurrir un cambio igual y opuesto en el objeto a cuál está acoplada la rueda. (Díaz, 2017)

Es un dispositivo comúnmente utilizado en naves espaciales y satélites para el control de actitud en el espacio. Si la nave espacial quiere girar en una dirección específica, puede usar una rueda de reacción para crear una rotación opuesta. Al hacer girar la rueda de reacción en la dirección opuesta, la nave espacial experimenta un par o fuerza de torsión. Esta fuerza hace que la nave espacial gire en la dirección deseada.

Cada rueda puede producir torque solo a lo largo de su eje de rotación. Entonces, al montar tres ruedas de reacción a lo largo de ejes perpendiculares entre sí, se obtiene la magnitud y dirección deseada del par, permitiendo el control preciso de la orientación de la nave. Sin embargo, existe un límite en el par que pueden producir, porque las ruedas no se pueden acelerar y desacelerar sin límites.



*Figura 2. Volante de inercia*

Una de las principales ventajas de las ruedas de reacción es su alta precisión y exactitud. Pueden hacer ajustes muy pequeños en la actitud de la nave y pueden hacerlo sin causar vibraciones ni perturbaciones. Esto es particularmente importante para las naves espaciales científicas que necesitan mantener una orientación precisa en todo momento.

En este caso, se utilizará una rueda de reacción para controlar el ángulo de inclinación, con respecto del eje  $x$ , del modelo en todo momento. Para esto la rueda deberá ser capaz de regular la velocidad para poder mantener la orientación deseada en caso de que el sistema se vea afectado por perturbaciones.

### **2.3 PÉNDULO INVERTIDO CON VOLANTE DE INERCIA**

El péndulo invertido con volante de inercia es un sistema que consiste en un péndulo montado verticalmente sobre una base fija, con una rueda unida a la parte superior de la barra. La rueda es impulsada por un motor eléctrico y gira a alta velocidad para generar el par necesario para mantener el péndulo en posición vertical.

Al cambiar la velocidad de rotación de la rueda, se genera un par que afecta el movimiento del péndulo, lo que ayuda a mantenerlo en posición vertical. La rueda de reacción opera según el principio de conservación del momento angular donde los cambios en el momento angular de la rueda dan como resultado una reacción igual y opuesta en el péndulo.

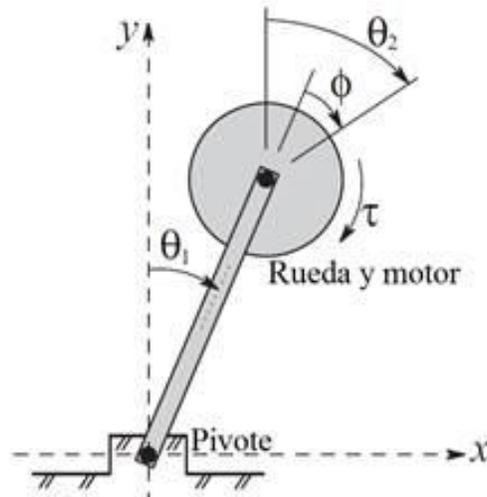


Figura 3. Péndulo invertido con volante de inercia

El péndulo invertido con sistema de rueda de reacción funciona según el principio de la teoría de control. La posición y la velocidad angular del péndulo se miden mediante sensores, y los datos se utilizan para calcular el par necesario para mantener el péndulo en posición vertical. La rueda de reacción genera un par contrarrestante que estabiliza el péndulo, evitando así su caída.

Para controlar este tipo de sistema se utiliza un algoritmo de control de retroalimentación en bucle cerrado. Este sistema mide continuamente la inclinación del péndulo y calcula la acción de control necesaria para mantener el péndulo en la posición deseada.

### 2.3.1 APLICACIONES

El péndulo invertido y los volantes de inercia han encontrado diversidad de aplicaciones en una variedad de campos. Las ruedas de reacción son utilizadas en aplicaciones espaciales, mayoritariamente en el control de actitud de satélites y naves espaciales. Mientras el péndulo invertido es utilizado en medios de transporte como el Segway.

#### **Segway**

El Segway utiliza el principio del péndulo invertido para lograr su función de autoequilibrio. Al emplear una serie de sensores, incluidos acelerómetros y giroscopios, el Segway mide continuamente su inclinación y aceleración. Estos datos luego son procesados por un sistema de

control que se encarga de analizar los datos de los sensores para determinar la posición actual y cualquier cambio en el equilibrio del Segway.

Según este análisis, los motores eléctricos conectados a las ruedas u otros mecanismos de equilibrio generan un par para ajustar la posición del Segway y devolverlo a su estado vertical. Este proceso de equilibrio dinámico ocurre en tiempo real, lo que permite que el Segway responda a los movimientos del conductor y mantenga la estabilidad mientras está en movimiento.



*Figura 4. Segway*

### **Aplicaciones Espaciales**

Estos dispositivos permiten el control de la actitud de un satélite sin necesidad de propulsores de posicionamiento ni ningún otro aplicador externo de par. El uso de ruedas de reacción presenta grandes ventajas, se reduce significativamente la carga útil necesaria para el combustible de los propulsores, además, dado que la rueda de reacción funciona con energía eléctrica, puede funcionar siempre que el satélite pueda suministrar energía.

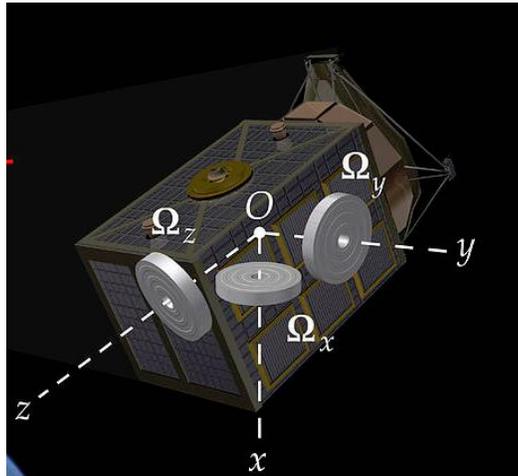


Figura 5. Sistema de posicionamiento de un satélite

Además, las ruedas de reacción son extremadamente precisas y pueden permitir un reposicionamiento preciso. A medida que cambia la actitud del satélite, la rueda de reacción se dirige automáticamente para inducir una contra rotación de la nave espacial mediante la conservación del momento angular.

### **Robots**

Las ruedas de reacción también son utilizadas en aplicaciones de robótica, su objetivo sigue siendo el mismo, controlar el movimiento y estabilidad del robot. Al ajustar la velocidad de rotación de las ruedas de reacción, el robot puede cambiar su orientación o mantener una orientación deseada en relación con su entorno. Esto es fundamental para tareas que requieren un control preciso de la posición y la orientación del robot, como mover objetos, sortear obstáculos o mantener la estabilidad en superficies irregulares.



*Figura 6. Robot unicycle con volante de inercia*

Los volantes de inercia se pueden utilizar para mantener la estabilidad y el equilibrio en los robots, especialmente en entornos dinámicos o inestables. Al controlar activamente la rotación de las ruedas de reacción, un robot puede ajustar su momento de inercia y generar pares para contrarrestar perturbaciones externas, como vibraciones, ráfagas de viento o impactos, y mantener la estabilidad y el equilibrio durante la operación.

También se utilizan para lograr una orientación precisa de sensores, cámaras u otras cargas útiles en un robot. Al ajustar la rotación de las ruedas de reacción, un robot puede controlar la orientación y la dirección de apuntado de sus sensores o cargas útiles, lo que le permite rastrear o escanear con precisión un objetivo, como en aplicaciones de detección remota, imágenes o inspección.

## **2.4 MOTORES ELÉCTRICOS**

Para elegir el motor adecuado que permita cumplir las especificaciones de funcionamiento del sistema, se deben de analizar los diferentes tipos de motores y sus características.

Los motores eléctricos convierten energía eléctrica en energía mecánica y se clasifican en dos categorías: Corriente Alterna (CA) y Corriente Continua (CC). Dentro de estas categorías se encuentran numerosos tipos de motores, cada uno ofrece características únicas que se adaptan bien a aplicaciones específicas.



### 2.4.1 MOTORES DE CORRIENTE ALTERNA

Un motor de CA es un tipo de motor eléctrico que opera utilizando una fuente de CA. Este tipo de motor consta de dos partes esenciales: un estator y un rotor. La parte estacionaria del motor, el estator, tiene bobinas y se alimenta con corriente eléctrica para producir un campo magnético giratorio. La parte giratoria del motor, el rotor, produce un segundo campo magnético giratorio, lo que permite su giro. La forma en que se produce este segundo campo magnético depende del tipo de motor eléctrico que se utilice. Existen dos tipos principales de motores de CA: motores asíncronos y motores síncronos.

Los motores asíncronos o de inducción, funcionan según el principio de inducción electromagnética, mediante el cual la corriente que pasa a través del devanado del estator produce un campo magnético giratorio, induciendo una corriente en el rotor, lo que hace que este gire. En este tipo de motor el campo magnético del rotor siempre gira a una menor velocidad al campo del estator.

Un motor síncrono funciona a una velocidad constante determinada por la frecuencia del voltaje aplicado y el número de polos del motor. En este tipo de motor, el estator contiene un conjunto de electroimanes que producen un campo magnético giratorio, mientras que el rotor contiene un conjunto de imanes permanentes. El campo magnético del rotor es atraído por el campo magnético giratorio del estator, lo que hace que el rotor gire a la misma velocidad que el campo magnético giratorio.

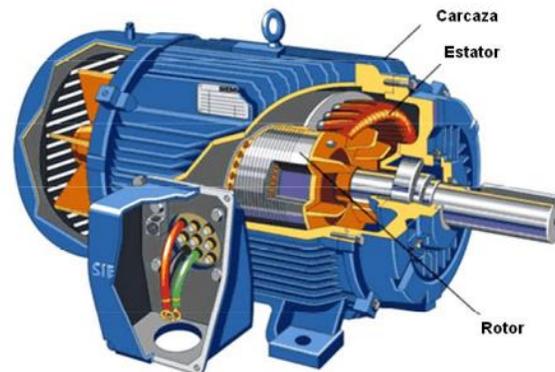


Figura 7. Motor de corriente alterna

Una de las principales desventajas de los motores de CA es el control, estos motores tienen problemas debido a la incapacidad de trabajar a velocidades bajas debido a consideraciones térmicas. El control variable de velocidad y el control preciso de posición de los estos motores resulta más difícil que el control de motores de CC. En cuanto al costo, para aplicaciones de baja potencia, los motores AC son considerablemente más caros que su contraparte. Además, los motores de CA pueden tener un factor de potencia menor en comparación con su contraparte, lo que puede resultar en un mayor consumo de energía y una eficiencia reducida.

Teniendo en consideración el costo y la capacidad de los motores para cumplir las especificaciones del proyecto, se ha descartado la posibilidad de utilizar motores de CA para el proyecto.

#### 2.4.2 MOTORES DE CORRIENTE CONTINUA

A diferencia de los motores de CA los motores de CC se alimentan con una fuente de CC. El diseño básico de un motor de CC consta de un estator y un rotor. El estator contiene un conjunto de imanes permanentes que crean un campo magnético, mientras que el rotor tiene una serie de bobinas conductoras que transportan corriente eléctrica. Cuando se aplica una corriente eléctrica al rotor, este interactúa con el campo magnético creado por el estator, lo que hace que el rotor gire.

Existen dos tipos principales de motores de CC: con escobillas y sin escobillas. Los motores con escobillas utilizan un conjunto de escobillas hechas de carbón para suministrar corriente eléctrica al rotor. Los motores sin escobillas, por otro lado, usan controladores electrónicos para suministrar la corriente al rotor.

Para decidir el tipo de motor que más se acomoda a las especificaciones del proyecto, se debe de primero analizar las características, ventajas y desventajas de estos.

### Motor de CC con escobillas

Los motores de CC con escobillas son motores simples con una construcción sencilla. Alrededor del exterior de la carcasa del motor se encuentran los imanes del estator, estos imanes son los que se encargan de generar el campo magnético estático. En el centro del motor, conectado al eje del motor, se encuentran los devanados de alambre conectados a placas de metal llamadas armaduras. (Mayuresh, 2020)

En el lado opuesto de los devanados del eje del motor está el conmutador, también llamado colector, un par de placas de metal unidas a los devanados. Finalmente, las escobillas, de donde toma su nombre este tipo de motor, se encuentran comúnmente en el lado del motor opuesto al eje del motor.

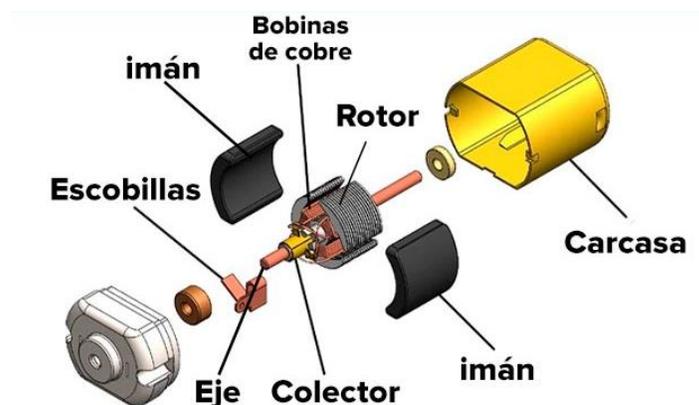


Figura 8. Motor de corriente continua



Las escobillas hacen contacto físico con el conmutador, cuando se aplica un voltaje de CC a las escobillas, ese voltaje se transfiere al conmutador que, a su vez, alimenta los devanados. Esta entrada eléctrica genera un campo magnético alrededor de la armadura.

El polo negativo del campo generado en la armadura se repele del imán con polo negativo, y se atrae al imán positivo. Mientras el polo positivo del campo se aleja del imán positivo. Al alternar constantemente la polaridad del campo magnético alrededor de la armadura, se hace que el eje del motor gire continuamente.

Los motores con escobillas son relativamente simples y económicos, el control de velocidad es relativamente simple, ya que, a mayor voltaje o ciclo de trabajo, el motor girará más rápido. Además, estos motores producen un alto par a bajas velocidades. Esto es importante porque este alto par de arranque permite que el motor alcance la velocidad deseada rápidamente, incluso si hay una carga en el motor.

Una de las principales desventajas es su vida útil limitada debido al desgaste de las escobillas y el conmutador. Otras desventajas de los motores con escobillas incluyen la disipación de calor inadecuada causada por las limitaciones del rotor, la alta inercia del rotor, el bajo rango de velocidad debido a las limitaciones impuestas por las escobillas y la interferencia electromagnética generada por el arco de las escobillas.

Los motores con escobillas se utilizan en una gran variedad de aplicaciones, desde dispositivos electrónicos de uso diario hasta aplicaciones industriales. Estos motores son utilizados en aplicaciones que necesitan altos pares de arranque o cambios bruscos en los requerimientos de par. Su utilización es común en equipos de uso doméstico. También se utilizan en aplicaciones de automoción, incluyendo limpiaparabrisas, posicionadores de asientos eléctricos y en ventanas eléctricas.

## Motor de CC sin escobillas

Como su nombre lo indica, los motores de CC sin escobillas, o brushless DC (BLDC), no usan escobillas. En este tipo de motor, las bobinas no están ubicadas en el rotor. En cambio, el rotor es un imán permanente; las bobinas no giran, sino que están fijas en el estator. Debido a que las bobinas no se mueven, no se necesitan escobillas ni conmutador.

