



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA



Escuela Técnica Superior de Ingeniería del Diseño



TRABAJO FIN DE MÁSTER  
INGENIERÍA MECATRÓNICA

# DISEÑO, IMPLEMENTACION Y CONTROL DE PENDULO CON VOLANTE DE INERCIA

**AUTOR:**

Fabio Andres Rocchi

**DIRECTOR:**

Vicente Casanova Calvo

Agosto 2018

*“El estudioso es el que lleva a los demás a lo que él ha comprendido: la verdad”*  
Santo Tomas de Aquino

## **AGRADECIMIENTO**

A Dios, Padre,Hijo y Espiritu Santo, a Maria y a Jose.

A Mis Padres, que me han dado todo lo que pudieron dentro de su alcance, y lo mas valioso la vida,y las pasion para salir al mundo a buscar lo que me motiva.

A mis hermanos Gabriela y Ariel, los cuales siempre apoyaron mis aventuras y desventuras.

Y a todos los cientificos, escritores, medicos, historiadores, filosofos, santos etc, que me motivaron y despertaron en mi la sed de curiosidad que motiva mi vida para tener una vida con proposito, en busca de la verdad.

Y por ultimo y no menos importante a mi tutor Vicente, por su ayuda constante su confianza y su paciencia, pero por sobretodo a arriesgarse a creer en mi, a mis amigos Danilo y Henry compañeros, que me ayudaron siempre en este Master, el cual me apasiono desde el primer día, y al apoyo de Ramon y Oscar por su guía y su sapienza.

*“Yo no sé quién me ha puesto en este mundo, ni qué es el mundo, ni qué soy yo; me encuentro en una terrible ignorancia de todas estas cosas; no sé lo que es mi cuerpo, ni mis sentidos, ni mi alma, ni siquiera esta parte de mi yo que piensa lo que digo, que reflexiona sobre todo y sobre sí misma y que no se conoce a sí misma mejor que el resto. Veo estos terribles espacios del universo que me envuelven, y me encuentro atado a un rincón de esta vasta extensión, sin que sepa por qué estoy situado en este lugar y no en otro, ni por qué este poco de tiempo que me ha sido concedido para vivir me ha sido asignado en este momento y no en otro de toda la eternidad que me he precedido y de toda la que me sigue. No veo más que infinitudes por todas partes que me envuelven como a un átomo y como a una sombra que no dura más que un instante sin retorno. Todo lo que yo sé es que debo morir pronto; pero lo que más ignoro es, precisamente, esa muerte que no sabré evitar. Como no sé de dónde vengo, tampoco sé a dónde voy; y solo sé que al salir de este mundo caeré para siempre o en la nada, o en las manos de Dios irritado, sin saber a cuál de estas dos condiciones debo estar eternamente sujeto. He aquí mi estado, lleno de debilidad e incertidumbre”*

Fragmento de Pensamientos, Blaise Pascal

## ÍNDICE

<b>1. INTRODUCCION .....</b>	<b>1</b>
<b>1.1. Objetivo especifico del Trabajo.....</b>	<b>2</b>
<b>1.2. Reseña de los sistemas mecanico subactuados; principalmente pendulos invertidos y aplicaciones. ....</b>	<b>3</b>
<b>1.3. Variante de Pendulos Invertidos .....</b>	<b>7</b>
<b>1.4. Etapas de Implementacion del prototipo.....</b>	<b>9</b>
<b>2. FUNDAMENTACIÓN TEÓRICA.....</b>	<b>10</b>
<b>2.1. Péndulo invertido con volante de inercia o rueda de reacción.[3] .....</b>	<b>11</b>
2.1.1. Modelo matemático [3] .....	12
<b>3. DISEÑO CAD Y SIMULACIONES .....</b>	<b>19</b>
<b>3.1. Diseño del prototipo en SolidWorks y Analisis de Comportamiento de péndulo Libre.19</b>	
<b>3.2. Simulación Mediante Multibody Simulink Simscape .....</b>	<b>22</b>
3.2.1. Control del angulo de posición: .....	26
<b>4. IMPLEMENTACION DEL PENDULO CON VOLANTE DE INERCIA.....</b>	<b>32</b>
<b>4.1. Componentes .....</b>	<b>32</b>
4.1.1. Piezas de DM cortadas con laser.....	33
4.1.2. Láminas de metacrilato y Barra de redondo de Metacrilato .....	35
4.1.3. Cubo Anclaje para el modelo con varrilla de metacrilato.....	36
4.1.4. Base de Pendulo Invertido con Volante inercial. ....	37
4.1.5. Adaptador de eje en Alumino.....	40

<b>4.2. Componentes Electrónicos</b> .....	40
4.2.1. Driver de potencia L298N.....	51
<b>5. IMPLEMENTACIÓN DEL PENDULO CON VOLANTE DE INERCIA.CODIGO DE PROGRAMACION.</b> .....	55
<b>5.1. Programa Utilizado.</b> .....	57
<b>5.2. Graficas y resultados obtenidos.</b> .....	63
<b>5.3 Ajuste de parámetros del regulador</b> .....	65
<b>6.CONCLUSIONES Y PROPUESTAS A FUTURO.</b> .....	66
<b>7.BIBLIOGRAFIA</b> .....	66
<b>1. ANEXOS</b> .....	¡Error! Marcador no definido.
<b>1.4. Planos</b> .....	¡Error! Marcador no definido.
<b>1.5. Valoración económica</b> .....	¡Error! Marcador no definido.
<b>1.6. Programas</b> .....	¡Error! Marcador no definido.
1.6.1. Arduino Nano.....	¡Error! Marcador no definido.
1.6.2. STM32F446RE .....	¡Error! Marcador no definido.

## ÍNDICE DE FIGURAS

<b>Figura 1:</b> Pendulo con Volante de Inercia/Rueda de Reacción.....	2
<b>Figura 2:</b> Robot Asimo Fabricado por HONDA Japon.....	4
<b>Figura 3:</b> Mantenimiento de direccion de Cohete .....	5
<b>Figura 4:</b> Segway.....	6
<b>Figura 5:</b> Pendulo invertido con control sobre carro de movimiento lineal .....	7
<b>Figura 13:</b> Diferentes Vistas de Detalle del Prototipo a Simular. ....	19
<b>Figura 14:</b> Caracteristicas Geometricas y Físicas .....	20
<b>Figura 15:</b> Grafica del momento necesario según la posición.....	21
<b>Figura 16:</b> Energia Cinetica y potencial del sistema .....	21
<b>Figura 17:</b> Analisis del movimiento armonico del péndulo libre del sistema. ....	22
<b>Figura 18:</b> Sistema exportado y ensamblado en simulink Simscape. ....	23
<b>Figura 19:</b> Pendulo con Volante de Inercia/Rueda de Reacción.....	24
<b>Figura 20:</b> Modelo diseñado en Simulink - Multibody .....	25
<b>Figura 21:</b> Bloque "Base" ampliado con sus transformaciones de orientación para poder ser utilizado correctamente en el espacio de la simulación. ....	25
<b>Figura 22:</b> Modelo diseñado en Simulink – Multibody correspondiente a la base donde se encuentra el encoder para medir el angulo del péndulo. ....	26
<b>Figura 23:</b> Modelo diseñado en Simulink – Multibody de la vinculación del volante y motor dc con el brazo del péndulo.....	26
<b>Figura 24:</b> Regulador de posicion sobre modelo de Multibody .....	27
<b>Figura 25:</b> Comportamiento del péndulo simulado .....	28
<b>Figura 26:</b> Error de angulo de péndulo vs referencia a seguir. ....	28
<b>Figura 27:</b> Error de angulo de péndulo vs referencia a seguir (Fig Detallada).....	29
<b>Figura 27:</b> Variacion dela velocidad Absoluta (rojo) RPM vs Variacion de la aceleracion.....	30
<b>Figura 28:</b> Comportamiento de la trayectoria del péndulo vs perturbaciones. ....	30
<b>Figura 29:</b> Comportamiento de la velocidad y aceleración del pensulo vs las perturbaciones .....	31
<b>Figura 30:</b> Brazo del Pendulo .....	33
<b>Figura 31:</b> Volante de inercia del pendulo .....	34
<b>Figura 32:</b> Laminas de metacrilato y Redondo en 16mm.....	35

<b>Figura 33: Cubo de Anclaje</b> .....	36
<b>Figura 34: Ensamblaje de la Base Quanser con Encoder en Base de Metacrilato.</b> .....	37
<b>Figura 35: Ejemplo de Pendulos Invertidos de la firma Quanser que utilizan la base con encoder que fue reutilizada para nuestro modelo.</b> .....	38
<b>Figura 36: Base de Pendulo comercial reutilizada</b> .....	39
<b>Figura 37: Adaptadores de eje y características</b> .....	40
<b>Figura 38: figura de un disco de encoder incremental</b> .....	41
<b>Figura 39: Lector Optico Caracteristicas</b> .....	42
<b>Figura 40: Empaque Encoder</b> .....	43
<b>Figura 41: Motor Maxon Caracteristicas</b> .....	44
<b>Figura 42: Motor Maxon Caracteristicas</b> .....	45
<b>Figura 43: Motor Maxon Caracteristicas</b> .....	46
<b>Figura 44: Motor Maxon Caracteristicas</b> .....	47
<b>Figura 45: Motor Maxon Caracteristicas</b> .....	48
<b>Figura 46: Arduino Due Imagen Y Caracteristicas</b> .....	49
<b>Figura 47: Arduino Due disposición de pines y funciones</b> .....	50
<b>Figura 48: Imagen Driver o etapa de potencia L298N</b> .....	51
<b>Figura 49: Caracteristicas técnicas de la etapa de potencia L298N</b> .....	52
<b>Figura 50: Caracteristicas técnicas de la etapa de potencia L298N</b> .....	53
<b>Figura 52: Caracteristicas técnicas de la etapa de potencia L298N</b> .....	54
<b>Figura 53: Esquemático de ejemplo de un coneccionado entre arduino, etapa de potencia y motor.</b> .....	55
<b>Figura 54: Diagrama de ineraccion de nuestro sistema</b> .....	55
<b>Figura 55: Etapa de comienzo de construcción de prototipo físico.</b> .....	56
<b>Figura 56: Prototipo en funcionamiento</b> .....	56
<b>Figura 57: Prototipo en calibración y pruebas.</b> .....	57
<b>Figura 58: Comportamiento del Levantamiento hasta referencia 0° y posteriormente seguimiento de senoidal.</b> .....	64
<b>Figura 59: Seguimiento de referencia senoidal variable hasta perder la referencia.</b> .....	65

## ÍNDICE DE TABLAS

<b>Tabla 1:</b> Descripción de variables y constantes del actuador-motor .....	12
<b>Tabla 2:</b> Descripción de variables y constantes del péndulo .....	12



## 1. INTRODUCCION

La motivacion para la realizacion de este trabajo, se debio principalmente a la implementación de un sistema Mecatrónico, que nucleè diferentes asignaturas aprendidas en el master y poder llevar a cabo tanto la simulacion, como la materializacion del prototipo.

Para el desarrollo, diseño, e implementacion nos hemos basado en material aportado por materias como son:

1. Control Automatico.
2. Dinamica de los sistemas Mecánicos.
3. Sistemas embebidos.
4. Electronica de Potencia
5. Tecnicas Avanzadas de control
6. Control Aplicado a sistemas mecatronicos.
7. Sistemas de Medicion y Actuacion.
8. Electronica e Instrumentacion
9. Maquinas electricas.
10. Modelado y Control de Maquinas Electricas.

Tambien, conocimientos como modelado, simulación en 3d y mecanizado de piezas.