En el motor con escobillas, la rotación se logra controlando los campos magnéticos generados por las bobinas en el rotor, mientras que el campo magnético generado por los imanes estacionarios permanece fijo. Para cambiar la velocidad de rotación, se cambia el voltaje aplicado a las bobinas. En un motor brushless, es el imán permanente el que gira; la rotación se logra cambiando la dirección de los campos magnéticos generados por las bobinas estacionarias circundantes. (Mayuresh, 2020)



Figura 9. Motor sin escobillas

Una gran ventaja de este tipo de motores es la eficiencia, ya que estos motores pueden funcionar continuamente al máximo par. Los motores con escobillas, por el contrario, alcanzan el par máximo solo en ciertos puntos de la rotación. La segunda gran ventaja, relacionada con la primera, es la capacidad de control. Los motores BLDC se pueden controlar mediante mecanismos de retroalimentación para proporcionar con precisión el par y la velocidad de rotación deseados.



Los motores BLDC también ofrecen alta durabilidad y baja generación de ruido eléctrico, gracias a la ausencia de escobillas. Con motores con escobillas, las escobillas y el conmutador se desgastan como resultado del contacto en movimiento continuo y también producen chispas y ruido en el lugar donde se hace contacto.

Entre las principales desventajas del motor con escobillas, se encuentran, el costo de un motor sin escobillas es mayor en comparación con un motor con escobillas de las mismas especificaciones. El cableado y la operación en un motor BLDC es más complejo debido a la participación de controles electrónicos.

Estos tipos de motores han encontrado aplicaciones en todo tipo de dispositivos, desde equipos médicos, robots y drones hasta automóviles y herramientas eléctricos. Se utilizan esencialmente en aplicaciones que requieren alta eficiencia y longevidad.

### **Motor paso a paso**

Un motor paso a paso de CC es un tipo de motor que convierte pulsos eléctricos en movimiento de rotación. Es un tipo de motor sin escobillas que divide una rotación completa en varios pasos de igual tamaño. El motor funciona según el principio de atracción y repulsión electromagnética, con el rotor alineándose con el campo magnético generado por el estator. A diferencia de un motor de CC normal, que gira continuamente cuando se aplica energía, un motor paso a paso gira en pasos pequeños y precisos. (TME, 2020)

Los motores paso a paso de CC se usan comúnmente en una variedad de aplicaciones que requieren un posicionamiento preciso, como en robótica, máquinas CNC e impresoras 3D. Ofrecen una serie de ventajas, incluido un alto par a bajas velocidades, un control de posicionamiento preciso y un bajo consumo de energía.

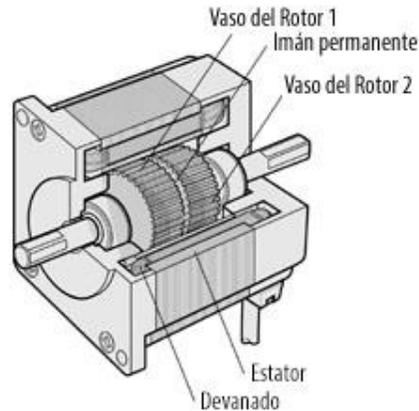


Figura 10. Motor paso a paso

## 2.5 UNIDAD DE MEDICIÓN INERCIAL

Para controlar el movimiento del modelo es necesario medir la inclinación actual del modelo en todo momento, para esto se utiliza una unidad de medición inercial (IMU). Una IMU es un dispositivo electrónico utilizado para medir velocidades, aceleraciones y fuerzas acelerómetro que actúan sobre un objeto. Las IMU están comúnmente formados por acelerómetros y giroscopios, y en algunos casos magnetómetros. (SBG Systems, 2023)

Un acelerómetro es un sensor electrónico que mide las fuerzas de aceleración que actúan sobre un objeto. El dispositivo consta de una masa unida a un resorte que está conectado a un marco de referencia. Cuando el dispositivo experimenta un cambio de velocidad, la masa se mueve en relación con el marco de referencia y el resorte se estira o comprime en respuesta. La cantidad de desplazamiento de la masa es proporcional a la aceleración experimentada por el dispositivo. El acelerómetro mide este desplazamiento y lo convierte en una señal eléctrica que puede usarse para determinar la aceleración del dispositivo. (Omega, 2023)

Por otra parte, un giroscopio es un dispositivo que mide los cambios de rotación o de orientación de un objeto al detectar las fuerzas que actúan sobre una rueda giratoria en el interior. Proporciona información crucial sobre los movimientos del objeto, que luego es utilizada por

diferentes sistemas y aplicaciones para la estabilidad, detección de orientación y control. (5Hertz, 2023)

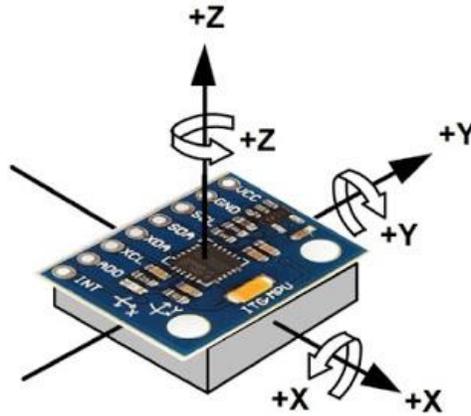


Figura 11. IMU

Combinando las mediciones de los sensores integrados en la IMU, es posible obtener una medida precisa del ángulo de inclinación de un objeto.

## 2.6 CONTROLADOR PROPORCIONAL-INTEGRAL-DERIVATIVO

El control Proporcional-Integral-Derivativo (PID) es el algoritmo de control más común utilizado en la industria y ha sido universalmente aceptado en el control industrial. La popularidad de los controladores PID se puede atribuir en parte a su sólido rendimiento en una amplia gama de condiciones de funcionamiento y en parte a su simplicidad funcional. Como sugiere el nombre, el algoritmo PID consta de tres coeficientes básicos; proporcional, integral y derivativo que se varían para obtener una respuesta óptima.

El sistema en bucle cerrado es un sistema de control típico, la variable de proceso es el parámetro del sistema que debe controlarse. Se utiliza un sensor para medir la variable del proceso y proporcionar retroalimentación al sistema de control. El punto de ajuste es el valor deseado para la variable de proceso. En cualquier momento dado, la diferencia entre la variable de proceso y el punto de referencia es utilizada por el algoritmo del sistema de control para determinar la salida del actuador deseada para accionar el sistema. (Wang, 2020)

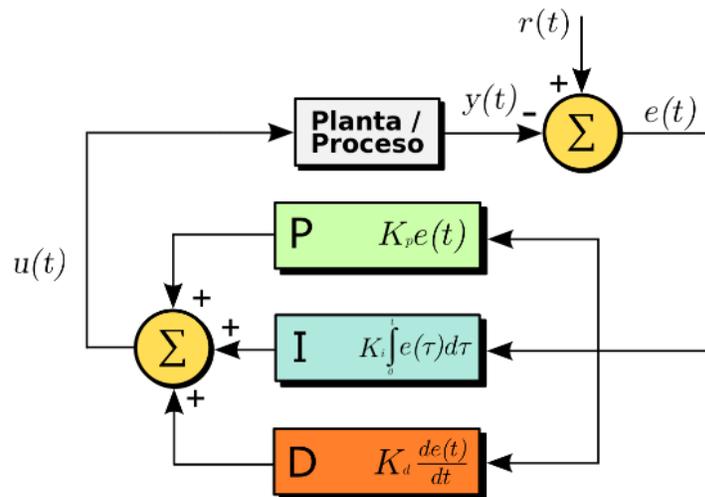


Figura 12. Control PID

El componente proporcional depende únicamente de la diferencia entre la referencia y la variable de proceso, también conocida como error. La ganancia proporcional  $k_p$  determina la relación entre la señales de control y de error. En general, aumentar la ganancia proporcional aumentará la velocidad de respuesta del sistema de control. Sin embargo, si la ganancia proporcional es demasiado grande, la variable del proceso comenzará a oscilar, lo que conlleva a inestabilidad en el sistema.

El componente integral suma el término de error a lo largo del tiempo. La respuesta integral aumentará continuamente con el tiempo a menos que el error sea cero, por lo que el objetivo es llevar el error de estado estacionario a cero. El error de estado estacionario es la diferencia final entre la variable de proceso y el punto de referencia.

El componente derivativo hace que la salida disminuya si la variable del proceso aumenta rápidamente. La respuesta derivada es proporcional a la tasa de cambio de la variable del proceso. El aumento de la constante derivativa  $k_d$  hace que el sistema de control reaccione con más fuerza a los cambios en el término de error y aumentará la velocidad de respuesta general del sistema de control.

## 2.7 MODELO MATEMÁTICO DEL PÉNDULO INVERTIDO CON RUEDA DE REACCIÓN

El modelo matemático expuesto en esta sección tiene como finalidad brindar una fundamentación teórica del comportamiento dinámico del mecanismo. El análisis matemático expuesto se basa en estudios ya desarrollados sobre el comportamiento dinámico del péndulo invertido. (Belascuen, 2018) (Thasleem, 2015)

### 2.7.1 MODELO MATEMÁTICO DEL SISTEMA MECÁNICO

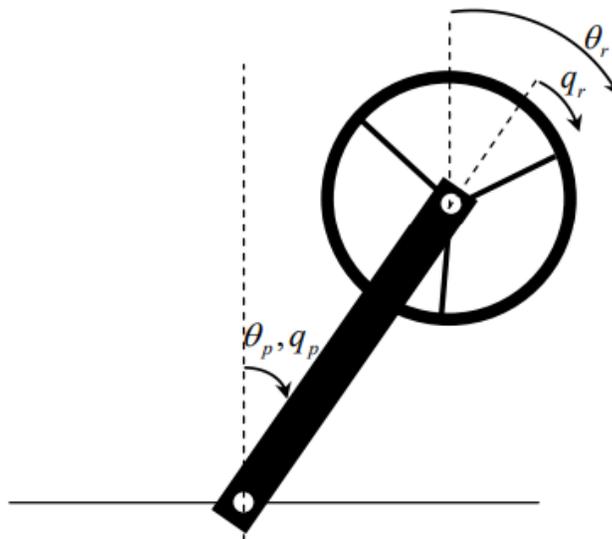


Figura 13. Diagrama del péndulo invertido simple

Para realizar el modelo matemático es necesario definir un diagrama del péndulo con rueda de reacción donde se muestren las coordenada generalizadas necesarias para la formulación matemática. El ángulo,  $\theta_p$ , es el grado de inclinación del péndulo medido desde la vertical, mientras  $\theta_r$  es el ángulo de la rueda medido de la misma forma.

Con ayuda de un sensor de inclinación se puede obtener el ángulo del péndulo  $q_p$ , de la misma forma, el ángulo de la rueda  $q_r$  se puede obtener por medio de un tacómetro o se puede estimar.



Debido a que la rueda está montada sobre el péndulo, el ángulo  $q_r$  es medido con respecto al marco del péndulo, por lo que se debe formular la siguiente relación para representar ambas medidas con respecto al marco principal.

$$\theta_p = q_p \quad (2.1)$$

$$\theta_r = q_p + q_r \quad (2.2)$$

Seguidamente, se deben de introducir las demás variables de interés.

**Tabla 1. Variables de interés**

$m_p$	Masa del péndulo	Kg
$m_r$	Masa de la rueda de reacción	Kg
$J_p$	Momento de inercia del péndulo respecto a su centro de masa	$\text{Kg} \cdot \text{m}^2$
$J_r$	Momento de inercia de la rueda con respecto a su centro de masa	$\text{Kg} \cdot \text{m}^2$
$l_p$	Distancia desde la base hasta el centro de masa del péndulo	m
$l_r$	Distancia desde la base hasta el eje de rotación de la rueda	m
$m$	Masa combinada del mecanismo	kg
$l$	Distancia $l_r + l_p$	m
$\tau_r, \tau_p$	Pares generalizados	$\text{N} \cdot \text{m}$
$\tau_m$	Par generado por el motor	$\text{N} \cdot \text{m}$
$k_r, k_p$	Coefficientes de fricción dinámica	

Para resolver el problema cinemático se recurre al método del Lagrangiano, la función lagrangiana, denotada por  $L$ , se usa comúnmente en el cálculo de variaciones y la mecánica



lagrangiana para describir la dinámica de un sistema físico. Se define como la diferencia entre la energía cinética (T) y la energía potencial (V) del sistema.

La función Lagrangiana proporciona una representación matemática concisa del comportamiento del sistema y es esencial para formular las ecuaciones de movimiento.

$$L = T - V \quad (2.3)$$

El primer paso es calcular la energía cinética del sistema, para esto se debe calcular la energía cinética del péndulo  $T_1$  y de la rueda  $T_2$ .

$$T_1 = \frac{1}{2}(m_p l_p^2 + J_p) \dot{q}_p^2 \quad (2.4)$$

$$T_2 = \frac{1}{2} m_r l_r^2 \dot{q}_p^2 + \frac{1}{2} J_r (\dot{q}_p + \dot{q}_r)^2 \quad (2.5)$$

La energía cinética total del sistema, T, está dada por:

$$T = T_1 + T_2 \quad (2.6)$$

Sustituyendo las coordenadas generalizadas se obtiene:

$$T = \frac{1}{2} (m_r l_r^2 + m_p l_p^2 + J_p) \dot{\theta}_p^2 + \frac{1}{2} J_r \dot{\theta}_r^2 \quad (2.7)$$

La energía potencial no depende de la posición del rotor ya que la masa del rotor es igualmente distribuida independientemente del ángulo del rotor. Por lo tanto, la energía potencial depende únicamente de la interacción de la masa del péndulo con la gravedad.

$$V = mgl \cos \theta \quad (2.8)$$

Una vez calculados los términos de T y V se puede calcular la función Lagrangiana dada por:



$$L = T - V = \frac{1}{2}J \dot{\theta}_p^2 + \frac{1}{2}J_r \dot{\theta}_r^2 - mgl \cos \theta \quad (2.9)$$

Donde  $I$  esta dado por:

$$J = (m_r l_r^2 + m_p l_p^2 + J_p) \quad (2.10)$$

Se calculan las derivadas parciales y se obtiene:

$$\begin{aligned} \frac{\partial L}{\partial \dot{\theta}_p} &= J \dot{\theta}_p & \frac{\partial L}{\partial \theta_p} &= mgl \cos \theta_p \\ \frac{\partial L}{\partial \dot{\theta}_r} &= J_r \dot{\theta}_r & \frac{\partial L}{\partial \theta_r} &= 0 \end{aligned}$$

En este caso, el par producido por el motor resultada en un par positivo,  $\tau$ , que actúa en la rueda y un par  $-\tau$ , que actúa en el péndulo. Además, existen fuerzas de fricción que actúan en el eje de rotación. En este caso en concreto actúan fuerzas de fricción estáticas y dinámicas. Para simplificar la resolución del problema, solo se toma en cuenta la fricción dinámica. La fricción dinámica que actúa en el eje es proporcional a la velocidad angular del eje y en la dirección opuesta a la velocidad angular. Esto produce los pares generalizados que actúan en la dirección  $\theta_r$  y  $\theta_p$  que se detallan a continuación.

$$\tau_r = \tau_m - k_r \dot{q}_r \quad (2.11)$$

$$\tau_p = -\tau_m + k_r \dot{q}_r - k_p \dot{q}_p \quad (2.12)$$

Sustituyendo las coordenadas generalizadas:

$$\tau_r = -\tau_m + k_r \dot{\theta}_p - k_r \dot{\theta}_r \quad (2.13)$$

$$\tau_p = -\tau_m + k_r \dot{\theta}_r - (k_r + k_p) \dot{\theta}_p \quad (2.14)$$

Sustituyendo las derivadas Lagrangianas y las fuerzas generalizadas se obtienen los siguientes sistemas de ecuaciones:



$$J\ddot{\theta}_p - mgl\sin(\theta_p) = -\tau_m + k_r\dot{\theta}_r - (k_r + k_p)\dot{\theta}_p \quad (2.15)$$

$$J_r\ddot{\theta}_r = \tau_m + k_r\dot{\theta}_p - k_r\dot{\theta}_r \quad (2.16)$$

Reordenando las ecuaciones se obtienen:

$$\ddot{\theta}_p = \frac{mgl}{J}\sin(\theta_p) - \frac{(k_p+k_r)}{J}\dot{\theta}_p + \frac{k_r}{J}\dot{\theta}_r - \frac{\tau_m}{J} \quad (2.17)$$

$$\ddot{\theta}_r = \frac{k_r}{I_r}\dot{\theta}_p - \frac{k_r}{I_r}\dot{\theta}_r + \frac{\tau_m}{I_r} \quad (2.18)$$

Las ecuaciones 2.17 y 2.18 caracterizan el comportamiento de un sistema de péndulo invertido con rueda de reacción, conociendo el par motor. Observando las ecuaciones se puede llegar a la conclusión que se está tratando con un sistema de ecuaciones no lineales, ya que, una de las ecuaciones contiene el termino de seno,  $\sin(\theta_p)$ , Este sistema se puede linealizar sustituyendo el  $\sin(\theta_p)$  con  $\theta_p$ , para valores muy pequeños de  $\theta_p$ . Las ecuaciones linealizadas se muestran a continuación:

$$\ddot{\theta}_p = \frac{mgl}{J}\theta_p - \frac{(k_p+k_r)}{J}\dot{\theta}_p + \frac{k_r}{J}\dot{\theta}_r - \frac{\tau_m}{J} \quad (2.19)$$

$$\ddot{\theta}_r = \frac{k_r}{I_r}\dot{\theta}_p - \frac{k_r}{I_r}\dot{\theta}_r + \frac{\tau_m}{I_r} \quad (2.20)$$

### 2.7.2 MODELO MATEMÁTICO DEL MOTOR DE CC

Para formular el modelo matemático del motor de CC se parte del circuito eléctrico que modela el funcionamiento del motor. En este modelo de circuito, la fuente de voltaje (V) está conectada en serie con la resistencia de la armadura (R), la inductancia de la armadura (L) y el propio motor. La fuerza contraelectromotriz (E) está incluida en el circuito del motor ya que se opone al voltaje aplicado y afecta la corriente que fluye a través del motor.

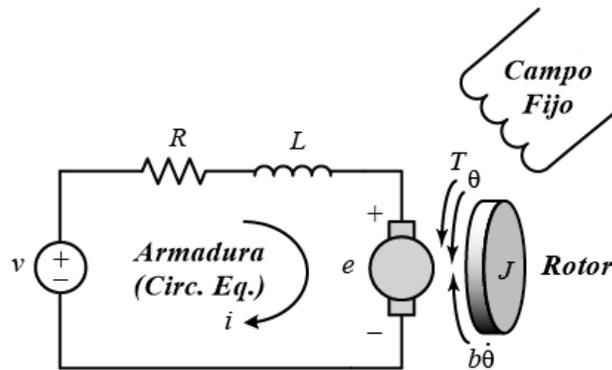


Figura 14. Circuito eléctrico de un motor cc

En un motor de CC, la corriente de armadura ( $I_a$ ) que fluye a través del devanado del motor crea un campo magnético que interactúa con el campo magnético estático producido por el estator. Esta interacción da como resultado un par, que hace que el motor gire. La constante de par ( $K_t$ ) relaciona el par y la corriente del motor. Es esencialmente un factor de conversión que convierte la corriente eléctrica en par mecánico. Esta relación se caracteriza por la siguiente ecuación:

$$\tau = k_t i \quad (2.21)$$

La fuerza contraelectromotriz ( $E$ ) es el voltaje que genera el propio motor como resultado de su rotación. Se induce debido a la interacción entre la bobina del campo magnético giratorio y el campo magnético estático.  $E$  es proporcional a la velocidad angular del motor y está representada por la ecuación:

$$E = k_e \dot{\theta} \quad (2.22)$$

Donde  $k_e$  es la constante de la fuerza electromotriz, que toma el mismo valor que  $k_t$ .

Utilizando las ecuaciones anteriormente mencionadas y aplicando la ley de voltaje de Kirchoff se obtiene la siguiente ecuación que caracteriza a un motor:

$$v - Ri - L \frac{di}{dt} - k_e \dot{\theta} = 0 \quad (2.23)$$

### III. MODELADO TRIDIMENSIONAL Y SIMULACIÓN

El primer paso para poder realizar simulaciones de funcionamiento del sistema es el diseño de los modelos mediante el uso de una herramienta de diseño asistido por ordenador (CAD). En este caso, se hará uso del software SolidWorks, se decidió por este software debido a su facilidad de uso, así como a la gran variedad de utilidades que brinda para diseñar piezas.

#### 3.1 DISEÑO DEL MODELO MEDIANTE SOLIDWORKS

Para el diseño de la base se busca una geometría que sea simétrica para facilitar el equilibrio de este. El diseño final se asimila a un figura humana con los brazos extendidos de forma horizontal. Se ha diseñado un agujero en la parte superior del modelo para colocar el motor y conectar el eje a la rueda. Además, se han diseñado pequeños agujeros para facilitar la conexión de los elementos electrónicos montados sobre el modelo. Por último, se ha diseñado una pequeña ranura en la parte inferior de la base para poder montar el soporte. El diseño final de la base se muestra a continuación.



*Figura 15. Base del péndulo*

La rueda está formada por un disco central con un agujero para conectar el eje del motor y un anillo exterior. En el disco central existen dos pequeños agujeros que permiten acoplar el eje del motor con la rueda mediante tornillos. Además, la rueda consta con pequeños agujeros en el anillo exterior para agregar pequeñas masas, de esta forma se permite la variación del momento de inercia producido por la rueda sin necesidad de hacer modificaciones al diseño de la rueda.

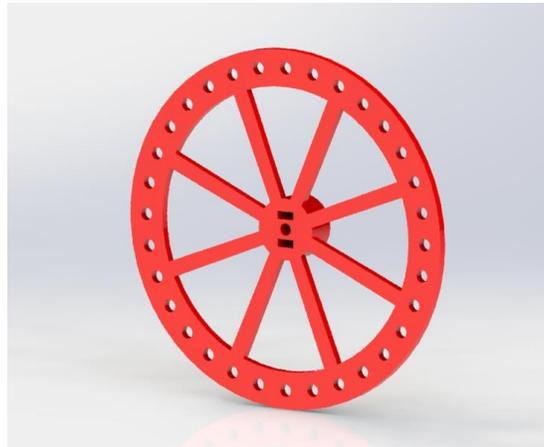


Figura 16. Volante de inercia

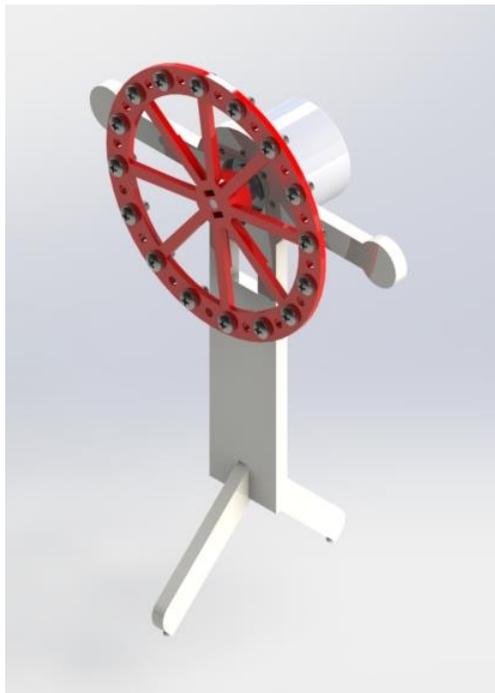
Finalmente, se ha diseñado un soporte para sostener al modelo, este soporte se acopla en la parte inferior de la base y permite el movimiento del modelo en el eje X. El soporte cuenta con dos ranuras para facilitar su posicionamiento sobre distintas superficies. El diseño del soporte se muestra en la siguiente ilustración.



Figura 17. Soporte del péndulo



Con las diferentes piezas diseñadas, se procede a realizar el ensamblaje para representar de una forma preliminar como sería el montaje final. Es importante mencionar que se ha ensamblado el motor a la parte superior de la base y además se agregaron pequeñas masas, en este caso tornillos, a la rueda para poder simular el montaje en la vida real. Agregando tornillos a la rueda se lograr modificar el momento de inercia generado por la rueda.



*Figura 18. Ensamblaje final del péndulo invertido*

Por último, realizando un análisis de masa se obtiene las propiedades de masa más importantes del ensamblaje, como la masa total, ubicación del centro de masa y momentos de inercia. La ubicación del centro de masa que es de gran ayuda al momento de realizar las simulaciones de funcionamiento.

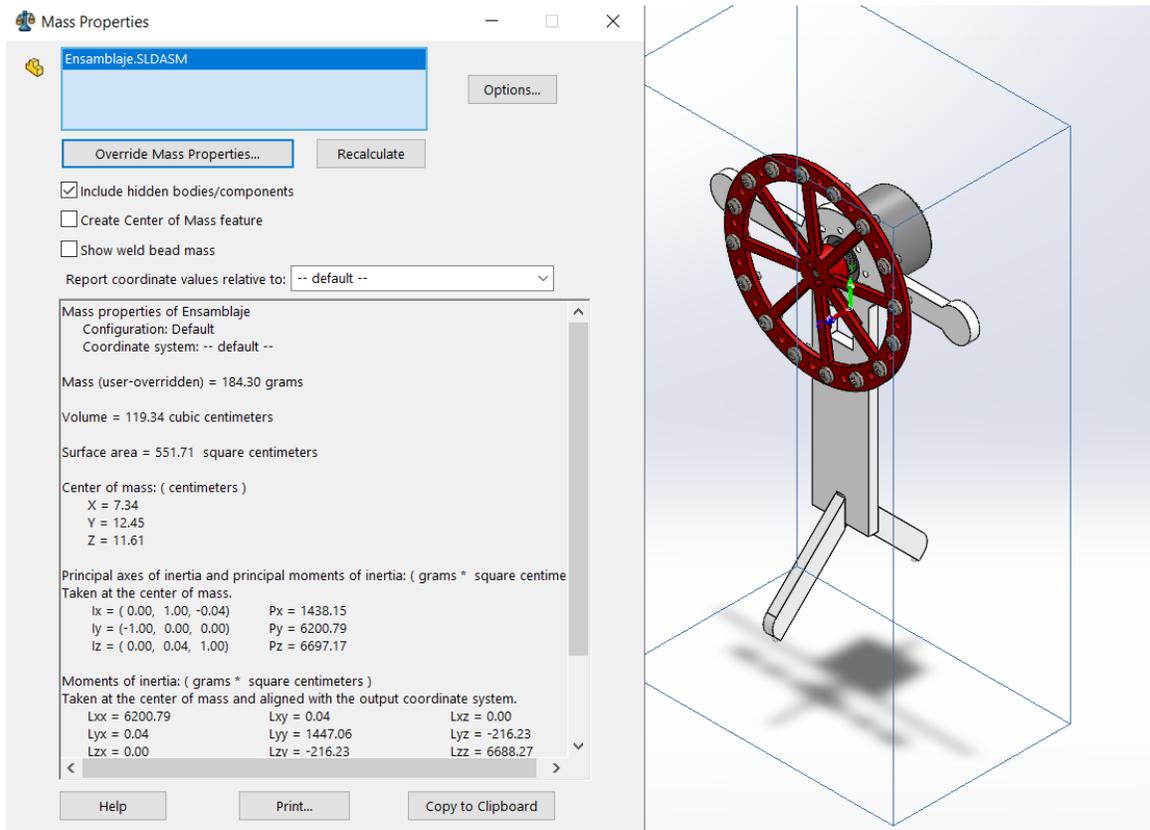


Figura 19. Análisis de masa del péndulo invertido

### 3.2 SIMULACIÓN DEL MODELO MEDIANTE SIMULINK

Para poder implementar el proyecto en la vida real es necesario llevar a cabo una serie de simulaciones para asegurarse del correcto funcionamiento del diseño y control del sistema. Para esto se ha diseñado un modelo en Simulink con la ayuda de las librerías de Simscape Multibody. Esta librería proporciona un entorno de simulación para sistemas mecánicos 3D, además, permite la utilización de bloques para representar piezas, uniones, restricciones de movimiento, elementos de fuerza y sensores.

El primer paso es introducir todos los modelos al entorno de simulación, esto se hace mediante el uso del bloque "File Solid". Cada pieza cuenta con puertos para facilitar las conexiones entre dos o más piezas, cada puerto está asociado a un eje de referencia que se puede colocar en cualquier característica geométrica de la pieza. Seguidamente, es necesario realizar las conexiones entre las piezas del modelo, para así poder representar la configuración final del modelo, esto es posible mediante la utilización de transformaciones geométricas que permiten modificar la posición y orientación de las piezas en el entorno.

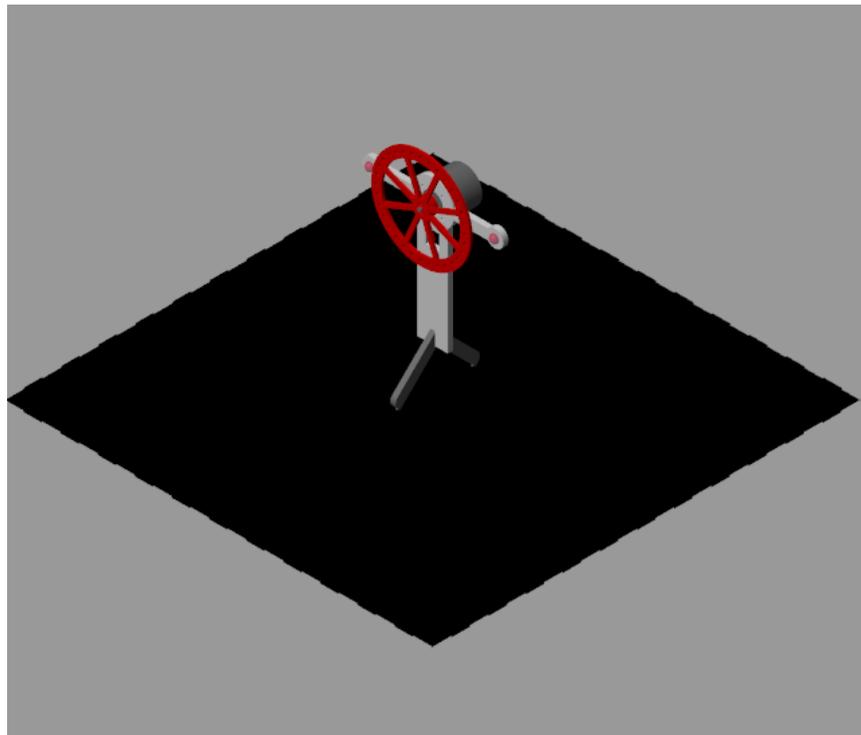
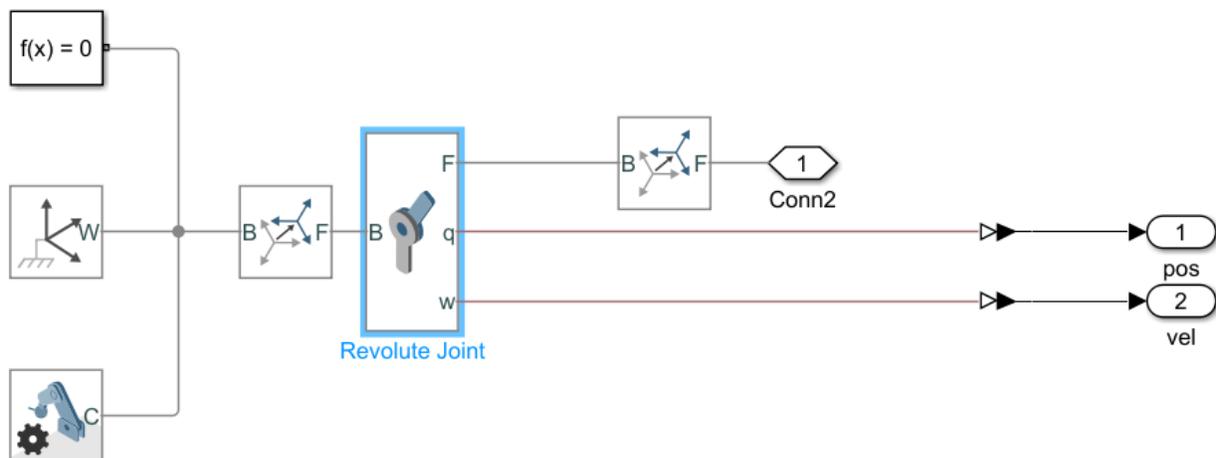


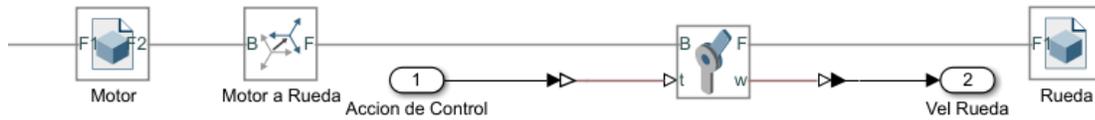
Figura 20. Montaje final del péndulo en simulink

En las conexiones donde exista una relación de movimiento entre dos o más piezas es necesario introducir juntas cinemáticas, como pares de revolución o prismáticos. Utilizando un par de revolución se introduce la primera relación de movimiento necesaria, la relación entre la base del modelo con el entorno. Creando esta relación de movimiento se hace posible el balanceo del modelo en el eje X horizontal.