Con lo cual se busco satisfacer el objetivo del master.

### ¿Cuál es el objetivo del Máster?

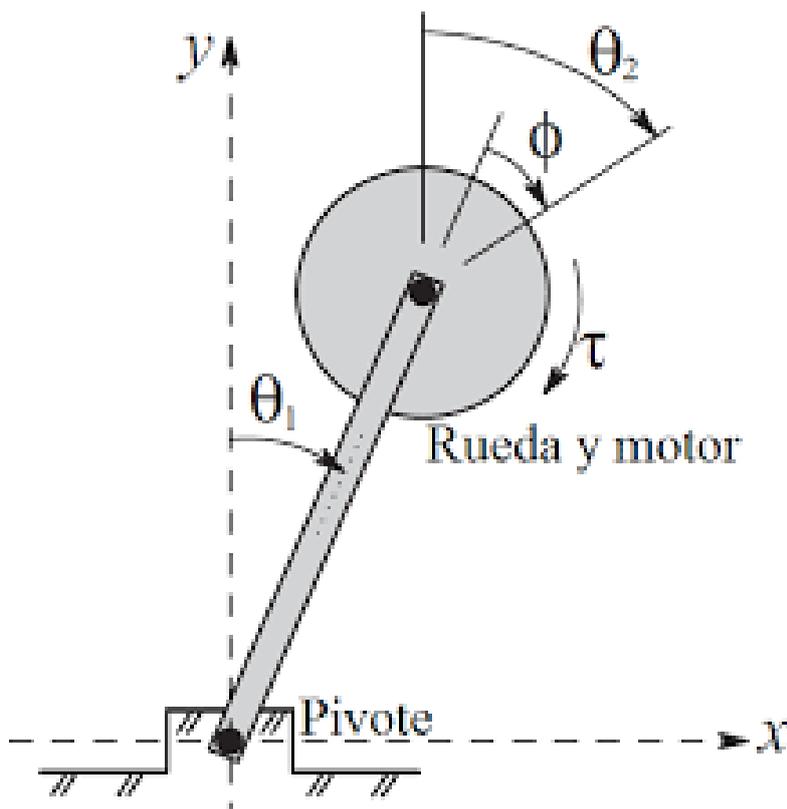
El objetivo fundamental del Máster Universitario de Ingeniería Mecatrónica es ofrecer a sus alumnos una formación integral en Ingeniería Mecatrónica, incluyendo el análisis, diseño e implementación de sistemas mecatrónicos en entornos industriales. Que dichos alumnos adquieran una capacidad de desarrollo, e incluso investigación, en Ingeniería Mecatrónica y en las metodologías y tecnologías que permitirán la creación de productos y sistemas de ingeniería definidos por la combinación sinérgica de tecnologías mecánica, electrónica, eléctrica, control e informática.

Extracto del díptico informativo del Master en Mecatrónica.

### 1.1. Objetivo específico del Trabajo.

El objetivo específico del siguiente trabajo es, diseñar e implementar un péndulo invertido con volante de inercia.

El péndulo con volante de inercia, es un sistema mediante en el cual debemos conocer su posición angular respecto a la vertical  $0^\circ$  la cual es sensada, en nuestro caso mediante un encoder incremental óptico, dicha posición se utiliza para que el sistema de control, programado en un sistema embebido, la reciba y realice las acciones necesarias de acelerar o desacelerar en uno u otro sentido al volante de inercia para generar el par suficiente y mantenerlo en equilibrio. De esta manera y ante pequeñas perturbaciones que desestabilicen al péndulo, el sistema corrija y mantenga su posición como también si así lo deseamos poder realizar un seguimiento de una referencia senoidal de movimiento del volante.



**Figura 1:** Pendulo con Volante de Inercia/Rueda de Reacción.

## **1.2. Reseña de los sistemas mecanico subactuados; principalmente pendulos invertidos y aplicaciones.**

Un pendulo invertido es un sistema mecanico en el cual una barra pivotea con respecto a un punto principalmente situado en su extremo fijado en el espacio.[Ver fig 1]

La dinamica del movimiento se mide en grados de libertad, siendo en estos un grado de libertad. Y cuyo movimiento puede ser controlado o no, siendo los controlados los sistemas en los cuales se basa los mecanismos de estudio.

Dentro de lo que refiere el control de estos sistemas y dependiendo de los grados de libertad, tenemos los sistemas mecanicos actuados y sistemas mecanicos subactuados.

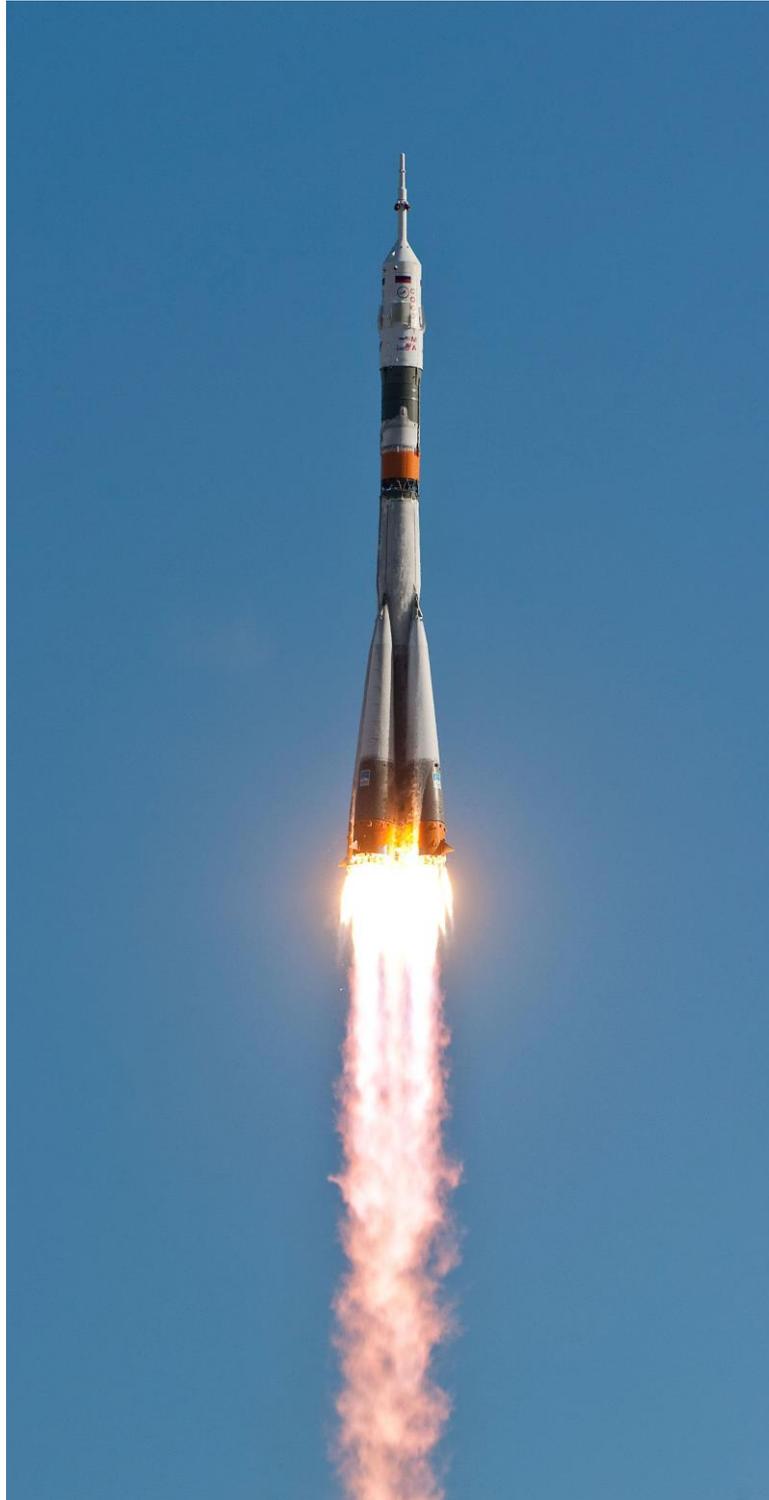
El beneficio de los subactuados sobre los actuados es que en estos se produce un ahorro de energia debido a que posee menos actuadores que grados de libertad posee el sistema.

Sobre estos ultimos los sistemas de control se han abocado a investigar, puesto que tienen aplicación directa en Robotica, Industria Aeroespacial, y vehiculos tanto submarinos como terrestres[1].A continuacion describiremos algunos ejemplo de utilizacion.



**Figura 2:** Robot Asimo Fabricado por HONDA Japon

En estos tipos de robots bipedos caminantes, la pierna de apoyo se modela como un pendulo invertido, mientras que la pierna suspendida se modela como un pendulo simple.



**Figura 3:** Mantenimiento de direccion de Cohete

En el despegue y ascenso de un cohete, se requiere un control activo para mantenerlo en la posición vertical, en estos casos el ángulo de ascenso/inclinación del cohete es controlado por el ángulo de la fuerza de empuje colocada en la base del cohete.



**Figura 4:** Segway

Pendulo invertido que su angulo es sentido por giroscopos en la base del mismo, y que de esta manera actuan sobre la traccion de las ruedas para mantener en equilibrio a las personas que transporta.

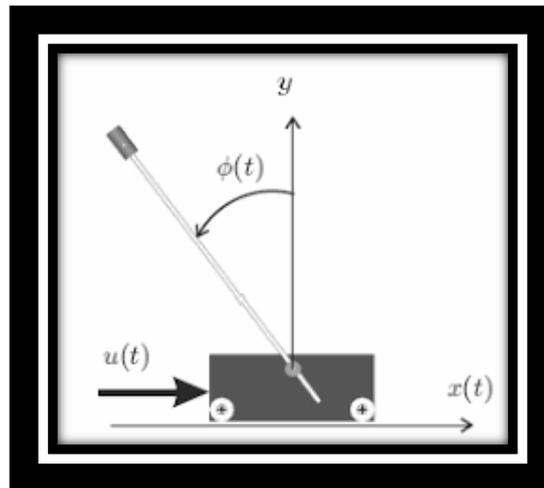
### 1.3. Variante de Pendulos Invertidos

Los pendulos invertidos, pertenecen a los sistemas subactuados, y clasificando los mismos por el tipo de mecanismo que conforma su movimiento en [2]:

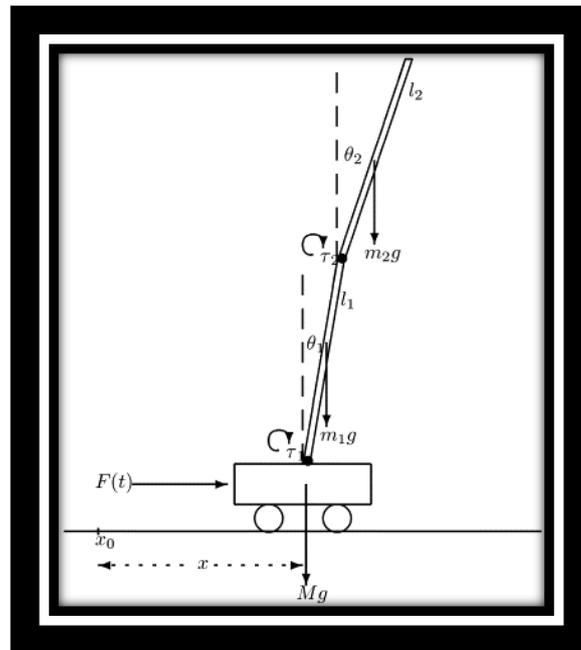
- Pendulo con movimiento lineal sobre un carro.
- Pendulo con movimiento rotacional o giratorio.

A continuacion podemos ver diferentes versiones de pendulos invertidos.

Pendulo con movimiento lineal sobre un carro:



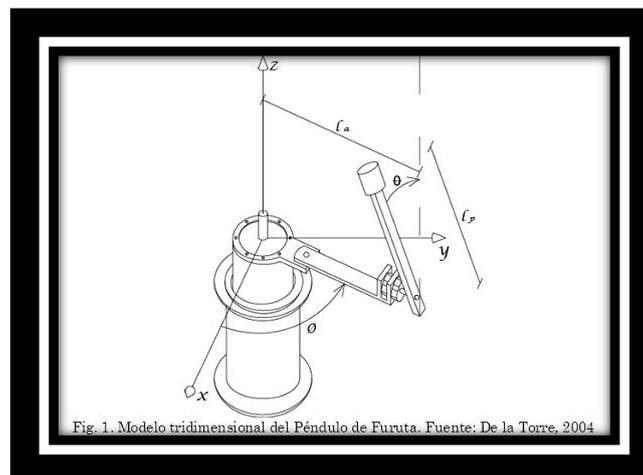
**Figura 5:** Pendulo invertido con control sobre carro de movimiento lineal



**Figura 6:** Doble Pendulo invertido con control sobre carro de movimiento lineal.

Sistema en el cual, mediante la aplicación de una fuerza que provoque las traslacion del carro, controlara, la posicion vertical de equilibrio del pendulo invertido. Tambien pueden construirse, dobles y triples.

Pendolo de Furuta:

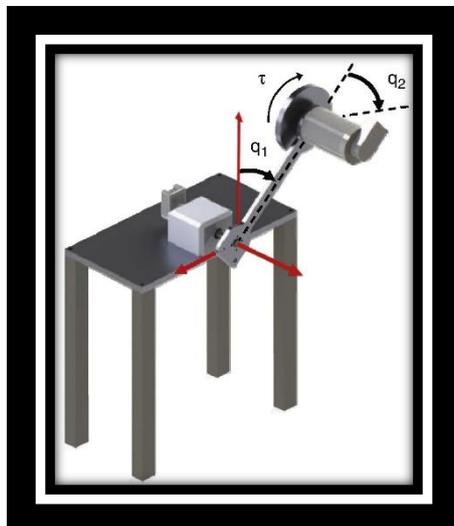


**Figura 6:** Pendulo de Furuta

EL pendulo de Furuta, llamado asi en honor a su creador, el profesor K.Furuta. Consta de dos eslabones,y dos articulaciones de rotacion, el primer eslabon llamado brazo,tiene un movimiento angular sobre el plano horizontal, y esta accionado por un motor, y el segundo

eslabon se llama posee un movimiento rotacional en el plano vertical, y se encuentra conectado al extremo del brazo, pivotando libremente con respecto a el.

Pendulo con rueda inercial:



**Figura 7 :** Pendulo con volante de inercia.

El péndulo con volante de inercia, como se ve en la figura consiste en un péndulo convencional en cuyo extremo libre se encuentra un volante de inercia, que en definitiva es un disco al que se le imprimen aceleraciones angulares mediante un motor de cc, y provoca el movimiento de la varilla. Esta clase de péndulo es el objeto de nuestro estudio y veremos en profundidad la física de su comportamiento en el capítulo 2.

#### **1.4. Etapas de Implementación del prototipo.**

A continuación enunciaremos las etapas que hemos considerado para realizar el trabajo.

- Cálculo de aproximación de las masas necesarias, dimensiones geométricas, y características eléctricas del motor de cc a seleccionar.
- Modelar el prototipo en software 3d, en nuestro caso hemos utilizado SolidWorks, definir materiales y pesos.
- Construir el modelo de control en Matlab-Simulink.
- Importar las piezas y componentes a Simulink y utilizando Simscape.
- Comprobar y validar funcionamiento de la simulación y sintonizar el sistema.
- Selección de componentes, construir prototipo y programar control en placa embebida.
- Sintonizar parámetros del prototipo.

- Extraer graficas y resultados para su analisis.

El control del prototipo debido a que la velocidad del lazo de contro sera de 20 ms, lo implementaremos en una placa embebida (en nuestro caso Arduino Due). Mientras que las ordenes de referencia, tanto de referencia cero, como senoidal las realizaremos desde el computador u dispositivo bluetooth.

## 2. FUNDAMENTACIÓN TEÓRICA

EL objetivo de de implementar un control de un pendulo invertido con volante de inercia o rueda de reacción es mantener el mismo en equilibrio vertical, formando un angulo de 0 grados, sin este mismo al ser esta posicion inestable, el mismo se caería, por efecto de la gravedad.

Sin embargo si implementamos un control adecuado, generaremos en la rueda de mismo o volante de inercia, aceleraciones y desaceleraciones en ambos sentidos, que a su vez crearan fuerzas en proporción con la masa del volante, que engendrarán, aplicadas a una distancia “l” correspondiente al brazo de longitud del volante, momentos de reacción que se opondrán a los momentos que forman la componente del peso tangencial a la trayectoria y el brazo del péndulo.

De esta forma el control equilibra al sistema, y reacciona en función del ángulo de desequilibrio, a mayor ángulo mayor aceleración, para aplicar de esta manera un momento mayor para corregir este desequilibrio.

Para poder implementar la resolución del mismo utilizaremos documentos en el cual explica la dinámica de un péndulo invertido con volante de inercia, para poder entender su funcionamiento matemático y dinamico, posteriormente modelaremos en NX y exportaremos a Simulink, para

poder experimentar con el control dinámico del sistema y obtener por aproximación las constantes de nuestro control PID.

### 2.1. Péndulo invertido con volante de inercia o rueda de reacción.[3]

Para este documento nos hemos basado en el Paper titulado “Control de un péndulo invertido” por Cristian Guajardo Yevenes, Alumno perteneciente al Departamento de Electrónica, Universidad Técnica Federico Santa María, 05 de mayo de 2004.

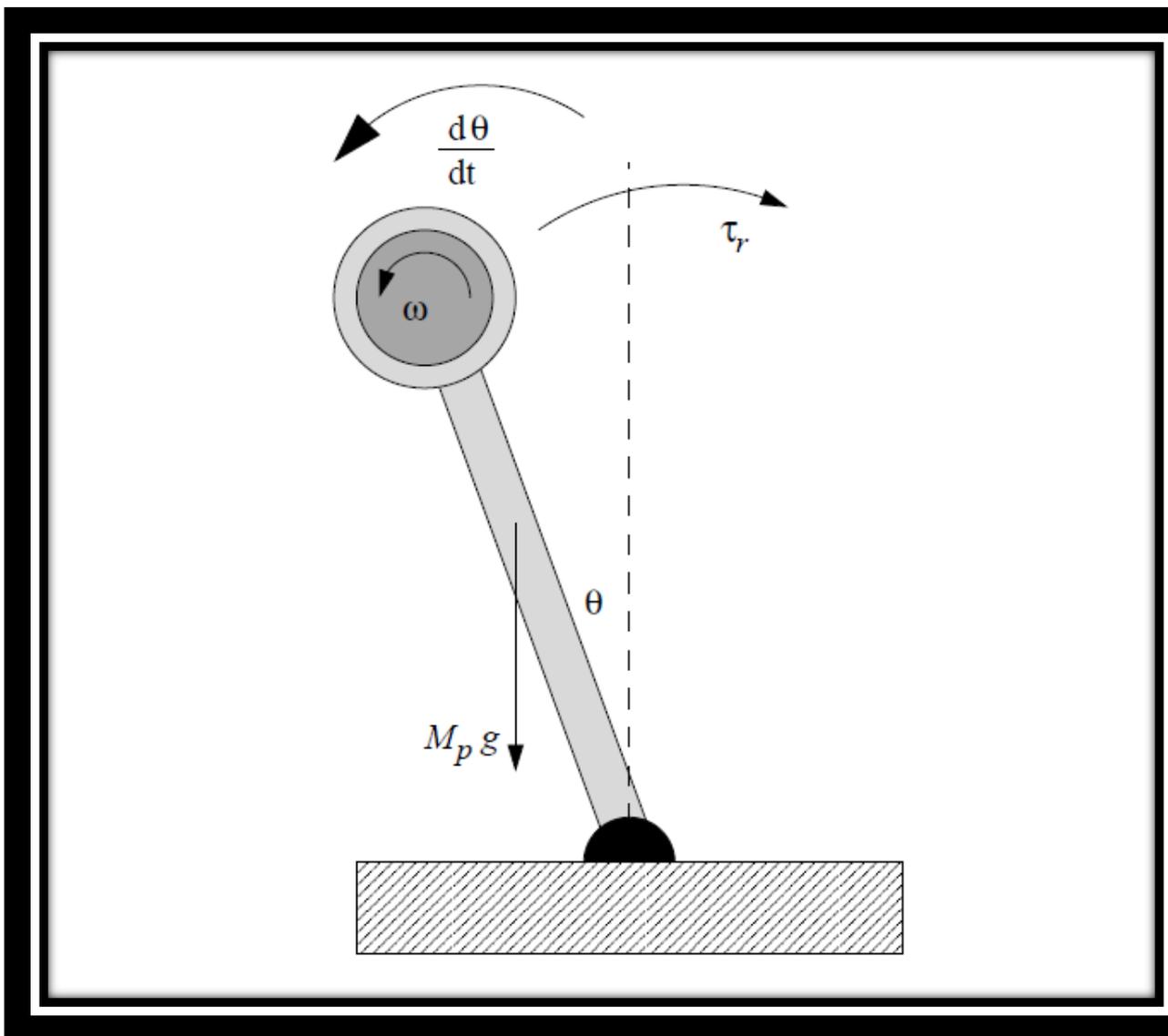


Figura 8: Péndulo Invertido sobre carro

El problema del pendulo invertido con volante de inercia es uno de los mas comunes utilizado en los problemas de control.

### 2.1.1. Modelo matemático [3]

En el siguiente modelo matemático se considera que la dinámica del motor es rápida, y que su rozamiento. Producto de su roce viscoso es despreciable o sea  $L_a = 0$  y  $D = 0$ .

A continuación se listan las tablas de parametros.