Simulink permite la medición de magnitudes físicas como la velocidad, aceleración y par de las juntas cinemáticas sin la necesidad de incluir sensores complejos al modelo. Es tan fácil como habilitar la magnitud que se quiere medir en las opciones de sensado y se crea un puerto que proporciona el valor de dicha magnitud.



Para simular la rotación de la rueda conectada al eje de un motor es necesario agregar una junta de revolución que conecte la rueda y el eje. Para el control de la rueda se habilita el control por fuerza externa de la junta, esta opción permite el control de la junta mediante una señal externa, en este caso se utilizará el control por señal de par. Al igual que en la junta cinemática anterior, se habilitan las opciones de sensado, en este caso la magnitud a medir es la velocidad de la rueda, para así monitorear el comportamiento del motor.



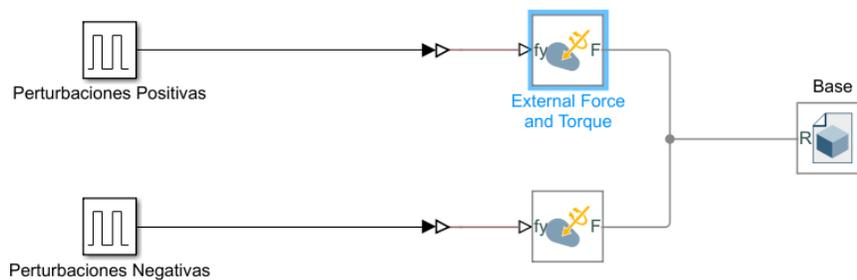
El siguiente paso consiste en introducir una superficie plana en donde se va a colocar el modelo, esto se hace con la ayuda del bloque "Infinite Plane" y bloques de contacto de fuerzas. Este último bloque se utiliza para aplicar una fuerza de contacto entre las dos geometrías a las que está conectado el bloque. Las fuerzas tienden a evitar la penetración al actuar en una dirección que acelera las geometrías alejándolas entre sí cerca del punto de contacto. Las fuerzas aplicadas son iguales y opuestas y se encuentran a lo largo de una línea de acción común.

Esta configuración permite que el modelo pueda situarse sobre este plano simulando así un suelo o una mesa en la vida real.

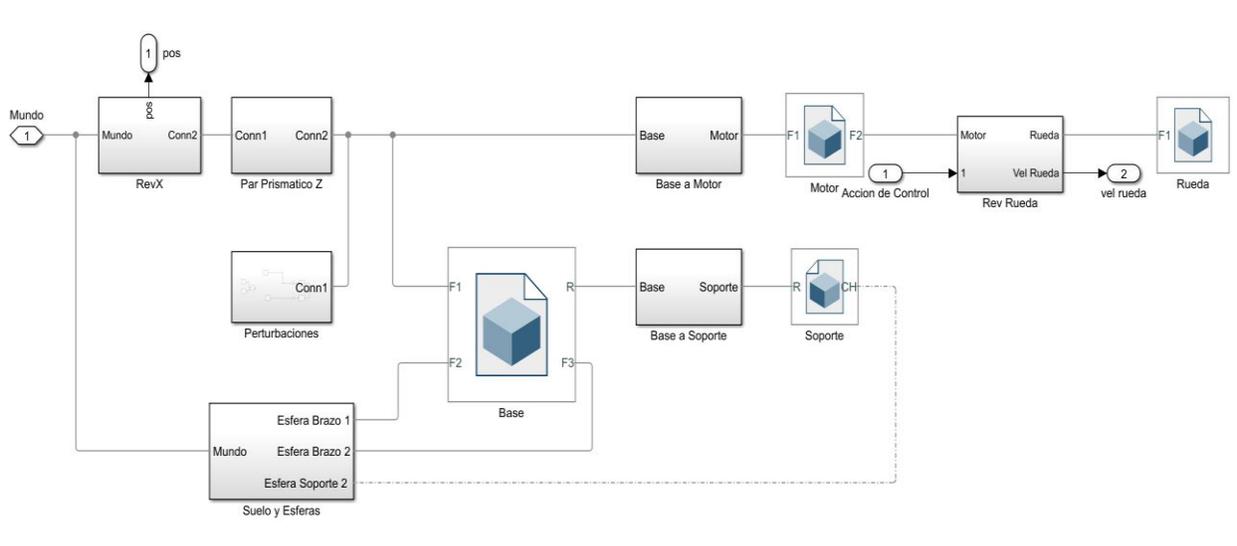


Por último, es necesario agregar perturbaciones al sistema para así poder simular pequeños movimientos del modelo que podrían ser causados por el viento o el contacto con un objeto. Estas perturbaciones tienen como objetivo desbalancear al sistema y provocar la caída de este. Para evitar la caída del modelo es necesario la inclusión de un controlador PID, este proceso se explica más en detalle en la siguiente sección del informe.

Las perturbaciones se simulan como fuerzas externas y se aplican al modelo mediante el bloque "External Torque and Force", este bloque aplica una fuerza externa y un par de torsión en la pieza adjunta. La fuerza y el par se especifican mediante las entradas de señales físicas. La señal física se crea con la ayuda del bloque "Pulse Generator", que permite generar una secuencia de pulsos, en nuestro caso perturbaciones. Estas fuerzas se aplican a la base y afectan a todas las piezas del modelo.



El modelo final del péndulo invertido con rueda de reacción se muestra a continuación:



### 3.3 CONTROL DEL PÉNDULO

El sistema de control que se ha implementado es un sistema de control en bucle cerrado basado en el algoritmo de control PID. Para asegurar un correcto funcionamiento del bucle de control es necesario realizar un ajuste de las ganancias del controlador.

El ajuste PID es el proceso de seleccionar valores apropiados para las ganancias proporcional ( $K_p$ ), integral ( $K_i$ ) y derivada ( $K_d$ ) de un controlador PID. El objetivo es lograr un control estable y receptivo. La sintonización se puede realizar manualmente mediante prueba y error, ajustando las ganancias en función de la respuesta del sistema, o utilizando métodos analíticos o técnicas automatizadas.

El primer paso del ajuste PID es ajustar la ganancia  $k_p$ , para esto se establecen  $k_i$  y  $k_d$  a cero y se aumenta  $k_p$  hasta que la respuesta del sistema muestre un nivel de control satisfactorio. Seguidamente se introduce la acción  $k_i$  para reducir el error de estado estacionario y mejorar la precisión del sistema. Por último, después de ajustar  $k_p$  y  $k_i$ , se agrega la acción derivada  $k_d$  para así proporcionar amortiguación y mejorar la estabilidad del sistema.

El proceso implica el ajuste iterativo de las ganancias, teniendo en cuenta factores como sobreoscilaciones, el tiempo de establecimiento y el error de estado estacionario, hasta que se logre un equilibrio óptimo entre la capacidad de respuesta y la estabilidad.

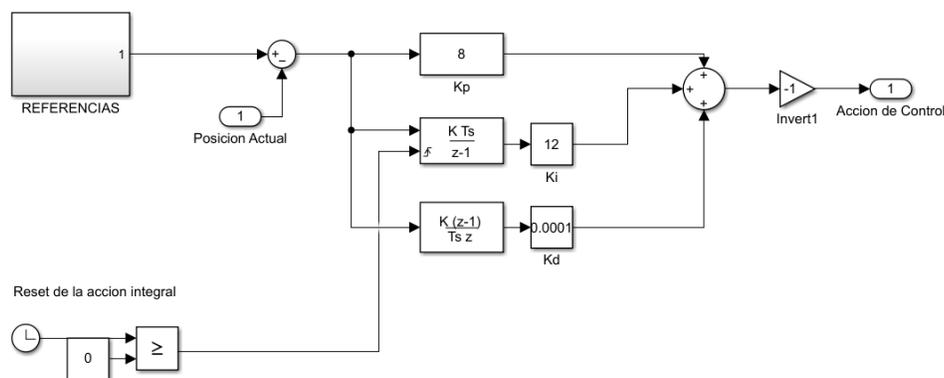


Figura 21. Bucle de control PID

### 3.4 RESULTADOS DE LAS SIMULACIONES

Con el objetivo de observar el comportamiento del péndulo y de poner a prueba el control diseñado se han definido varios casos de funcionamiento. Los casos se enumeran a continuación:

Rechazo de perturbaciones externas, en este caso el objetivo es mantener el péndulo en una posición vertical,  $0^\circ$  de inclinación, frente a perturbaciones con magnitudes aleatorias. Las perturbaciones se aplican como fuerzas perpendiculares a la base del péndulo.

Seguimiento de referencia senoidal, el objetivo es que el péndulo pueda seguir una referencia senoidal de amplitud y frecuencia definida. Este caso sirve para probar la respuesta del controlador frente a cambios constantes y permite analizar el error de seguimiento.

Seguimiento de referencia triangular, de forma parecida al caso anterior, el objetivo es que el péndulo siga una referencia triangular. En este caso los cambios en la señal de referencia son más bruscos que en los casos anteriores, por lo que es de utilidad para probar la velocidad de respuesta del control.

Las distintas referencias se pueden seleccionar mediante un selector ubicado en la pantalla principal del modelo, de la misma forma, se puede activar o desactivar las perturbaciones.

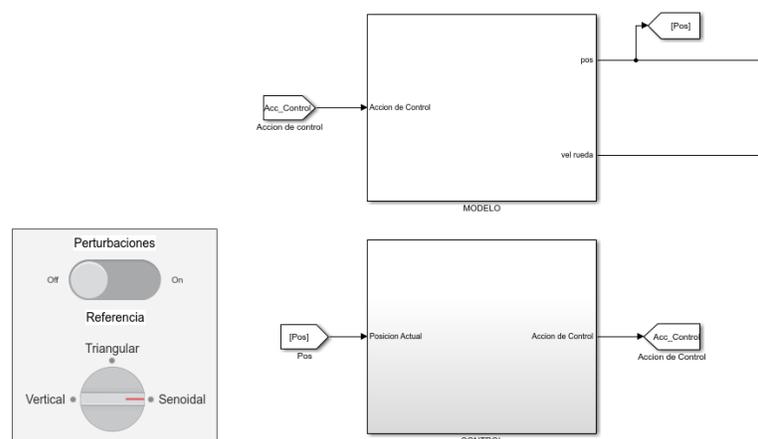


Figura 22. Selección de referencias

### 3.4.1 AJUSTE DE PARÁMETROS DEL CONTROLADOR

Como se mencionó en la sección anterior, el ajuste de los parámetros del controlador se lleva a cabo de forma iterativa, empezando con la ganancia proporcional y prosiguiendo con las demás ganancias. A continuación, se presenta un pequeño resumen del proceso de ajuste de los parámetros, es importante destacar que los valores de las ganancias se seleccionaron respetando los límites del motor y tratando de obtener resultados que puedan ser replicados en la vida real, ya que al utilizar valores elevados de  $k_p$  y  $k_i$  se pueden obtener respuestas muy buenas pero que no se asimilan a la realidad.

Si se aumenta el valor de la ganancia proporcional,  $k_p = 20$ , y se utiliza un valor de  $k_i$  que asegure un error de posición nulo. Cuando se aplica la perturbación, el péndulo cae hasta un ángulo de  $-2^\circ$  y luego se levanta. Por otra parte, el tiempo de establecimiento aumenta hasta los 10 segundos, un valor no aceptable, por lo que la respuesta no se da como satisfactoria. El péndulo en este caso es rápido para reaccionar ante la perturbación, pero es lento en el momento de levantamiento.

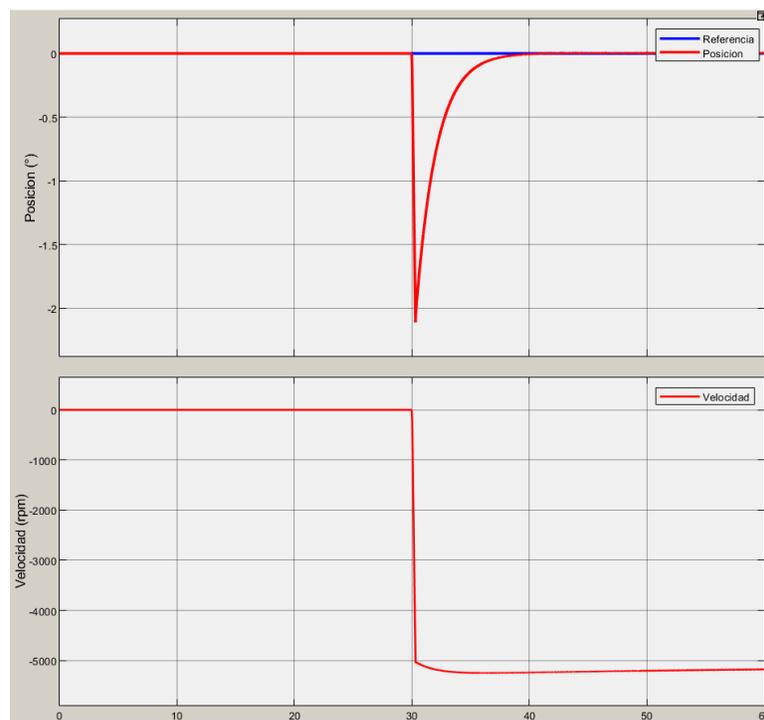


Figura 23. Respuesta con  $k_p$  elevada

Por otra parte, manteniendo un valor de  $k_p = 10$ , y aumentando el valor de la ganancia integral,  $k_i = 20$ , se disminuye el tiempo que tarda el péndulo en regresar a su punto de equilibrio. En comparación con el caso anterior, el péndulo cae un poco más, hasta un ángulo de  $-3.5^\circ$ . Las simulaciones permiten aumentar los valores de  $k_i$  hasta valores muy elevados, que brindan repuestas muy rápidas, pero que no se asimilan a la realidad, ya que, el aumento de esta ganancia conlleva en muchos casos a la saturación de la acción integral.