Parámetros	Detalles	Unidades
$w(t)$	Velocidad angular del volante de inercia.	$rad/s$
$T_e(t)$	Torque lectrico producido por el motor	$Nm$
$I_a(t)$	Corriente de Armadura	$A$
$v_a(t)$	Voltage de Armadura	$V$
$u(t)$	Voltage de actuación proveniente del computador	$V$
$I_p$	Velocidad angular del péndulo	$Kgm^2$
$D$	Coefficiente de roce viscoso	$Nms/rad$
$M_f$	Inductancia del Motor	$Nm/A$
$R_a$	Resistencia de armadura	$\Omega$
$L_a$	Inductancia de Armadura	$H$
$k_a$	Ganancia del amplificador de potencia	$V/V$
$\delta$	Voltage de offset de amplificador del sistema	$V$

**Tabla 1:** Descripcion de variables y constantes del actuador-motor

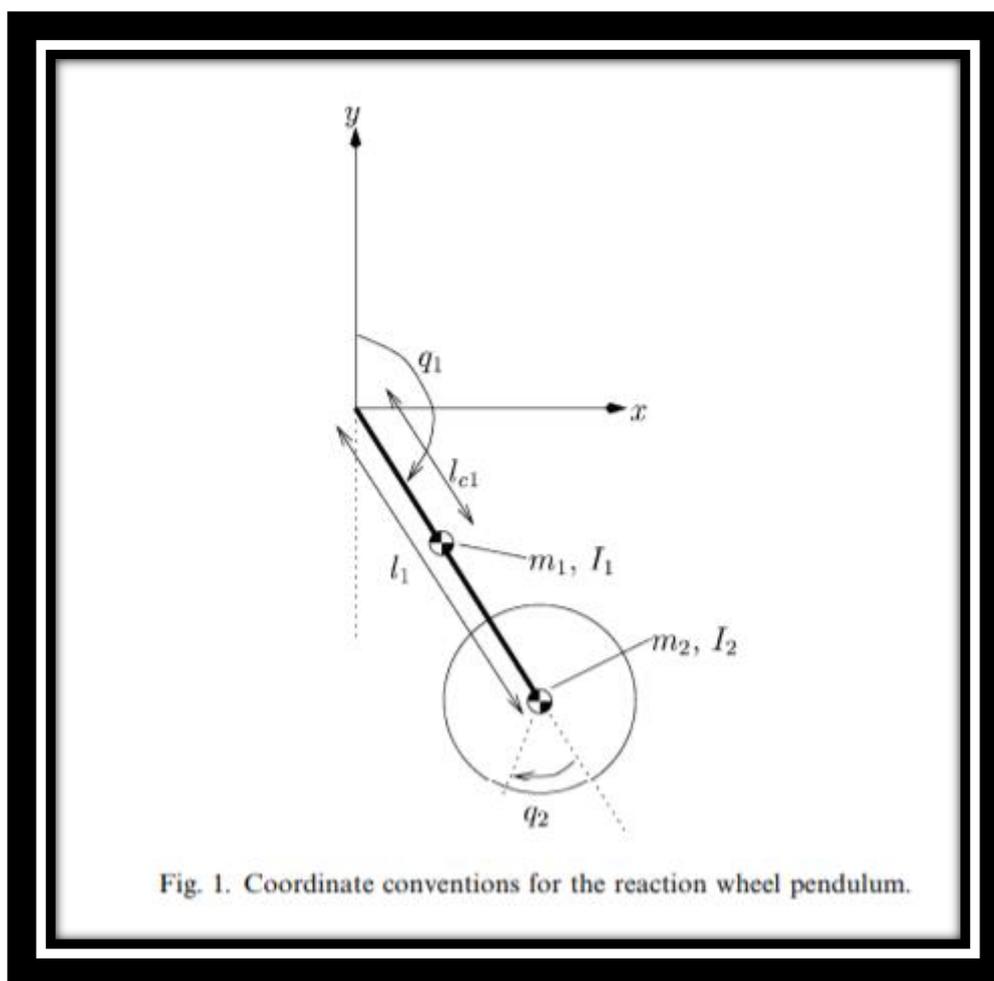
Parámetros	Detalles	Unidades
$T_r(t)$	Torque de reacción	$Kgm^2$
$W(t)$	Masa de la rueda	$rad/s$
$\theta(t)$	Angulo de la rueda	$rad$
$I_p$	Inercia del volante con respecto al eje del motor	$Kgm^2$
$I_p$	Inercia total del péndulo con respecto al pivote	$Kgm^2$
$M_p$	Masa total del pendulo	$kg$
$g$	Aceleracion de la gravedad	$m/s^2$
$L$	Largo del péndulo (pivote al centro de masa)	$m$

**Tabla 2:** Descripcion de variables y constantes del péndulo.

### MODELO DEL PÉNDULO ESTUDIO DINAMICO [3-4]

Para facilitar la derivada de las ecuaciones de la dinámica del péndulo invertido con rueda de reacción, el sistema puede modelarse como un robot de dos grados de libertad, donde el primer vínculo lo forma el péndulo con su pivote, y el segundo vínculo lo forma el disco y su eje de rotación.

Cabe destacar que por convención se toma como ángulo positivo en el sentido de las agujas del reloj.



**Figura 9: Péndulo** Invertido con rueda de reaccion

Con respecto a la dinámica del péndulo podemos decir que, sobre el péndulo actúan las fuerzas de gravedad, que se supone en el centro de masa del sistema pendular y la fuerza de torque que

reacciona desde el volante de inercia hacia el péndulo. Y es la cual nosotros controlaremos para mantener el sistema en equilibrio.

$$\tau_r(t) = I_v \frac{d\omega(t)}{dt}$$

$$I_p \frac{d^2\theta(t)}{dt^2} = M_p g L \text{sen } \theta(t) - \tau_r(t)$$

$$I_v \frac{d\omega(t)}{dt} = \tau_e(t) - D\omega(t)$$

$$\tau_e(t) = M_f i_a(t)$$

$$v_a(t) = R_a i_a(t) + L_a \frac{di_a(t)}{dt} + M_f \omega(t)$$

$$v_a(t) = k_a u(t) + \delta_a$$

Linealizando las ecuaciones del sistema se tiene

$$I_v \frac{d\Delta\omega(t)}{dt} = \Delta\tau_e(t) - D\Delta\omega(t)$$

$$\Delta\tau_e(t) = M_f \Delta i_a(t)$$

$$\Delta v_a(t) = R_a \Delta i_a(t) + L_a \frac{d\Delta i_a(t)}{dt} + M_f \Delta\omega(t)$$

$$\Delta v_a(t) = k_a \Delta u(t)$$

$$\frac{\Delta\Omega(s)}{\Delta U(s)} = \frac{M_f k_a}{I_v R_a s + M_f^2} \quad (1)$$

donde  $\Delta\Omega(s) = \mathcal{L}\{\Delta\omega(t)\}$  y  $\Delta U(s) = \mathcal{L}\{\Delta u(t)\}$ .

Con respecto a la dinámica del péndulo podemos decir que, sobre el péndulo actúan las fuerzas de gravedad, que se supone en el centro de masa del sistema pendular y la fuerza de torque que reacciona desde el volante de inercia hacia el péndulo. Y es la cual nosotros controlaremos para mantener el sistema en equilibrio.

$$\begin{aligned} \tau_r(t) &= I_v \frac{d\omega(t)}{dt} \\ I_p \frac{d^2\theta(t)}{dt^2} &= M_p g L \sin\theta(t) - \tau_r(t) \end{aligned}$$

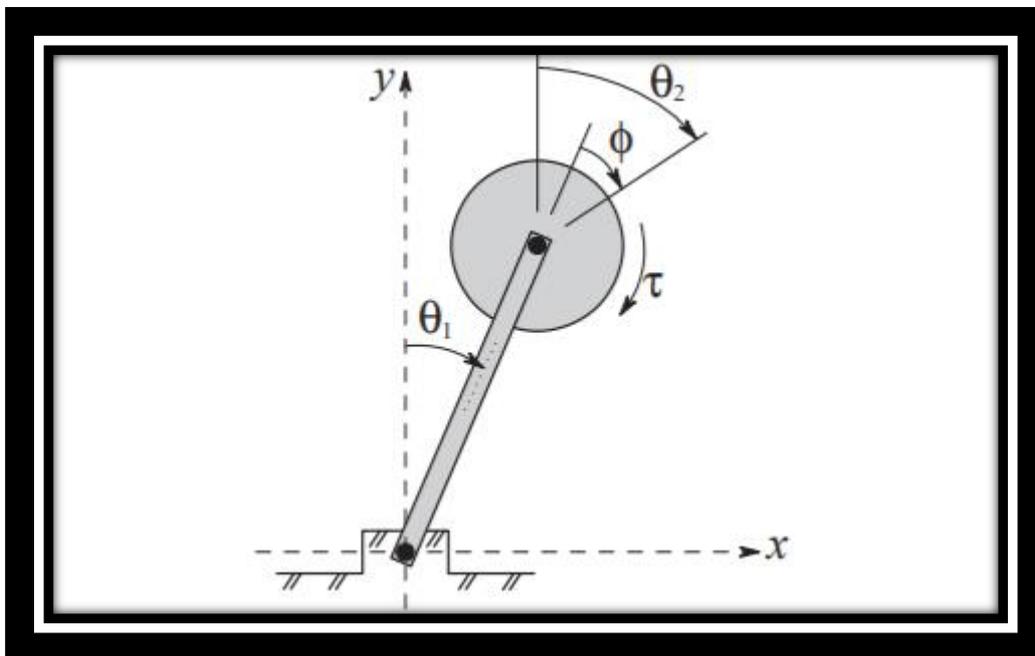
Si linealizamos las ecuaciones del péndulo obtenemos que:

$$\begin{aligned} \Delta\tau_r(t) &= I_v \frac{d\Delta\omega(t)}{dt} \\ I_p \frac{d^2\Delta\theta(t)}{dt^2} &= M_p g L \cos(\bar{\theta}) \Delta\theta(t) - \Delta\tau_r(t) \end{aligned}$$

$$\frac{\Delta\Theta(s)}{\Delta\Omega(s)} = \frac{-I_v s}{I_p s^2 - M_p g L \cos \bar{\theta}} \quad (2)$$

donde  $\Delta\Theta(s) = \mathcal{L}\{\Delta\theta(t)\}$  y  $\Delta\Omega(s) = \mathcal{L}\{\Delta\omega(t)\}$ .

Otra Alternativa es modelar el sistema mediante Lagrange.



**Figura 10: Péndulo Invertido con volante de inercia**

Observando el diagrama esquemático anterior resolvemos:

El ángulo  $\theta_1$  es el equivalente al ángulo formado entre la vertical y el brazo del péndulo,  $\theta_2$  es el ángulo del eje de la rueda con la vertical, y  $\phi$  es el ángulo relativo entre la barra y la rueda del sensor de ángulo relativo.

Las posiciones de equilibrio son  $\theta_1 = \pi * n$  siendo  $n$  un número entero, con lo cual si  $n$  es impar (el péndulo estará abajo en equilibrio estable y si  $n$  es par el péndulo estará arriba en equilibrio inestable).

El sensor de ángulo relativo mide el ángulo de la rueda con respecto al ángulo del brazo del péndulo;

$$\theta_2 = \theta_1 + \phi \quad [2.1]$$

Recordemos que el sistema y sus características están determinados por sus dimensiones, geometría, y sus masas.

La ecuación que nuclea al péndulo (barra, motor y rueda de reacción).

$$l = \frac{m_p l_p + m_r l_r}{m}$$

$m_p$  = masa del péndulo y motor

$m_r$  = masa de rueda

$m = m_p + m_r$

$J_p$  = momento de inercia para el péndulo.

$J_r$  = momento de inercia para el volante

$$J = J_p + m_p l_p^2 + m_r l_r^2 [2.2]$$

Uno de los métodos que podemos aplicar es la mecánica de Lagrange, mediante la cual podemos utilizar las funciones escalares de la energía del sistema, sin tener que recurrir a las utilización de Newton y por consecuencia funciones vectoriales.

Por lo tanto tomando para nuestro sistema las coordenadas generalizadas  $\theta_1$  y  $\theta_2$ .

La energía cinética está en función de las velocidades del péndulo y del volante de inercia.

$$E_c(\dot{\theta}_1, \dot{\theta}_2) = \frac{1}{2} J_p \dot{\theta}_1^2 + \frac{1}{2} J_r \dot{\theta}_2^2 [2.3]$$

La energía potencial de los componentes es si no son elásticos y para el ángulo de equilibrio;

( $\theta_1 = \pi$ ).

$$E_p = mgl(1 + \cos\theta_1) [2.4]$$

La energía potencial no tiene componente de  $\theta_2$  debido a que la rueda es simétrica.

La función de Lagrange (la diferencia entre energía cinética y la potencial es;

$$L = E_c - E_p$$

$$L = \frac{1}{2}J_p\dot{\theta}_1^2 + \frac{1}{2}J_r\dot{\theta}_2^2 - mgl(1 + \cos\theta_1)[2.5]$$

El torque aplicado es obtenido del motor, sobre la rueda se obtiene un torque  $\tau$  y como reacción aparece sobre la barra-  $\tau$ .

Por lo tanto las ecuaciones del movimiento del sistema según la mecánica de Lagrange son:[4]

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}_1}\right) - \frac{\partial L}{\partial \theta_1} = -\tau[2.6a]$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}_2}\right) - \frac{\partial L}{\partial \theta_2} = \tau[2.6b]$$

Entonces derivando [2.5] con respecto a las ecuaciones anteriores obtenemos;

$$J\ddot{\theta}_1 - mgl \sin\theta_1 = -\tau[2.7a]$$

$$J_r\ddot{\theta}_2 = \tau[2.7b]$$

Y sabiendo que el torque  $\tau$  del motor se obtiene en función de la corriente del motor  $\tau = ki$

Siendo k la constante de torque otorgada por el fabricante, con lo cual reemplazando obtenemos;

$$\ddot{\theta}_1 - \frac{mgl}{J} \sin\theta_1 = -\frac{k}{J}i [2.8a]$$

$$\ddot{\theta}_2 = \frac{k}{J_r}i [2.8b]$$

Y como el torque de nuestro motor se modela como proporcional a la corriente (modelo aprox para un motor de imanes permanentes), pero nuestra señal de control esta en función de la tensión, planteamos una señal de control que sea proporcional al torque.

$$u = k_u\tau = k_u ki$$

Remplazando en las ecuaciones 2.8 a y b obtenemos en función de las constantes.

$$\ddot{\theta}_1 - a \sin\theta_1 = -bu [2.9a]$$

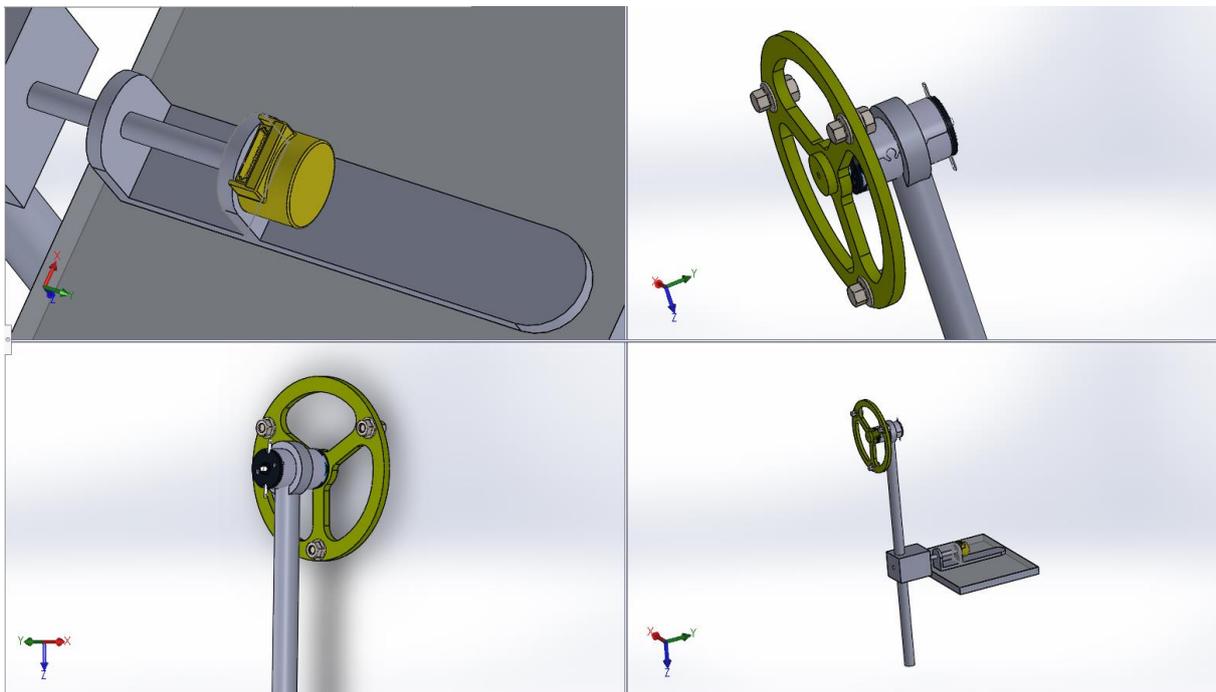
$$\ddot{\theta}_2 = cu [2.8b]$$

Donde u es la tensión de la señal de control.(su rango).

### 3. DISEÑO CAD Y SIMULACIONES

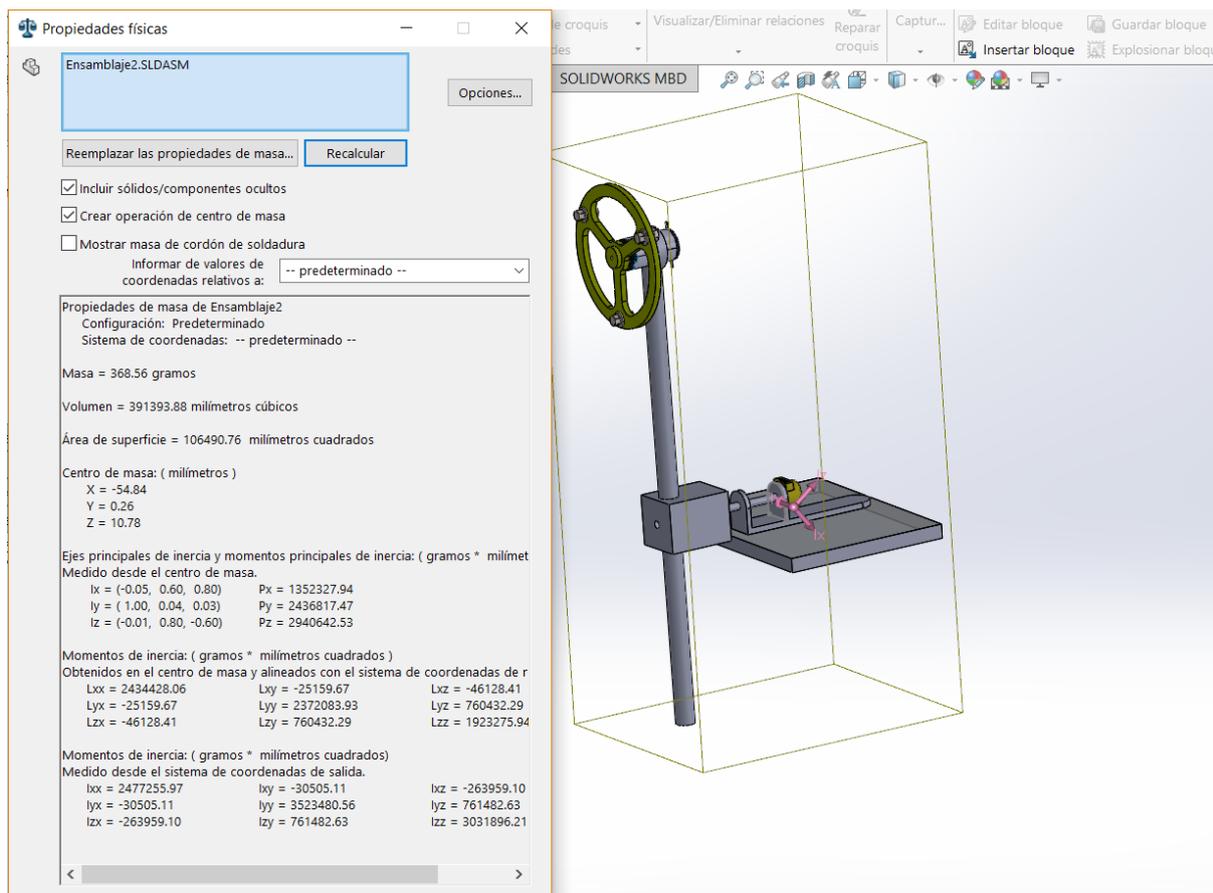
En el estudio de nuestro prototipo, se implemento una simulación virtual del comportamiento del mismo, teniendo en cuenta las formas, y los pesos del mismo para poder determinar su funcionamiento y prever las dimensiones adecuadas.

#### 3.1. Diseño del prototipo en SolidWorks y Analisis de Comportamiento de péndulo Libre.



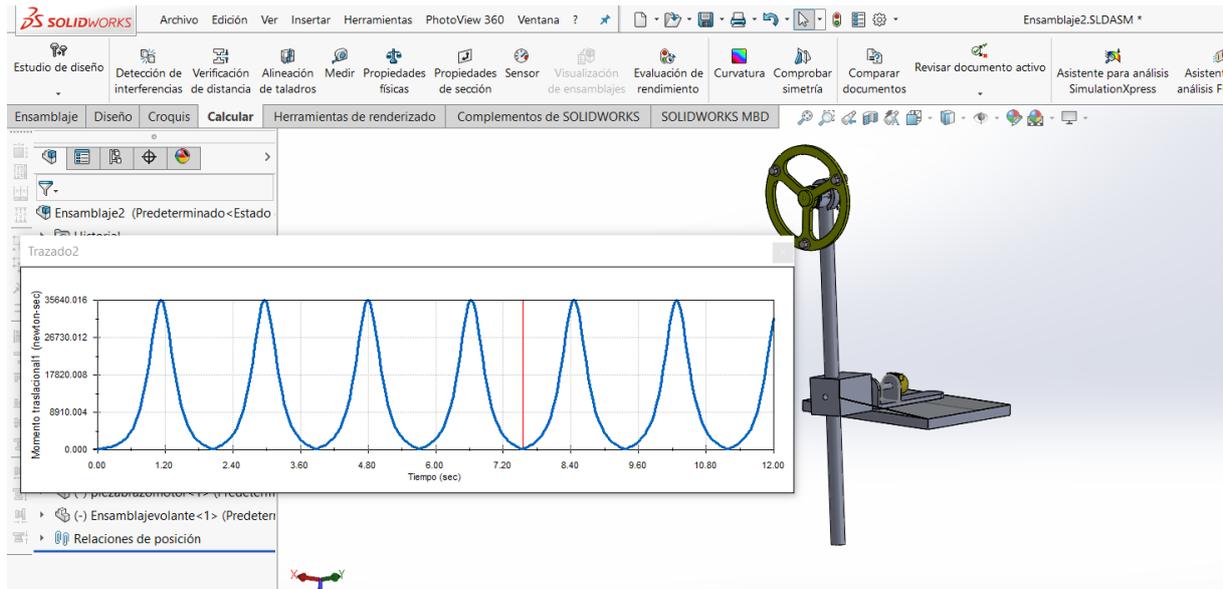
**Figura 6: Diferentes Vistas de Detalle del Prototipo a Simular.**

Para el diseño del modelo a simular empleamos el software Solidworks, en el mismo dibujamos en detalle el sistema con el fin de obtener una simulación que se asemeje a la realidad, no obstante, hubo variaciones en el peso del motor, y coeficientes de fricción, como así también al diseño del volante de inercia, se le agregaron más tuercas para aumentar su momento.