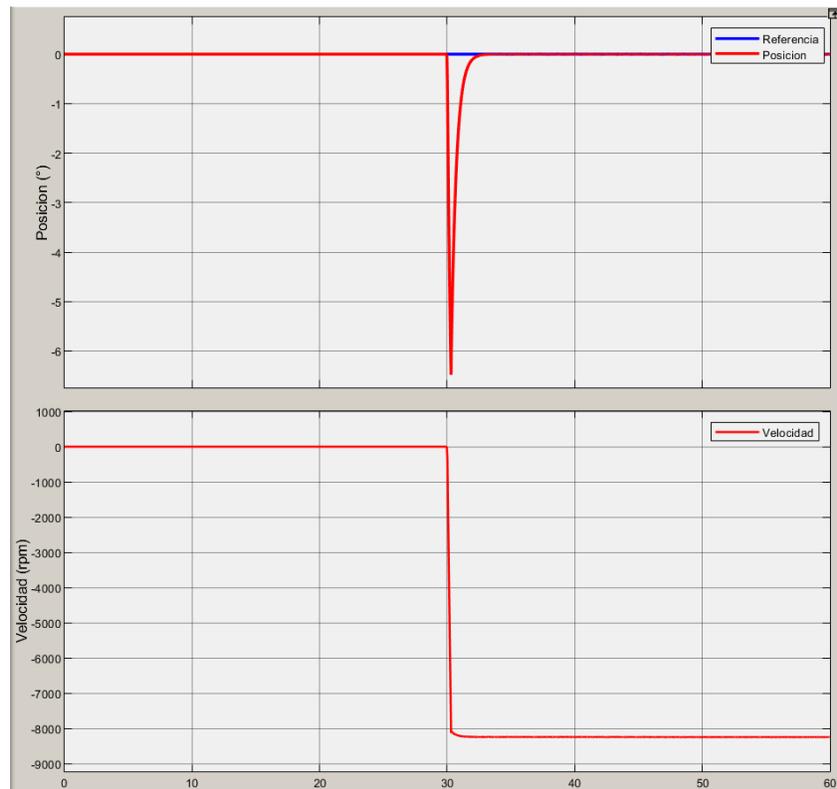


Figura 24. Respuesta con  $k_i$  elevada

Observando el comportamiento del péndulo y probando varios valores de las ganancias se puede llegar a un control adecuado del péndulo. Los valores de ganancias del PID utilizados son los siguientes:  $k_p = 8$ ,  $k_i = 12$ ,  $k_d = 0.003$ . Los resultados obtenidos se muestran en los siguientes casos de funcionamiento.

### 3.4.2 RECHAZO DE PERTURBACIONES

En esta prueba se parte desde la posición de equilibrio ( $0^\circ$ ) y se aplican perturbaciones de diferentes magnitudes cada 5 segundos. Manteniendo las ganancias del controlador mencionadas en el apartado anterior, en la siguiente figura se puede observar varias características de la respuesta del control diseñado. La primer característica destacable es el tiempo de establecimiento ( $t_e$ ), se puede observar que el  $t_e$  es de aproximadamente 2.5 segundos, teniendo en consideración la dinámica del motor se puede decir que es un valor aceptable para el tipo de aplicación.

De la misma forma, se observa que no se presentan sobreoscilaciones en la señal de respuesta, esto es un aspecto importante ya que el péndulo se estabiliza sin necesidad de sobrepasar la vertical.

Además, se observa como la rueda mantiene la velocidad después de rechazar la perturbación y alcanzar la referencia. El aumento o la reducción de la velocidad de la rueda depende del sentido en el que se aplica la perturbación, por lo que se debe de tener en cuenta que varias perturbaciones en un mismo sentido podrían significar la saturación del control.

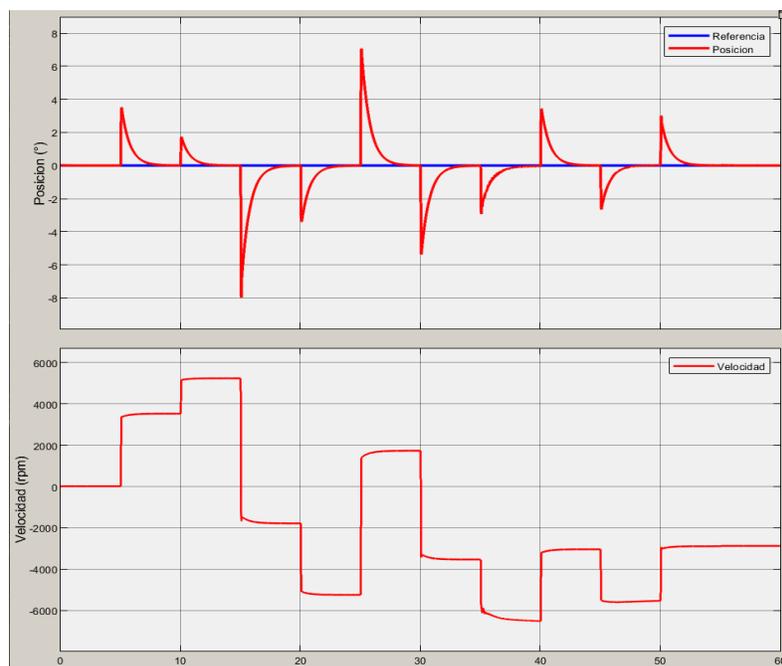


Figura 25. Rechazo de perturbaciones en posición vertical

### 3.4.3 SEGUIMIENTO DE REFERENCIA SENOIDAL

Al igual que en el caso anterior, el péndulo parte desde su posición de equilibrio y en este caso se aplica una referencia senoidal de  $8^\circ$  de amplitud con una frecuencia de 0.15 Hz. Manteniendo los valores del control se han obtenido respuestas buenas en el seguimiento de la referencia.

En la siguiente se pueden observar los resultados de la simulación, en la gráfica de posición se observa que en los recorridos lineales no se presenta un error de posición notable, mientras que en las crestas y los valles de la onda se presenta un leve error de posición, de  $0.3^\circ$  aproximadamente.

Además, se puede observar que, en los picos de la onda, la velocidad de la rueda es de 1100rpm, mientras en los valles de la onda la rueda llega al reposo.

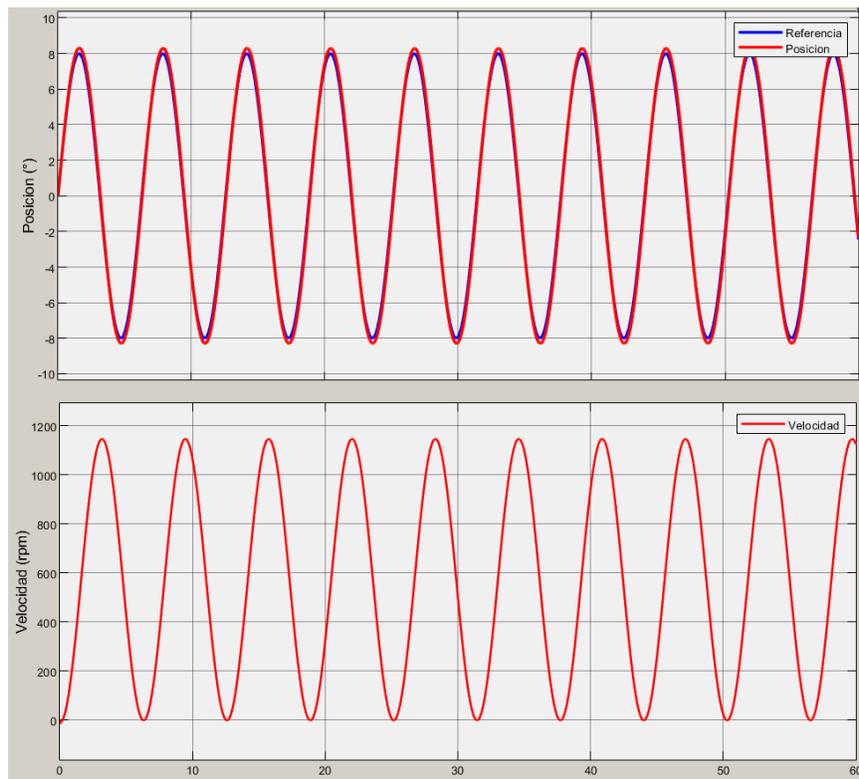


Figura 26. Seguimiento de referencia senoidal

### 3.4.4 SEGUIMIENTO DE REFERENCIA TRIANGULAR

Al igual que en los casos anteriores, el péndulo parte desde su posición de equilibrio y en este caso se aplica una referencia triangular de  $8^\circ$  de amplitud con una frecuencia de 1.5 Hz. Se han utilizado los mismos valores de las constantes del PID para comparar la respuesta del péndulo a las distintas referencias.

En la gráfica de posición se observa una mejora considerable del seguimiento de la referencia. En este caso, el error de posición es nulo a lo largo de la trayectoria del péndulo

Además, se puede observar que, en los picos superiores de la onda, la velocidad de la rueda es de aproximadamente 950rpm, mientras en los picos inferiores la rueda llega al reposo.

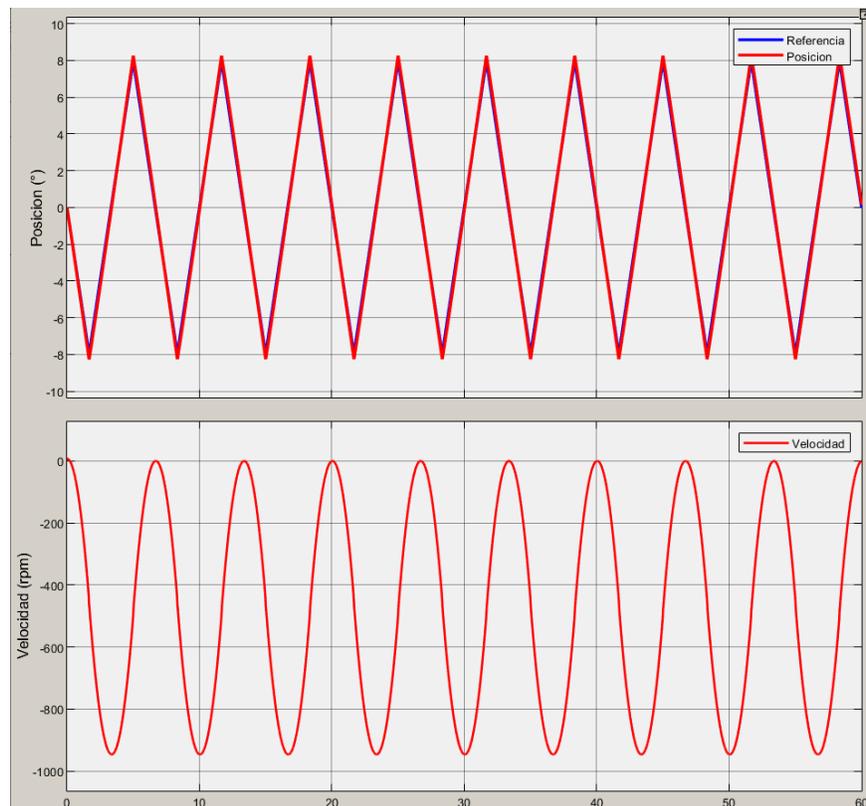


Figura 27. Seguimiento de referencia triangular

### 3.4.5 RECHAZO DE PERTURBACIONES EN SEGUIMIENTO DE REFERENCIAS

Al momento de introducir perturbaciones en los casos de seguimientos de referencias, se observa un cambio notable en la respuesta del sistema. Utilizando los mismos parámetros del PID que en los casos anteriores, procede a simular el caso de seguimiento de referencia senoidal.

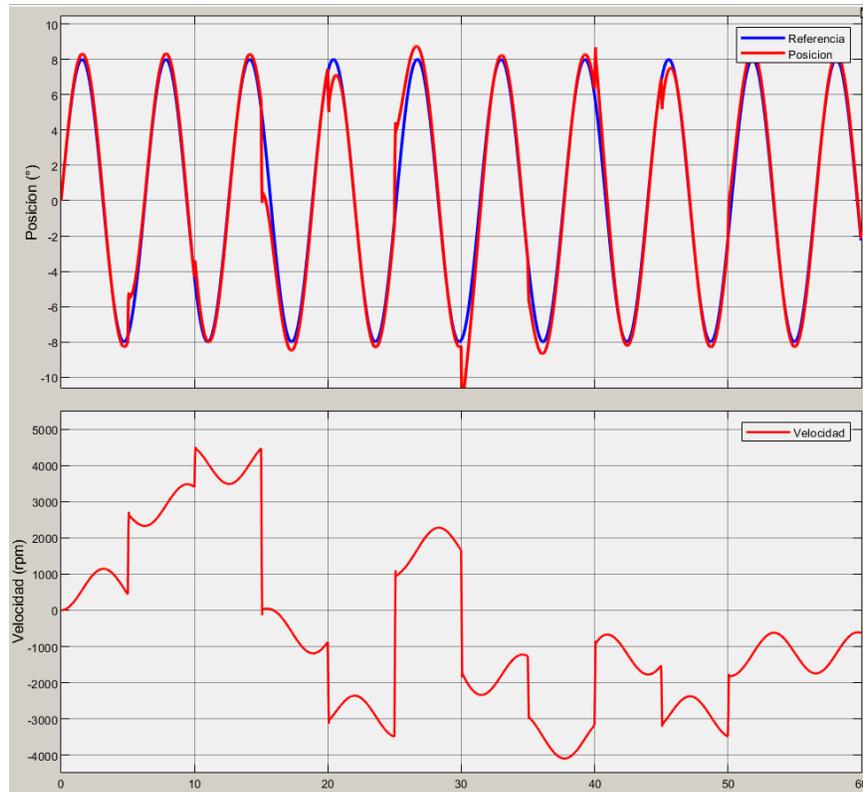


Figura 28. Seguimiento de ref. senoidal con perturbaciones

En la gráfica de posición se nota que al péndulo se le dificulta considerablemente el rechazo de las perturbaciones, lo que genera un error de posición grande y difícil de recuperar. En la gráfica de velocidad se observa que el motor acelera considerablemente para poder contrarrestar las perturbaciones, lo que en algunos casos lo puede llevar a superar sus límites de velocidad.

En el caso del seguimiento de la referencia triangular se presentan resultados parecidos al caso anterior.

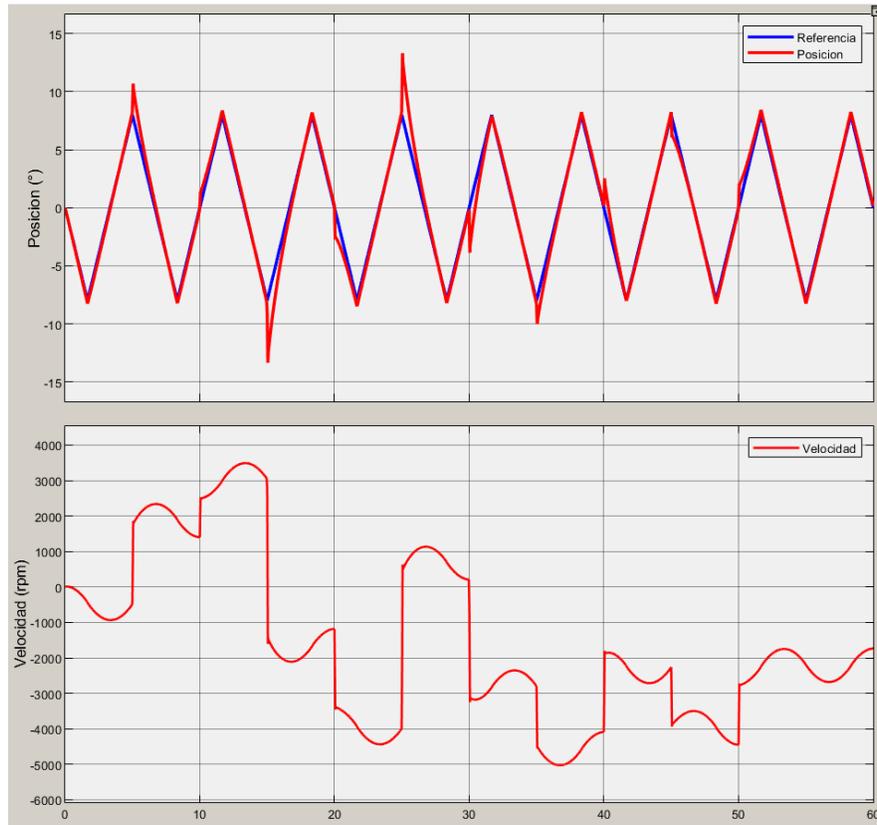


Figura 29. Seguimiento de ref. triangular con perturbaciones

En la grafica se observa una peor respuesta en comparación al caso anterior, esto es debido a que los cambios en la referencia triangular son más pronunciados y repentinos que en la senoidal. Además, cuando la perturbación se aplica en uno de los picos de la referencia, la respuesta empeora considerablemente debido a que el ángulo de inclinación ya es grande, lo que provoca una acción de control elevada.