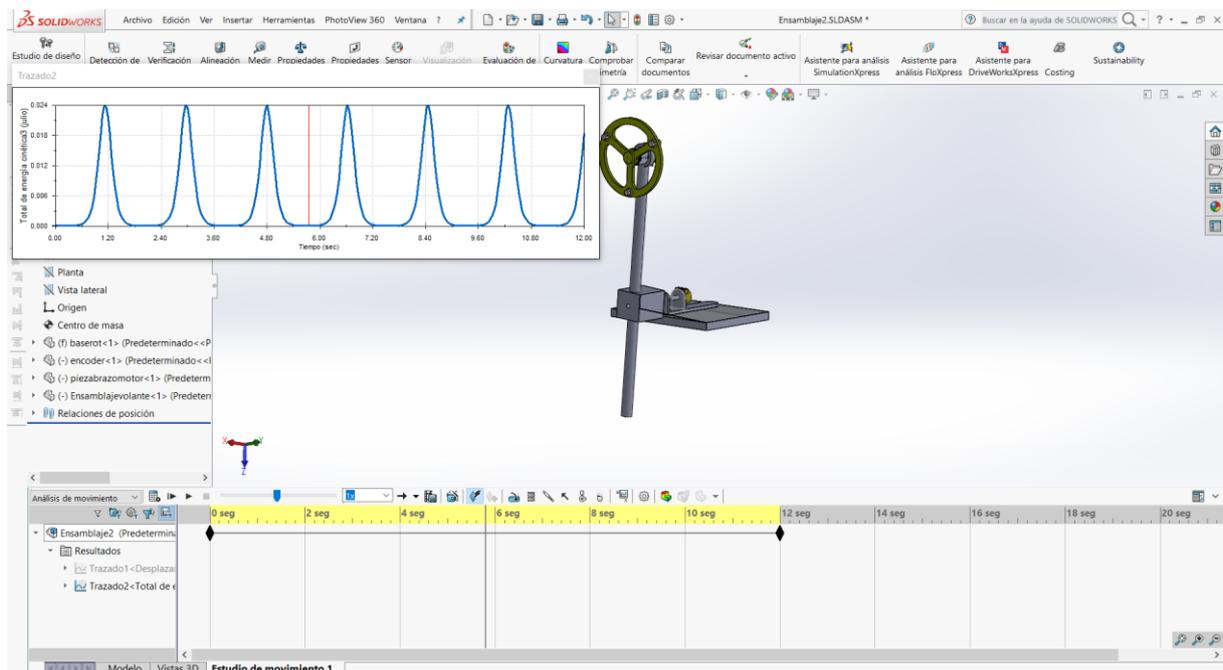
**Figura 7: Características Geométricas y Físicas**

Mediante la simulación arribamos a la conclusión que un volante con una masa concentrada en su periferia y con la opción de poder aumentar la misma mediante la inclusión de en nuestro caso mas tuercas era lo mas conveniente, pues nos permitia ajustar el mejor momento de inercia necesario. Como se puede apreciar en la figura mas abajo podemos ver según la simulación en SolidWork que momento necesitamos para levantar nuestro péndulo según el angulo en que se encuentre, siendo momento cero cuando nos encontramos con el péndulo en la posición de equilibrio inestable en el punto superior de su recorrido  $0^\circ$ .



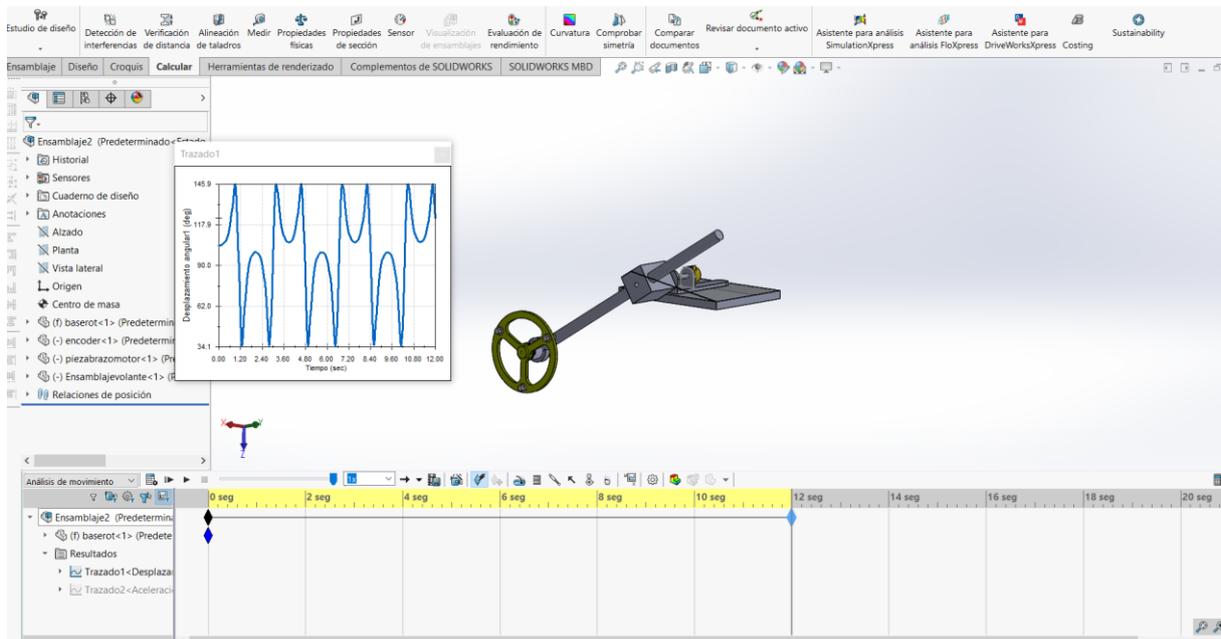
**Figura 8: Grafica del momento necesario según la posición.**

En la grafica superior podemos apreciar como el momento que genera el sistema según su posición, pudiendo identificar que en la posición de equilibrio inestable el momento es nulo, para mantenerlo en dicha posición.



**Figura 9: Energía Cinetica y potencial del sistema**

El estudio de la energía cinetica, es con el fin de evidenciar que la misma alcanza un valor máximo en la parte más baja del recorrido del sistema, y un valor nulo en el punto superior en donde la energía potencial es máxima. Mediante el análisis de energías también se puede determinar la potencia máxima del motor a necesitar.

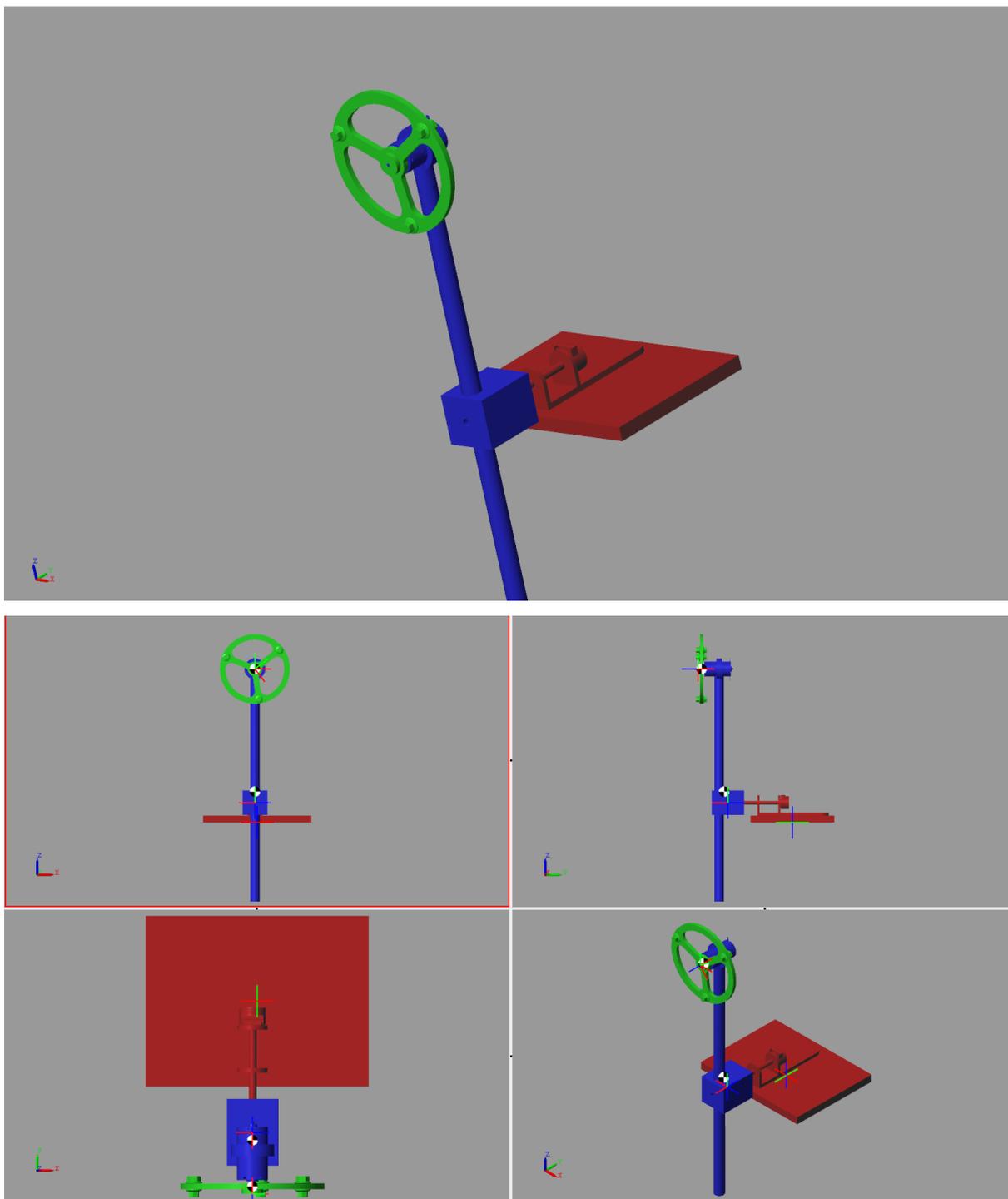


**Figura 10: Analisis del movimiento armonico del péndulo libre del sistema.**

### 3.2. Simulación Mediante Multibody Simulink Simscape

En nuestro sistema para construir la simulación, se utilizó el software CAD Solidworks, también hubiésemos podido utilizar el software Siemens Nx, posteriormente exportamos a formato step, este formato permite exportar la geometría aliviada de operaciones de dibujo. Para poder ser utilizada por Matlab Simulink, no obstante los parámetros físicos como pueden ser el peso o la densidad es necesario setearlos desde Multibody, ya que estas características no son exportadas.

A continuación en la imagen siguiente se verá como fue reconstruido las piezas exportadas a simulink, en donde se les asignó las vinculaciones o grados de libertad, y los sensores y actuadores para poder ser conectados a simulink y al control diseñado.



**Figura 11: Sistema exportado y ensamblado en simulink Simscape.**

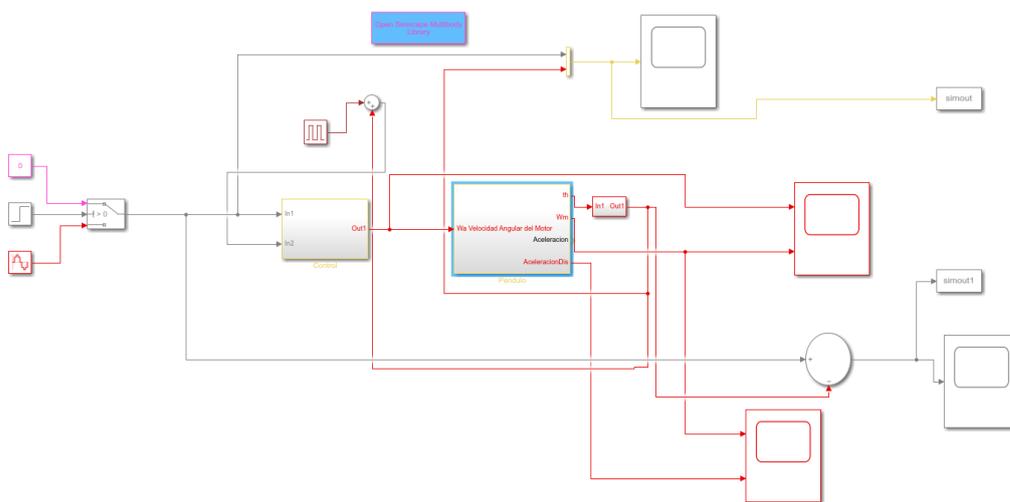
Para poder llevar a cabo la simulación dinámica con el control previamente seleccionado. A continuación enunciamos los pasos del que tuvimos en cuenta:

- Dibujar en un programa CAD las piezas a ser simuladas.