## IV. IMPLEMENTACIÓN REAL

En esta sección del informe se detalla el proceso de implementación física del péndulo invertido con volante de inercia. En los siguientes capítulos se especifican los componentes utilizados, el código empleado para el control del péndulo, las dificultades encontradas al momento de la implementación, así como los resultados finales.

### 4.1 COMPONENTES

Los componentes mecánicos y electrónicos utilizados en la realización del proyecto se detallan a continuación.

#### 4.1.1 MOTOR ELÉCTRICO

El motor eléctrico es el componente más importante del mecanismo, ya que, el funcionamiento correcto del sistema depende del motor. Se necesita un motor liviano, con una respuesta rápida y que permita su control mediante señales PWM.

El motor seleccionado es el motor Nidec 24H677, este motor de CC sin escobillas incorpora un regulador de velocidad controlado mediante PWM, consta con un freno y provee una retroalimentación del valor de la velocidad del motor.



*Figura 30. Motor Nidec 24H677*



**Tabla 2. Especificaciones técnicas del motor Nidec24H677**

Descripción	Valor
Tensión de operación	12 V
Corriente de operación	0.7 A
Potencia	11 W
Par nominal	25 mN*m
Par máximo	70 mN*m
Velocidad nominal	4200 rpm
Velocidad máxima	5800 rpm
Peso	110 g

Fuente: (Nidec Corporation, 2023)

#### 4.1.2 SENSOR DE INCLINACIÓN

Para la medición del ángulo de inclinación del péndulo se necesita un sensor que sea preciso, pequeño y liviano. Estas características son imprescindibles al momento de seleccionar un sensor, ya que, se necesita un sensor de tamaño reducido que no afecte el centro de gravedad del mecanismo y que brinde una medida precisa y fiable del ángulo de inclinación, de esta forma, asegurando un mejor control del péndulo.

Teniendo en consideración estas características, se ha decidido utilizar el sensor MPU6050. Este dispositivo es un módulo de sensor de giroscopio y acelerómetro de 6 ejes que combina mediciones de velocidad de rotación y aceleración en un solo sensor. Esta integración permite un seguimiento preciso del movimiento en tres dimensiones, lo que la hace ideal para aplicaciones que requieren un control preciso del movimiento y la orientación.



Figura 31. Sensor MPU6050

Tabla 3. Especificaciones técnicas del sensor MPU6050

Descripción	Valor
Tensión de operación	5 V
Acelerómetro (rangos)	$\pm 2g$ , $\pm 4g$ , $\pm 8g$ , $\pm 16g$
Giroscopio (rangos)	$\pm 250^\circ/s$ , $\pm 500^\circ/s$ , $\pm 1000^\circ/s$ , $\pm 2000^\circ/s$
Comunicación	I2C
Dimensiones	21 x 16mm
Peso	2.1g

Fuente: (InvenSense, 2013)

#### 4.1.3 MICROCONTROLADOR

El microcontrolador se encarga de controlar el comportamiento del péndulo invertido, el microcontrolador recibe los datos medidos por el sensor de inclinación, calcula la acción de control del PID y controla la velocidad de giro del motor. Es necesario un microcontrolador con una buena velocidad de procesamiento para asegurar un tiempo de muestreo de 10ms,

Se ha seleccionado el Arduino DUE debido a sus especificaciones y precio en comparación con otros microcontroladores. El Arduino DUE está basado en el procesador Atmel SAM3X8E ARM Cortex-M3 de 32 bits y 84 MHz, esto lo convierte en un microcontrolador con una alta velocidad

de ejecución de instrucciones. Con estas especificaciones se es posible cumplir con el requerimiento de tiempo de muestreo planteado.

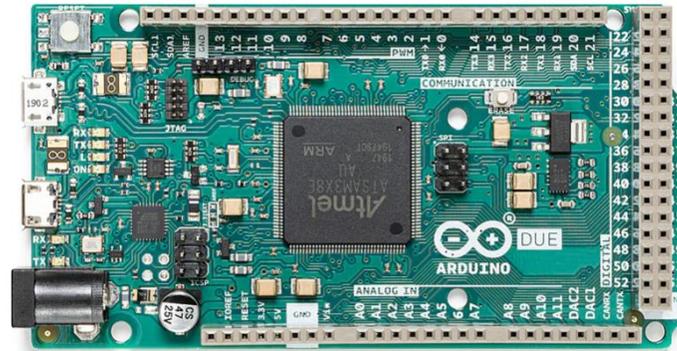


Figura 32. Arduino DUE

Tabla 4. Especificaciones técnicas del Arduino DUE

Descripción	Valor
Tensión de alimentación	7-12 V
Tensión de operación	3.3 V
Pines E/S digitales	54
Pines salida PWM	$\pm 250^\circ/s$ , $\pm 500^\circ/s$ , $\pm 1000^\circ/s$ , $\pm 2000^\circ/s$
Pines de entradas analógicas	12
Pines de salidas analógicas	2
Dimensiones	101.5 x 53.3 mm
Peso	36 g

Fuente: (Arduino, 2023)

#### 4.1.4 MÓDULO BLUETOOTH

Para facilitar la parametrización del PID y para controlar el estado del motor fue necesaria la implementación de un módulo de comunicación bluetooth. El módulo bluetooth HM-10 es el encargado de recibir los comandos enviados desde un celular y transmitirlos al Arduino.



Figura 33. Módulo Bluetooth HM-10

Tabla 5. Especificaciones técnicas del módulo bluetooth HM-10

Descripción	Valor
Tensión de alimentación	3.3 - 5 V
Corriente de operación	8.5 mA
Baudaje por defecto	9600 bauds
Alcance	< 100 m
Protocolo Bluetooth	4.0

Fuente: (Naylamp, 2023)

#### 4.1.5 FUENTE DE ALIMENTACIÓN DE CORRIENTE CONTINUA

Para el proceso de implementación del prototipo se ha utilizado una fuente de alimentación de CC variable, la fuente proporciona los 12 V para alimentar al motor.

#### 4.1.6 COMPONENTES 3D

Utilizando los modelos 3D diseñados en SolidWorks se ha llevado a cabo la impresión de todos los componentes necesarios para la implementación del prototipo. Como se mencionó anteriormente, los componentes a utilizar son: la rueda, la base y el soporte.



*Figura 34. Componentes 3D*

Para la impresión de las piezas se ha utilizado el material ABS con una resolución de 0.2mm y un relleno del 30%, con estos parámetros se han obtenido resultados satisfactorios al momento de unir las piezas y colocar la tornillería.

#### **4.2 MONTAJE**

Con la ayuda de tornillos y tuercas se montó el motor en la parte superior del modelo y se acopló la rueda al eje de este. El sensor de inclinación se montó en la parte frontal inferior del modelo para asegurar una correcta medida del ángulo en todo momento. El módulo bluetooth se encarga de recibir los comandos de inicio, paro y los cambios en los valores de las constantes del controlador PID. El montaje final del prototipo se muestra en la siguiente figura.

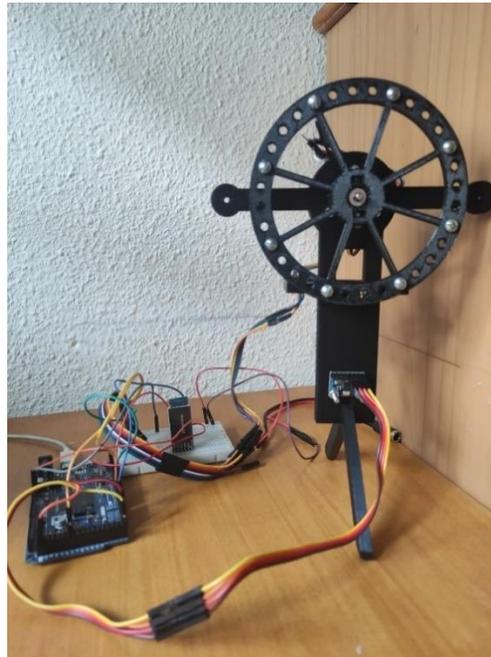


Figura 35. Montaje final

### 4.3 PROGRAMACIÓN DEL MICROCONTROLADOR

En esta sección se explica detalladamente los pasos a seguir para la programación del microcontrolador Arduino DUE y los componentes auxiliares en el entorno de programación Arduino IDE.

Como primer paso se deben de definir las librerías a utilizar, en este caso se utilizaron las librerías "MPU6050.h" y "pwm\_lib.h". La primera librería se utiliza para configurar y obtener las medidas del sensor MPU6050, mientras "pwm\_lib.h" se utiliza para generar una señal PWM de una frecuencia y ciclo de trabajo definido.

```
1 #include "MPU6050.h"  
2 #include "pwm_lib.h"
```

Figura 36. Definición de librerías

Seguidamente, se definen las variables a utilizar en el programa. Se definen los pines a los cuales se va a conectar el motor, conversiones útiles para el cálculo del ángulo de inclinación, variables para la implementación del controlador PID y algunas variables auxiliares.

```
// Motor Nidec 24H677
#define CW 0 // Sentido Horario
#define CCW 1 // Sentido Antihorario
#define Pin_BRK 3 // Freno (verde)
#define Pin_DIR 4 // Dirección (amarillo)
#define U_MAX 255 // Accion de control Max
```

Figura 37. Definición de pines

Luego, se debe de configurar el pin encargado de generar la señal PWM (pin 42), en este caso se necesita una señal de 20kHz, con un ciclo de trabajo variable.

```
// Señal PWM
using namespace arduino_due::pwm_lib;
#define PWM_PERIOD_PIN_42 5000 // 100 usecs in hundredth of usecs (1e-8 secs)
#define PWM_DUTY_PIN_42 4999 // 100 usec in hundredth of usecs (1e-8 secs)
pwm<pwm_pin::PWMH1_PA19> pwm_pin42;
```

Figura 38. Configuración PWM

El siguiente paso es la configuración inicial de todos los componentes a utilizar. Esta configuración se programa en el apartado "void setup()". Las instrucciones que se colocan en esta sección se ejecutan una sola vez al inicio de la ejecución.

En la configuración inicial se configuran los puertos de comunicación serie para poder recibir datos desde el modulo bluetooth y poder enviar datos al puerto serie del ordenador. Además, se configura el modo de utilización de los pines, en este caso solo se utilizarán pines en estado "OUTPUT". Por último, inicializa la generación de la señal PWM y el sensor MPU6050.

```
54 void setup() {
55 // Inicializacion de los puertos de comunicacion serial
56 Serial.begin(115200);
57 Serial3.begin(115200);
58
59 // Configuracion de los pines
60 pinMode(Pin_DIR,OUTPUT) ; pinMode(Pin_BRK,OUTPUT) ;
61 digitalWrite(Pin_BRK,HIGH) ;
62 pinMode(RT,OUTPUT) ;
63
64 // Inicializacion de la señal PWM
65 pwm_pin42.start(PWM_PERIOD_PIN_42,PWM_DUTY_PIN_42);
66
67 // Inicializacion del sensor
68 init_MPU6050(33,31) ;
69
70 }
```

Figura 39. Configuración inicial

En el apartado “void loop()” se colocan las instrucciones que se ejecutan continuamente. En esta sección se programa el controlador PID así como los demás aspectos del proceso, como la lectura del ángulo de inclinación, la recepción de comandos desde el modulo bluetooth y el control del motor.

Se ha configurado un sistema de inicio/paro del sistema a través de comandos recibidos por bluetooth, cuando el sistema está en modo paro, se activa el freno del motor y se envía la señal PWM correspondiente que corresponde al paro del motor, en este caso el motor Nidec 24H677 se para con una señal con ciclo de trabajo del 100%.

```
u = 0; acc_int = 0; acc_der = 0; err_ant = 0; int_vel = 0; ref = 0;
digitalWrite(Pin_BRK,LOW); // Se activa el freno
change_duty(pwm_pin42,5000,PWM_PERIOD_PIN_42); // Se envia señal de paro al motor
```

Además, por motivos de seguridad, cuando el péndulo presenta una inclinación mayor a los 12°, se envía la señal de paro al motor.

Seguidamente, se procede a calcular la acción de control necesaria para mantener el péndulo en su posición de equilibrio. Para esto se debe de calcular la acción proporcional, integral y derivativa. El proceso de calculo y ajuste de los parámetros del controlador se mencionó en la sección del marco teórico. Dependiendo del signo de la acción de control se define el sentido de giro del motor.

```
99     err = (ref - Xf); // Se calcula el error
100     acc_int += constrain(err * 0.01,-U_MAX/1.5,U_MAX/1.5); // Se calcula la accion integral
101
102     // Reduccion de velocidad del motor en posicion vertical
103     if (abs(int(Xf)) < 3 ) {
104         int_vel += Xf;
105     } else int_vel = 0;
106
107     acc_der = (err - err_ant) / 0.010 ; // Se calcula la accion derivativa
108
109     u = constrain(kp * err + ki * acc_int + kd * acc_der +kr * -int_vel,-U_MAX,U_MAX); // Se calcula la accion de control
110     change_duty(pwm_pin42,map(abs(int(u)), 0, U_MAX, 5000, 0),PWM_PERIOD_PIN_42); // Se cambia el ciclo de trabajo del motor
111
112     if (int(u) <= 0) digitalWrite(Pin_DIR, LOW); // Dirección CCW Izquierda
113     else digitalWrite(Pin_DIR, HIGH); // Dirección CW
```

Figura 40. Controlador PID



Finalmente, se definen las funciones auxiliares que van a ser utilizadas en el código. En este caso se crearon seis funciones, entre ellas están: `init_MPU6050`, `muestra_MPU6050`, `Parámetros`, `Valores`, `Medidas` y `Medidas1`.

Las primeras dos funciones se utilizan para inicializar y calibrar el sensor y para calcular el ángulo de inclinación en base a los datos medidos. La función `Parámetros` se utiliza para procesar los caracteres recibidos a través del módulo bluetooth. Las funciones restantes se utilizan para visualizar medidas importantes en la consola serial.

Los extractos de código correspondientes a las funciones auxiliares se muestran en la sección de anexos.

#### **4.4 DIFICULTADES EN LA IMPLEMENTACIÓN**

Inicialmente se había optado por utilizar el motor Nidec 24H404 de 24V, proporcionaba mejores especificaciones que el modelo 24H677. El diseño de la rueda se había diseñado tomando en consideración el acople del motor mencionado, pero el motor presentó problemas de funcionamiento al poco tiempo por lo que se tuvo que cambiar a otro motor del mismo fabricante. El nuevo motor utiliza un diferente acople que creó dificultades al momento de montar la rueda. Realizando unas pequeñas modificaciones en el montaje, fue posible acoplar la rueda al eje del motor. No obstante, el montaje de la rueda no es el más indicado y debido a esto se generan unas pequeñas vibraciones en el péndulo cuando la rueda gira a altas velocidades. Como el motor no gira constantemente a velocidades elevadas, las vibraciones no afectan el comportamiento del sistema.

El motor Nidec 24H677 está diseñado para ser controlado mediante señales PWM de 20.000 Hz de frecuencia. El Arduino DUE genera señales PWM de 1000 Hz por defecto, lo que provoca problemas, como vibraciones y ruidos, en el funcionamiento del motor.

Para solucionar este problema, fue necesario investigar diferentes formas de generar señales PWM de altas frecuencias. Se optó por utilizar una librería para la generación de PWM a diferentes frecuencias. Esta librería permite la modificación del tiempo de muestreo y ciclo de trabajo de la



señal generada y de esta forma se puede generar una señal de 20.000 Hz. Al trabajar a la frecuencia indicada por el distribuidor, el motor presenta mejoras notables en su funcionamiento.