- Exportar las piezas a formato step, (formato aliviado para poder ser leído por Simscape).
- Separar el sistema en tantas partes como componentes relativos tengan cada una de las subpartes entre si.
- Ensamblar el sistema, en Simscape, cada unión de subpartes que tienen entre si deben asignarse el tipo de vinculación, por ejemplo si es de rotación o de traslación etc.
- Asignar la posición correcta en función del sistema de ejes coordenados.
- Asignar las características físicas a cada subsistema parte, ya sea masa o densidad.
- Asignar el tipo de sensor y actuador, en el caso de leer un ángulo por ejemplo sensor de medición de ángulo y en el caso de generar un movimiento de rotación un actuador de torque.(que en nuestro caso como controlamos por torque simulamos al motor).
- Conectar a nuestro modelo de control, previamente transformando las señales de físicas a Simulink, para poder ser interpretadas.

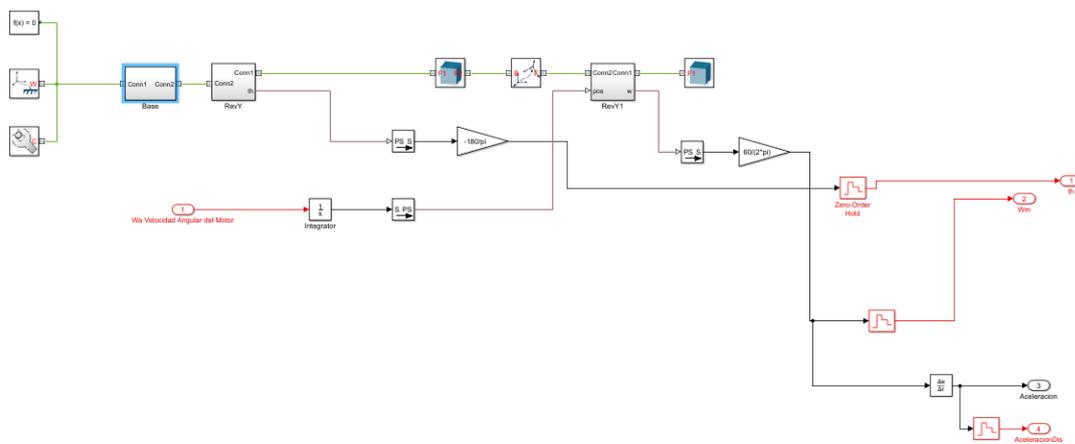
### Modelo Dinámico implementado.

En la figura de abajo se muestra el diagrama completo del modelo dinámico y el control donde se han utilizado los bloques de la librería “Multibody” y “Contact Forces”.



**Figura 129:** Péndulo con Volante de Inercia/Rueda de Reacción.

En la figura n°20 mostramos el bloque del modelo con sus adaptaciones al espacio físico. Los modelos cuando son importados a Simulink desde solidworks deben localizarse en su debida orientación mediante bloques de orientación puesto que desde simulink según sean importados el programa les elige una posición arbitraria.



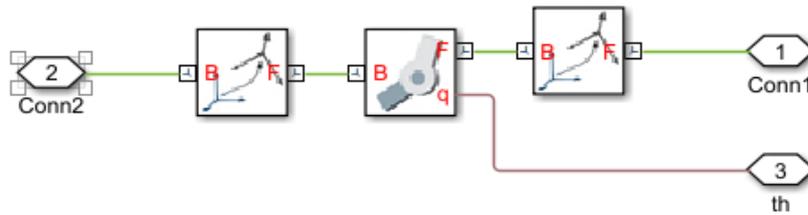
**Figura 130: Modelo diseñado en Simulink - Multibody**

El bloque denominado “Base” se presenta con más detalle en la siguiente figura:

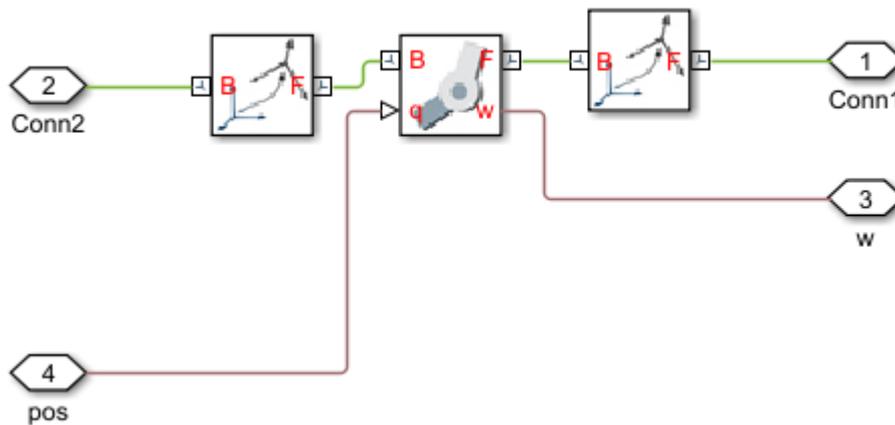


**Figura 141: Bloque "Base" ampliado con sus transformaciones de orientación para poder ser utilizado correctamente en el espacio de la simulación.**

En la siguiente figura podemos ver como se puede ver que en la vinculación de la base del péndulo que se encuentra restringida en el espacio, con el brazo del péndulo que pivotea rotacionalmente, extraemos la lectura de medida de ángulo “theta”.



**Figura 152: Modelo diseñado en Simulink – Multibody correspondiente a la base donde se encuentra el encoder para medir el ángulo del péndulo.**

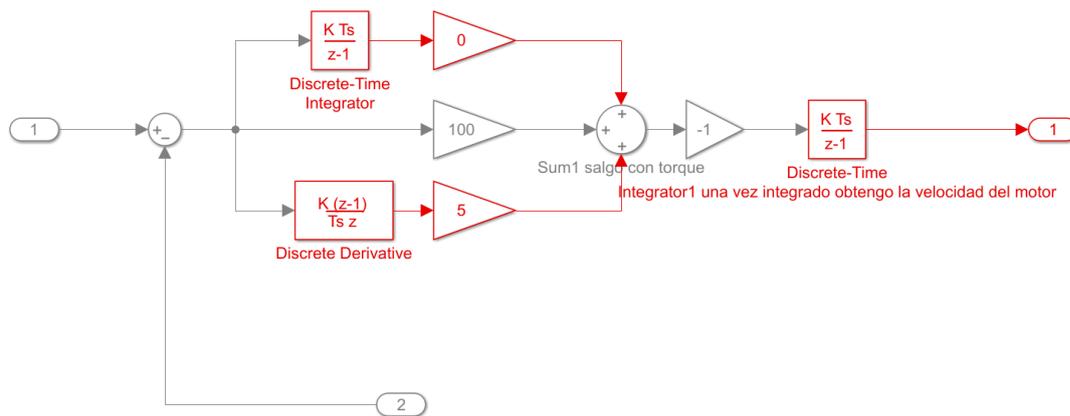


**Figura 163: Modelo diseñado en Simulink – Multibody de la vinculación del volante y motor dc con el brazo del péndulo.**

Se han empleado bloques que permiten enviar señales del modelo de Multibody (PS) a Simulink (S), y de éste modo poder enviar y recibir información hacia o desde el modelo dinámico diseñado.

### 3.2.1. Control del ángulo de posición:

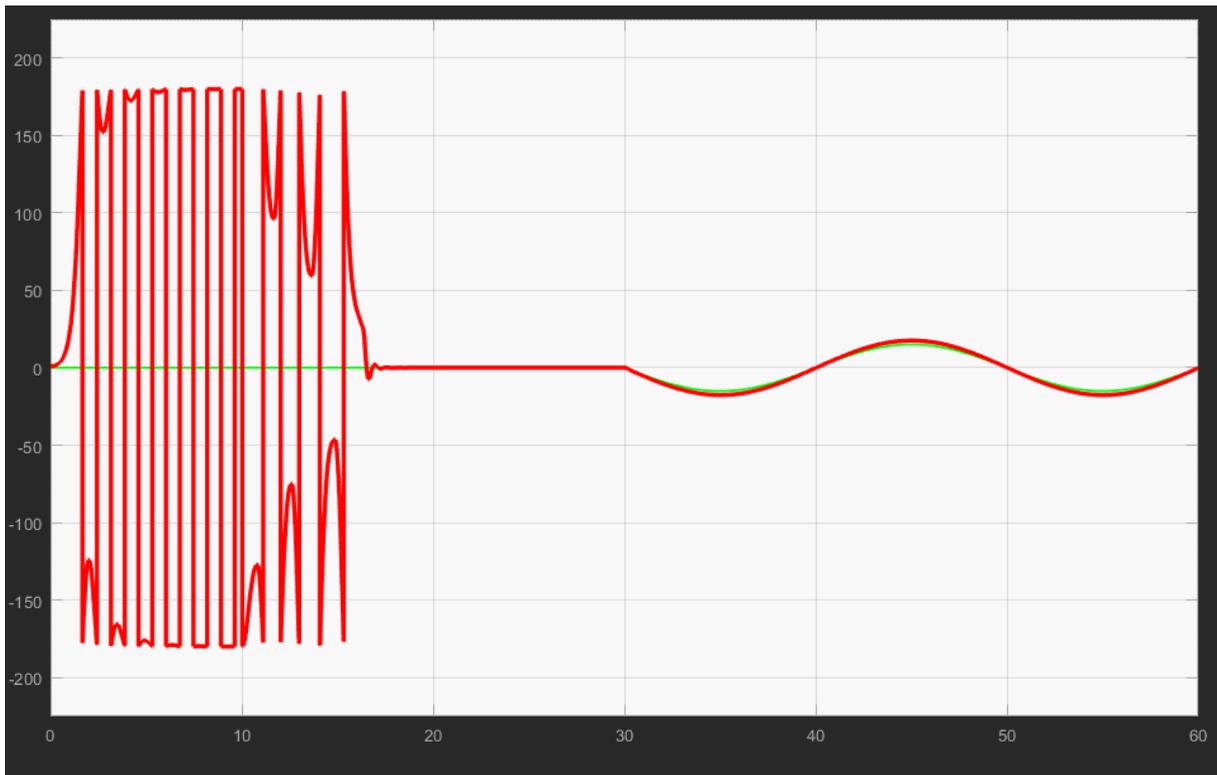
Para el control del ángulo utilizamos un control PID, en cual alimentamos con la posición del ángulo, y actuamos sobre la velocidad del motor del volante de inercia acelerando o desacelerando con el fin de aumentar o disminuir el torque del mismo.