En el proceso de ajuste de los parámetros del controlador PID, el motor empezó a presentar pequeñas vibraciones en su funcionamiento, dichas vibraciones aprecian de forma intermitente, de todas formas, se continuó con la implementación, pero pasado el tiempo el motor dejó de funcionar. Esto pudo ser causado por los cambios repentinos en la dirección y velocidad de giro cuando el péndulo se situaba cerca de la posición de equilibrio. Debido a cuestiones de tiempo se imposibilitó la compra de un motor de repuesto para continuar con la implementación real.



## **V. PLIEGO DE CONDICIONES**

### **5.1 DEFINICIÓN Y ALCANCE DEL PLIEGO**

El objeto de este pliego de condiciones es establecer las condiciones necesarias para el diseño, simulación e implementación de un mecanismo de péndulo invertido con volante de inercia. En este pliego se cubren los etapas principales del desarrollo del prototipo, siendo estas, la etapa de modelado tridimensional, simulación, programación e implementación real.

El alcance del proyecto contempla el diseño del sistema mecánico, incluyendo el péndulo, el volante de inercia y los componentes de soporte, la simulación del funcionamiento del péndulo utilizando MATLAB/SIMULINK, la selección de los materiales adecuados para el montaje del sistema, el diseño del sistema de control y la construcción y ensamblaje del péndulo invertido.

### **5.2 CONDICIONES GENERALES**

Con el motivo de cerciorarse que la persona que realice el proyecto no corra riesgos en el proceso de puesta en marcha del prototipo, es de carácter obligatorio el cumplimiento de toda la normativa relacionada a la realización de proyectos electrónicos de baja tensión.

### **5.3 CONDICIONES TÉCNICAS**

#### **5.3.1 OBJETO**

La finalidad de esta sección es establecer las condiciones técnicas para el correcto desarrollo de todos los aspectos del proyecto.

#### **5.3.2 CONDICIONES DE LOS MATERIALES**

Los componentes a emplear serán de primera calidad y deberán cumplir toda la normativa establecida para su uso. Además, para asegurar su correcto funcionamiento, todos los materiales deberán de ser sometidos a pruebas que acrediten su calidad. El uso de estos componentes



deberá de ir complementado por la especificaciones detalladas en las fichas técnicas de cada componente.

### 5.3.3 CONDICIONES DE LA EJECUCIÓN

Al establecer condiciones de ejecución se puede garantizar un desarrollo fluido de las etapas de modelado, simulación e implementación del proyecto. Por esto, es de vital importancia el establecimiento de requisitos mínimos de software y hardware.

Los requisitos de software principales son:

- Sistema operativo Windows 7 o superior
- SOLIDWORKS 2018 o superior
- MATLAB 2018b o superior
- Arduino IDE

De la misma forma, se recomiendan una serie de requisitos mínimos de hardware.

- CPU de doble núcleo
- Memoria RAM de 8GB
- 20 GB de almacenamiento libre
- GPU RAM de 2GB

### 5.3.4 PRUEBAS DE FUNCIONAMIENTO

Las pruebas de funcionamiento deberán de ser realizadas en un lugar designado para la elaboración de proyectos de esta índole, lugar que cuente con las protecciones necesarias para el trabajo con dispositivos de baja tensión.



## VI. PRESUPUESTO

El presupuesto para la elaboración del prototipo incluyendo las horas de trabajo y la compra de componentes se especifica en la tabla a continuación.

*Tabla 6. Presupuesto*

Descripción	Costo unitario (€)	Cantidad	Coste (€)
Motor NIDEC 24H677	5,00	1	5,00
Microcontrolador Arduino DUE	50,00	1	50,00
IMU MPU6050	3,00	1	3,00
Modulo Bluetooth HM-10	3,00	1	3,00
Fuente de voltaje	9,00	1	9,00
Placa breadboard	5,00	1	5,00
Cables de conexión varios	0,20	35	7,00
Tornillos y tuercas	0,03	40	1,2
Horas – Diseño de piezas 3D	15,00	25	375,00
Horas – Simulación Matlab/Simulink	20,00	50	1.000,00
Horas – Programación Arduino	20,00	45	900,00
Horas – Implementación	25,00	70	1.750,00
Impresión 3D - Base	10,50	1	10,50
Impresión 3D - Soporte	5,35	1	5,35
Impresión 3D – Rueda	7,15	1	7,15
<b>TOTAL</b>			<b>4.131,20</b>



## VII. CONCLUSIONES

1. Se ha realizado un estudio exhaustivo sobre el comportamiento, aplicaciones y componentes utilizados en el sistema del péndulo invertido con volante de inercia. Esto ha permitido adquirir un sólido conocimiento del funcionamiento del sistema y sentar las bases para los pasos siguientes.
2. Se ha formulado un modelo matemático prototipo que tiene en cuenta tanto la dinámica del péndulo como la del motor. Este modelo proporciona una representación precisa del sistema y servirá como base para el diseño y control posteriores.
3. Los componentes mecánicos del péndulo se han diseñado utilizando SOLIDWORKS, lo que ha permitido crear modelos virtuales y realizar modificaciones antes de la fabricación. Esto garantiza la precisión y la eficiencia en el proceso de diseño.
4. Se ha creado un modelo del prototipo en Simulink, lo que ha facilitado el estudio y comprensión del comportamiento del péndulo en un entorno simulado. Esto ha permitido identificar posibles problemas y optimizar el diseño antes de pasar a la etapa de la implementación real.
5. Mediante el uso de Simulink, se han realizado simulaciones exhaustivas del funcionamiento del sistema. Estas simulaciones han permitido analizar en detalle el comportamiento del péndulo en diferentes escenarios y condiciones, lo que ha ayudado a optimizar aún más el diseño y el control.
6. Se ha construido el prototipo final del péndulo utilizando tecnología de impresión 3D y ensamblaje de los componentes mecánicos diseñados previamente. Esto ha permitido materializar el diseño y obtener un prototipo físico que representa fielmente el sistema diseñado.
7. Se ha programado el microcontrolador responsable de controlar el comportamiento del sistema. En el proceso de ajuste de los parámetros del controlador PID, se ha estropeado el motor, lo que imposibilitó la implementación real.



## VIII. TRABAJO A FUTURO

Debido a que el prototipo es sensible a pequeñas perturbaciones, es importante disminuir la cantidad de aspectos del montaje que puedan afectar el comportamiento de manera negativa. Uno de los principales factores que influyen en la estabilidad del sistema son los cables que conectan a los sensores y el motor al microcontrolador. Una posible mejora es la sustitución del Arduino DUE por un Arduino NANO montado sobre el prototipo, esto aislaría al prototipo de componentes externos.

Debido a que la lectura del tacómetro del motor utilizado no es fiable, se recomienda utilizar un motor distinto que brinde una medida precisa de la velocidad del motor en este momento. Conociendo el valor de la velocidad se podría implementar una realimentación para reducir la velocidad de la rueda cuando el péndulo está en su posición de equilibrio, así evitando que la rueda gire a altas velocidades.

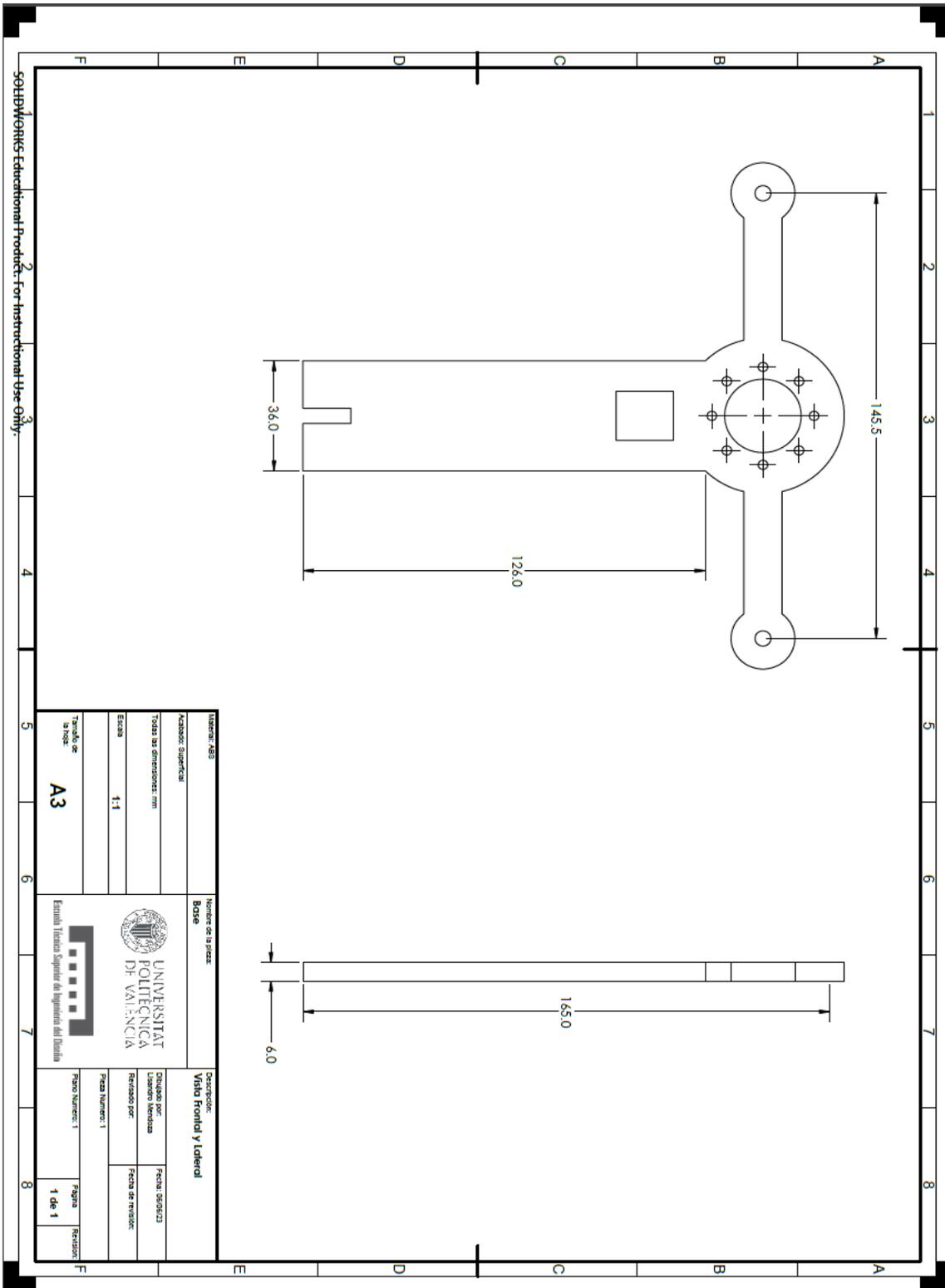


## IX. BIBLIOGRAFÍA

1. 5Hertz. (2023). *Introducción al giroscopio*. Recuperado de [https://www.5hertz.com/index.php?route=tutoriales/tutorial&tutorial\\_id=13](https://www.5hertz.com/index.php?route=tutoriales/tutorial&tutorial_id=13)
2. Arduino. (2023). *Arduino Due*. Recuperado de <https://docs.arduino.cc/hardware/duo>
3. Belascuen, A. (2018). *Design, Modeling and Control of a ReactionWheel Balanced Inverted Pendulum*.
4. Díaz, P. Á. (2017). *Metodología para el diseño de envolventes de volantes de inercia*. Recuperado de [https://e-archivo.uc3m.es/bitstream/handle/10016/27857/TFG\\_Pablo-Angel\\_Prieto\\_Diaz.pdf?sequence=1&isAllowed=y](https://e-archivo.uc3m.es/bitstream/handle/10016/27857/TFG_Pablo-Angel_Prieto_Diaz.pdf?sequence=1&isAllowed=y)
5. InvenSense. (2013). *MPU-6050 Product Specification*. Recuperado de <https://invensense.tdk.com/wp-content/uploads/2015/02/MPU-6000-Datasheet1.pdf>
6. Mayuresh. (2020). *Electric Motors*. UNICORN Publishing Group.
7. Naylamp. (2023). *MÓDULO BLUETOOTH 4.0 BLE HM-10*. Recuperado de <https://naylampmechatronics.com/inalambrico/133-modulo-bluetooth-40-ble-hm-10.html>
8. Nidec Corporation. (2023). *CMC\_24H series*. Recuperado de <https://www.nidec.com/en/product/search/category/B101/M102/S100/NCJ-24H-24-01/>
9. Omega. (2023). *Accelerometers*. Recuperado de <https://www.omega.com/en-us/resources/accelerometers>
10. SBG Systems. (2023). *IMU - Unidad de medición inercial*. Recuperado de <https://www.sbg-systems.com/es/unidad-de-medicion-inercial-sensor-imu/#:~:text=IMU%2C%20que%20significa%20Unidad%20de,necesidades%20de%20cabo%20%2C%203%20magnet%C3%B3metros>.
11. Thasleem, M. (2015). *Swing Up and Balancing of a Reaction Wheel Inverted Pendulum*.
12. TME. (2020). *Motor Paso a Paso*. Recuperado de <https://www.tme.eu/es/news/library-articles/page/41861/Motor-paso-a-paso-tipos-y-ejemplos-del-uso-de-motores-paso-a-paso/#:~:text=El%20motor%20paso%20a%20paso,eje%20cada%201%2C8%20B0>.
13. Wang, L. (2020). *PID Control System Design and Automatic Tuning Using MATLAB/Simulink*. Wiley. Recuperado de [https://www.google.es/books/edition/PID\\_Control\\_System\\_Design\\_and\\_Automatic/1NXKDwAAQBAJ?hl=en&gbpv=0](https://www.google.es/books/edition/PID_Control_System_Design_and_Automatic/1NXKDwAAQBAJ?hl=en&gbpv=0)



## X. PLANOS

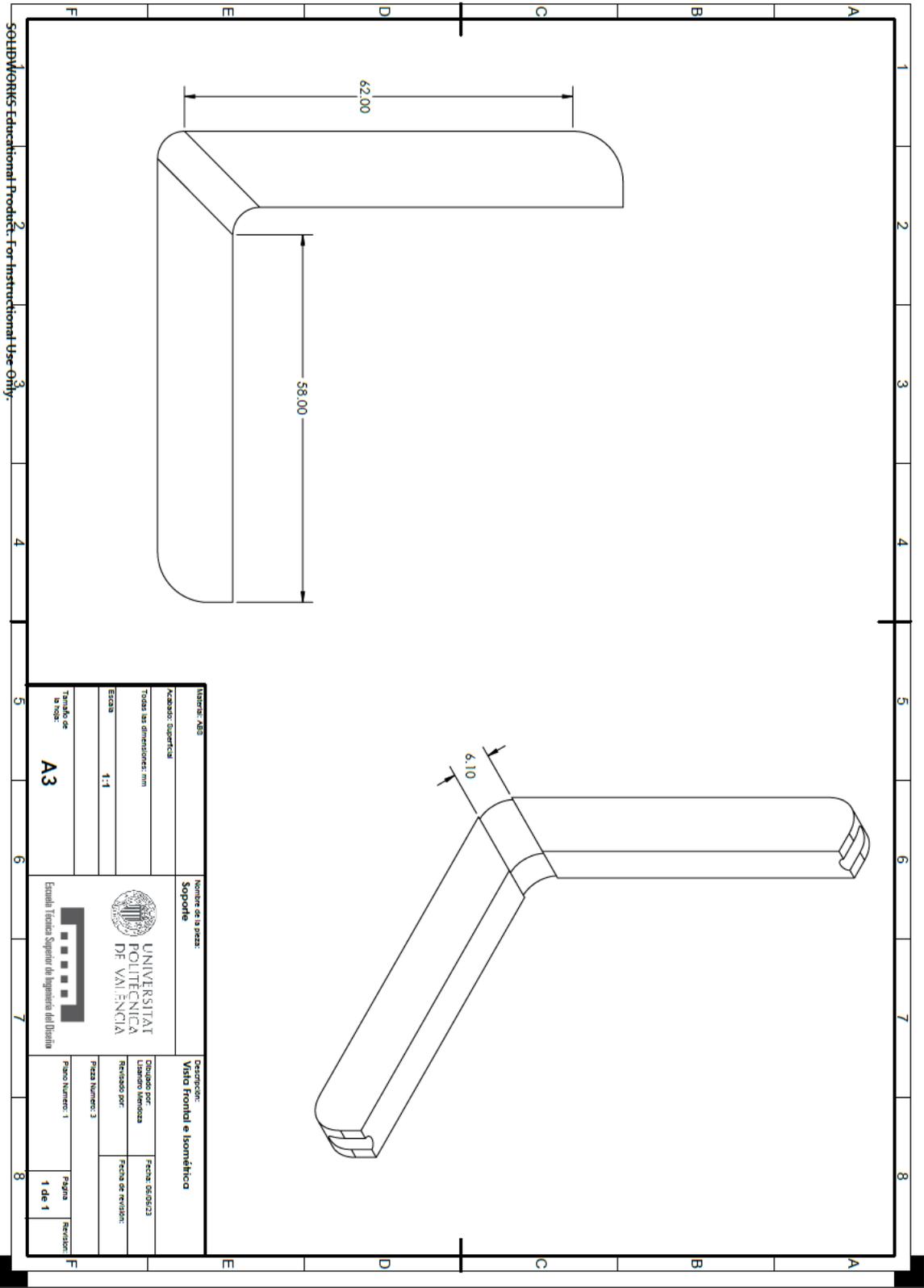


SOLIDWORKS Educational Products: For Instructional Use Only.

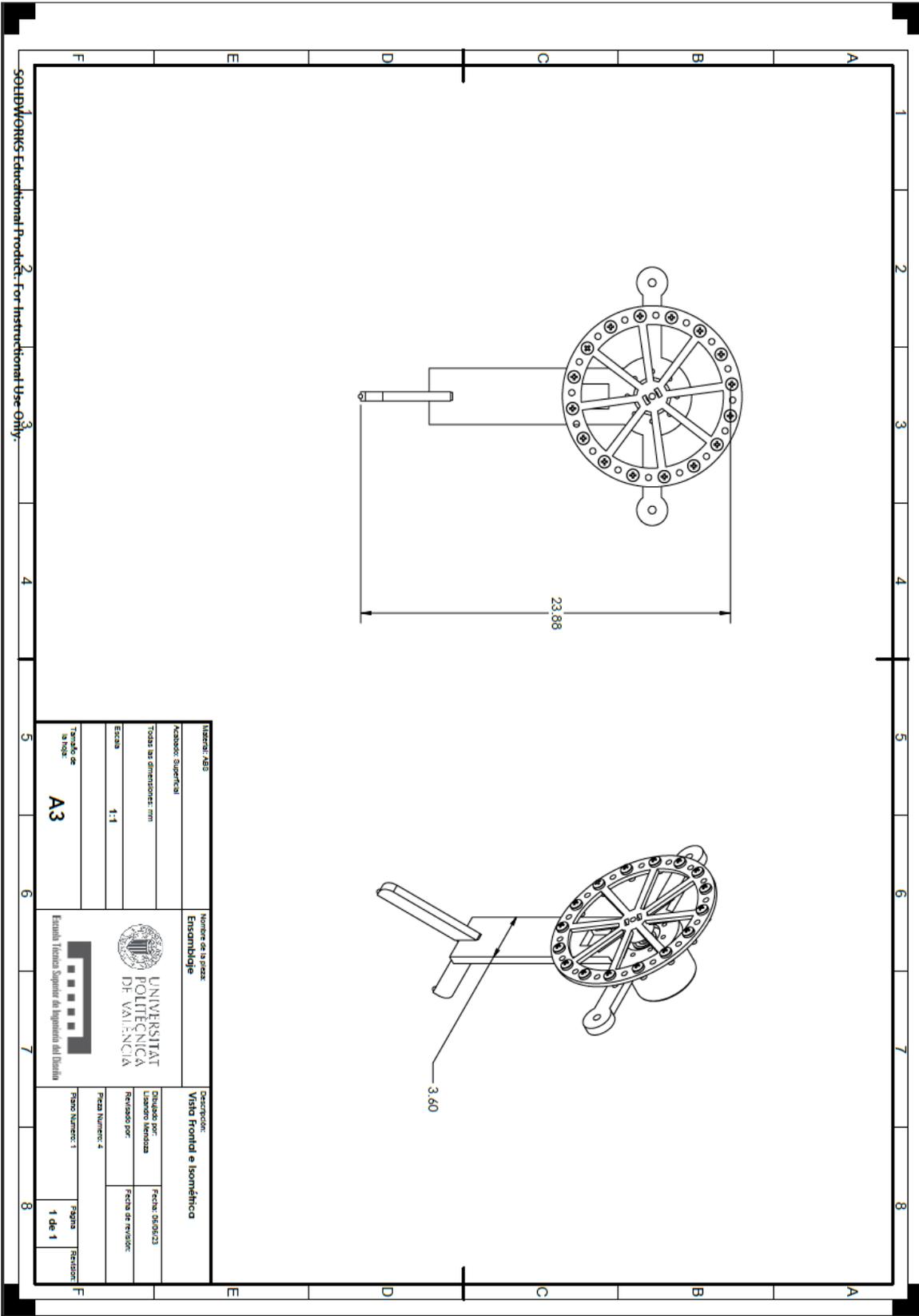
Director: ABS Acetador: Supercal	Nombre de la pieza: <b>Rueda</b>	Descripción: <b>Vista Frontal e Isométrica</b>
Todas las dimensiones: mm Escala: 1:1		Trabajo por: Cliente: Mercedes Revisado por: Fecha de revisión:
Título de la hoja: <b>A3</b>	Escuela Técnica Superior de Ingeniería del Diseño	Hoja: <b>1 de 1</b>
		Fecha Número: 1 Fecha Número: 2



SOHWORKS Educational Product - For Instructional Use Only



Número: 423 Nombre de la pieza: <b>Soporte</b>		Descripción: <b>Vista frontal e Isométrica</b>	
Acabado: Superficial Todas las dimensiones: mm Escala: 1:1		Diseñado por: Revisado por: Fecha de revisión:	
Trabajo de: a: <b>A3</b>		Pieza Número: 3 Plano Número: 1 Página: 1 de 1	
UNIVERSITAT POLITÈCNICA DE VALÈNCIA Escuela Técnica Superior de Ingeniería del Diseño		Fecha de entrega: Revisor:	





## XI. CÓDIGO DE ARDUINO

```
#include "MPU6050.h"
#include "pwm_lib.h"

// Declaracion de constantes y pines

// Motor Nidec 24H677
#define CW 0 // Sentido Horario
#define CCW 1 // Sentido Antihorario
#define Pin_BRK 3 // Freno (verde)
#define Pin_DIR 4 // Dirección (amarillo)
#define U_MAX 255 // Accion de control Max

// Señal PWM
using namespace arduino_due::pwm_lib;
#define PWM_PERIOD_PIN_42 5000 // 100 usecs in hundredth of usecs (1e-8 secs)
#define PWM_DUTY_PIN_42 4999 // 100 usec in hundredth of usecs (1e-8 secs)
pwm<pwm_pin::PWMH1_PA19> pwm_pin42;

// Sensor MPU6050
#define A_R 16384.0
#define G_R 131.0
#define RAD_A_DEG = 57.295779
MPU6050 mpu ;
float X = 0.0, Xi = 0.0, Xf = 0.0, Xf_m1;

// Tiempo de muestreo
#define RT 13
long t1 = 0 ;
long t2 = 0 ;
int T = 10 ; // Tiempo de muestreo objetivo

// Configuracion Inicial PID
float ref = 0.0; // Señal de referencia
float y = 0.0; // Señal de retroalimentacion (posicion del pendulo)
float err = 0.0; // Señal de error
float err_acum = 0.0; // Error acumulado
float err_ant = 0; // Error ciclo anterior
float u = 0.0; // Accion de control
float kp = 16.0; // Constante accion proporcional
float ki = 0.0; // Constante accion integral
```



```
float kd = 0.0;          // Constante accion derivativa
float kr = 0.0;          // Constante reduccion de velocidad
float acc_int = 0.0;     // Accion integral
float acc_der = 0.0;     // Accion derivativa
float int_vel = 0.0;     // Reduccion de velocidad

// Variables auxiliares
int inicio = 0;
int i = 0;
int m = 0;

void setup() {
    // Inicializacion de los puertos de comunicacion serial
    Serial.begin(115200);
    Serial3.begin(115200);

    // Configuracion de los pines
    pinMode(Pin_DIR,OUTPUT) ; pinMode(Pin_BRK,OUTPUT) ;
    digitalWrite(Pin_BRK,HIGH) ;
    pinMode(RT,OUTPUT) ;

    // Inicializacion de la señal PWM
    pwm_pin42.start(PWM_PERIOD_PIN_42,PWM_DUTY_PIN_42);

    // Inicializacion del sensor
    init_MPU6050(33,31) ;
}

// Prototipo de funcion para cambiar el ciclo de trabajo
#define change_duty_definition \
template<typename pwm_type> void change_duty( \
    pwm_type& pwm_obj, \
    uint32_t pwm_duty, \
    uint32_t pwm_period \
) \
{ \
    uint32_t duty=pwm_duty; \
    pwm_obj.set_duty(duty); \
}
change_duty_definition;
```



```
////////////////////////////////////  
void loop() {  
  muestra_MPU6050();  
  Parametros();  
  
  if (inicio == 1) {  
    if (abs(int(Xf)) > 12) {  
      digitalWrite(Pin_BRK,LOW); // Se activa el freno  
      change_duty(pwm_pin42,5000,PWM_PERIOD_PIN_42); // Se envia señal de paro al  
motor  
      u = 0; acc_int = 0; acc_der = 0; err_ant = 0; int_vel =0; ref = 0;  
    }  
    else {  
  
      digitalWrite(Pin_BRK, HIGH); // Se desactiva el freno  
  
      err = (ref - Xf); // Se calcula el error  
      acc_int += constrain(err * 0.01,-U_MAX/1.5,U_MAX/1.5); // Se calcula la  
accion integral  
  
      // Reduccion de velocidad del motor en posicion vertical  
      if (abs(int(Xf)) < 3 ) {  
        int_vel += Xf;  
      } else int_vel = 0;  
  
      acc_der = (err - err_ant) / 0.010 ; // Se calcula la accion derivativa  
  
      u = constrain(kp * err + ki * acc_int + kd * acc_der +kr * -int_vel,-  
U_MAX,U_MAX); // Se calcula la accion de control  
      change_duty(pwm_pin42,map(abs(int(u)), 0, U_MAX, 5000,  
0),PWM_PERIOD_PIN_42); // Se cambia el ciclo de trabajo del motor  
  
      if (int(u) <= 0) digitalWrite(Pin_DIR, LOW); // Dirección CCW Izquierda  
      else digitalWrite(Pin_DIR, HIGH); // Dirección CW  
  
      err_ant = err; // Se almacena el error  
  
      if (m == 1) Medidas1();  
      else Medidas();  
    }  
  }  
}
```



```
else {
    u = 0; acc_int = 0; acc_der = 0; err_ant = 0; int_vel = 0; ref = 0;
    digitalWrite(Pin_BRK,LOW); // Se activa el freno
    change_duty(pwm_pin42,5000,PWM_PERIOD_PIN_42); // Se envia señal de paro al
motor

}

// Verficacion del tiempo de muestreo
t2=millis() ;
if (t2-t1>T) {digitalWrite(RT,HIGH) ;}
else {digitalWrite(RT,LOW) ;}
while (t2-t1<T) {t2=millis() ;}
t1=millis() ;

}

// Funciones auxiliares

void muestra_MPU6050() {
    int16_t ax, ay, az ;
    int16_t gx, gy, gz ;
    float Ac1, Ac2 ;
    float Gy1, Gy2 ;

    mpu.getMotion6(&ax,&ay,&az,&gx,&gy,&gz) ;

    if (ax != 0 && ay != 0 && az != 0) {
        Ac1=atan((ay/A_R)/sqrt(pow((ax/A_R),2)+pow((az/A_R),2)))*RAD_TO_DEG ;
        Gy1=gx/G_R;
        X=0.965*(X+Gy1*0.01)+0.035*Ac1;
        Xf = X-Xi+64.75;
    }
}
```



```
void init_MPU6050(int vcc,int gnd) {
  Serial.println("Reset MPU6050") ;
  pinMode(gnd,OUTPUT) ; pinMode(vcc,OUTPUT) ;
  digitalWrite(gnd,LOW) ; digitalWrite(vcc,LOW) ;
  delay(1000) ; digitalWrite(vcc,HIGH) ; delay(1000) ;

  Wire.begin() ;
  Serial.println("Inicializa I2C...") ;
  mpu.initialize() ;
  Serial.println("Prueba de conexion...") ;
  Serial.println(mpu.testConnection() ? "MPU6050 OK" : "MPU6050 XX") ;

  mpu.setXGyroOffset(127) ;

  for (int jj=0 ; jj<9 ; jj++) {
    Serial.print("0") ;
    for (int ii=0 ; ii<99 ; ii++) {
      muestra_MPU6050() ; delay(10) ;
    }
  }
  int cc=0 ;
  for (int jj=0 ; jj<9 ; jj++) {
    Serial.print("0") ;
    for (int ii=0 ; ii<99 ; ii++) {
      muestra_MPU6050() ; cc++ ;
      Xi=Xi+X ;
      delay(10) ;
    }
  }
  Xi=Xi/cc ;
  Serial.println("") ;
}

int Parametros() {
  if (!Serial3.available()) return 0;
  delay(2);
  char cmd = Serial3.read();
  if (!Serial3.available()) return 0;
  char cmd2 = Serial3.read();
  Serial.flush();
  switch (cmd) {
    case 's':
      if (cmd2 == '0') inicio = 0;
  }
}
```



```
        if (cmd2 == '1')    inicio = 1;
        break;
    case 'p':
        if (cmd2 == '+')    kp += 1;
        if (cmd2 == '-')    kp -= 1;
        Valores();
        break;
    case 'i':
        if (cmd2 == '+')    ki += 0.05;
        if (cmd2 == '-')    ki -= 0.05;
        Valores();
        break;
    case 'd':
        if (cmd2 == '+')    kd += 0.001;
        if (cmd2 == '-')    kd -= 0.001;
        Valores();
        break;
    case 'r':
        if (cmd2 == '+')    kr += 0.005;
        if (cmd2 == '-')    kr -= 0.005;
        Valores();
        break;
    case 'm':
        if (cmd2 == '1')    m = 1;
        if (cmd2 == '2')    m = 0;
        Valores();
        break;
    }
}

void Valores() {
    Serial.print(" kp: "); Serial.print(kp);
    Serial.print(" ki: "); Serial.print(ki);
    Serial.print(" kd: "); Serial.print(kd, 4);
    Serial.print(" kr: "); Serial.println(kr, 5);
}

void Medidas() {
    Serial.print(" ref: "); Serial.print(ref);
    Serial.print(" pos: "); Serial.print(Xf);
    Serial.print(" err: "); Serial.print(err);
    Serial.print("  u: "); Serial.println(u);
}
```



```
void Medidas1() {  
    Serial.print(" ref: "); Serial.print(ref);  
    Serial.print(" p: "); Serial.print(kp * err);  
    Serial.print(" i: "); Serial.print(ki*acc_int);  
    Serial.print(" d: "); Serial.print(kd*acc_der,4);  
    Serial.print(" v: "); Serial.print(kr*-int_vel);  
    Serial.print(" u: "); Serial.println(u);  
  
}
```