**Figura 174: Regulador de posición sobre modelo de Multibody**

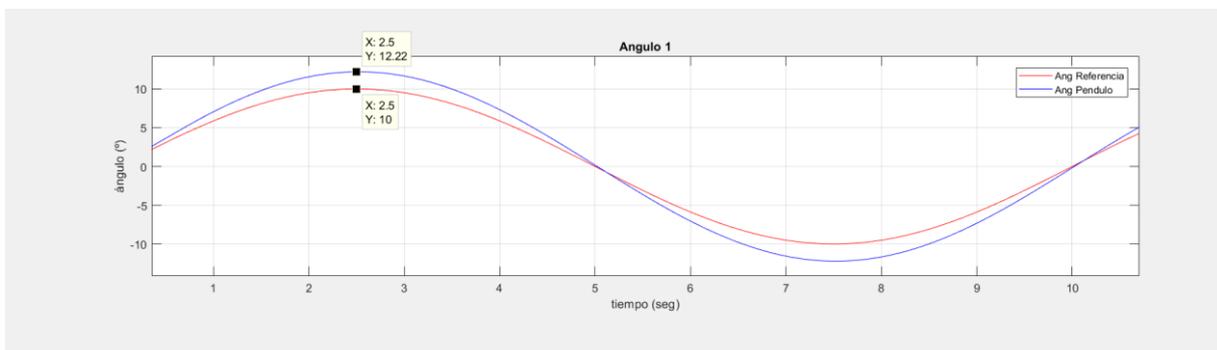
En la grafica anterior podemos ver la etapa implementada en MatLab Simulink del control PID utilizado en nuestro sistema de control

En la grafica a continuación apreciamos el comportamiento del movimiento del sistema simulado vs la referencia, en la grafica de abajo podemos ver como se levanta el sistema, logrando la posición 0, se mantiene y luego comienza a seguir la referencia.



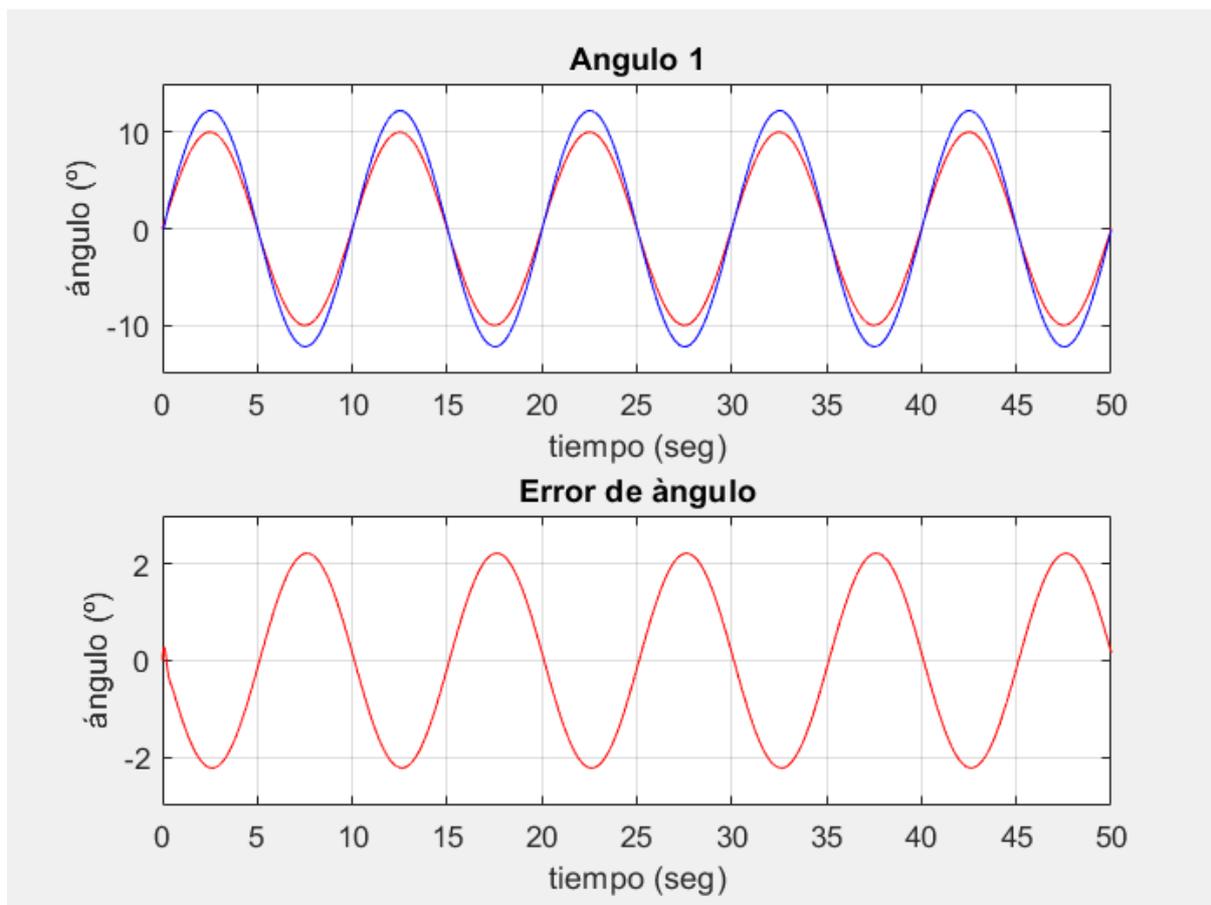
**Figura 185: Comportamiento del péndulo simulado**

En la figura siguiente se puede observar la señal de referencia roja, vs el comportamiento de la señal del sistema real azul, en donde debido a la inercia del sistema vemos una diferencia entre la amplitud de casi  $2,3^\circ$  aprox y sin retraso.



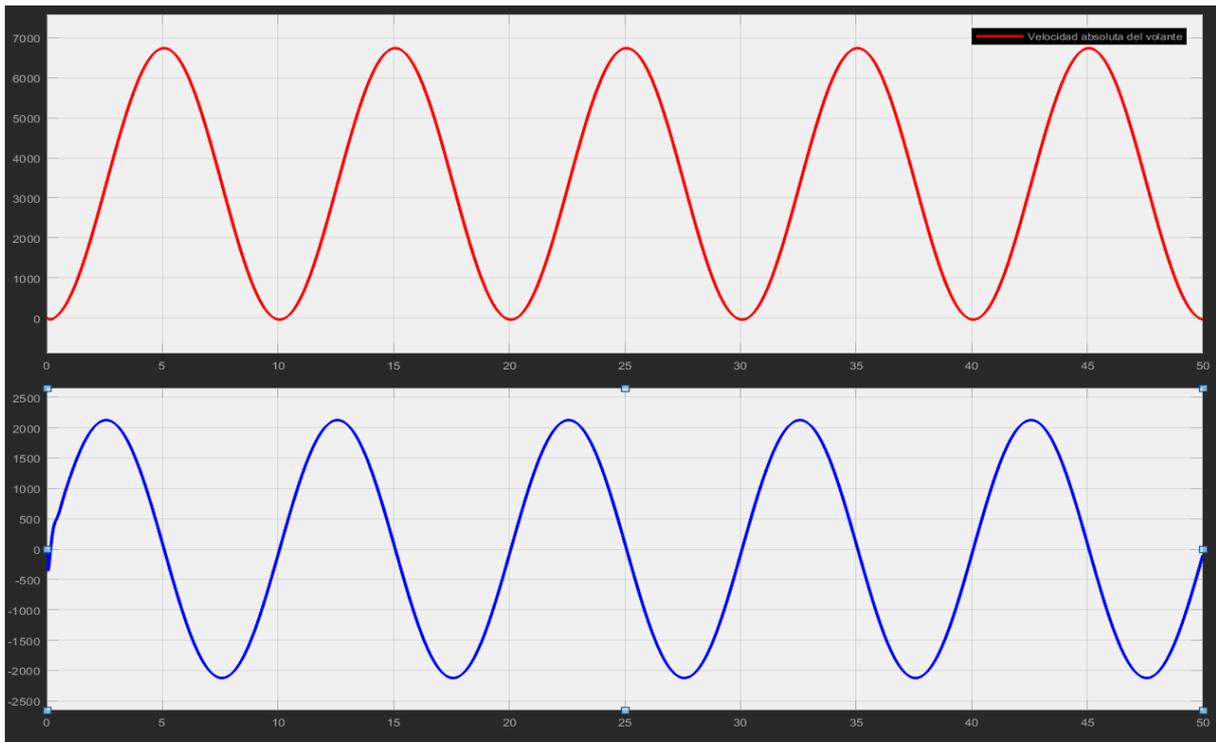
**Figura 196: Error de ángulo de péndulo vs referencia a seguir.**

Debido a la inercia del sistema se produce un error en el seguimiento que en nuestro caso es del orden de los  $2,5^\circ$ .



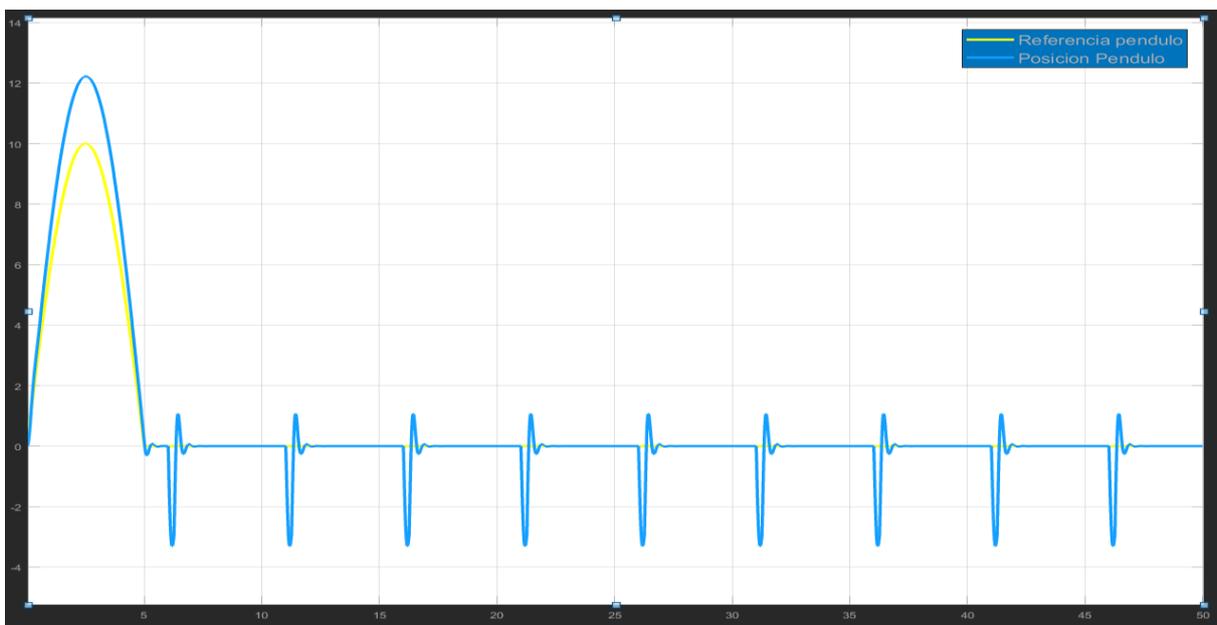
**Figura 207: Error de angulo de péndulo vs referencia a seguir (Fig Detallada).**

En la figura de abajo podemos apreciar como varia la velocidad de nuestro motor evidenciándose que alcanza los valores máximos y mínimos cuando la aceleración cruza por cero. Podemos ver que trabajamos hasta velocidades de casi 7000rpm. Mientras que en aceleraciones nos encontramos en 2000 rpm/seg



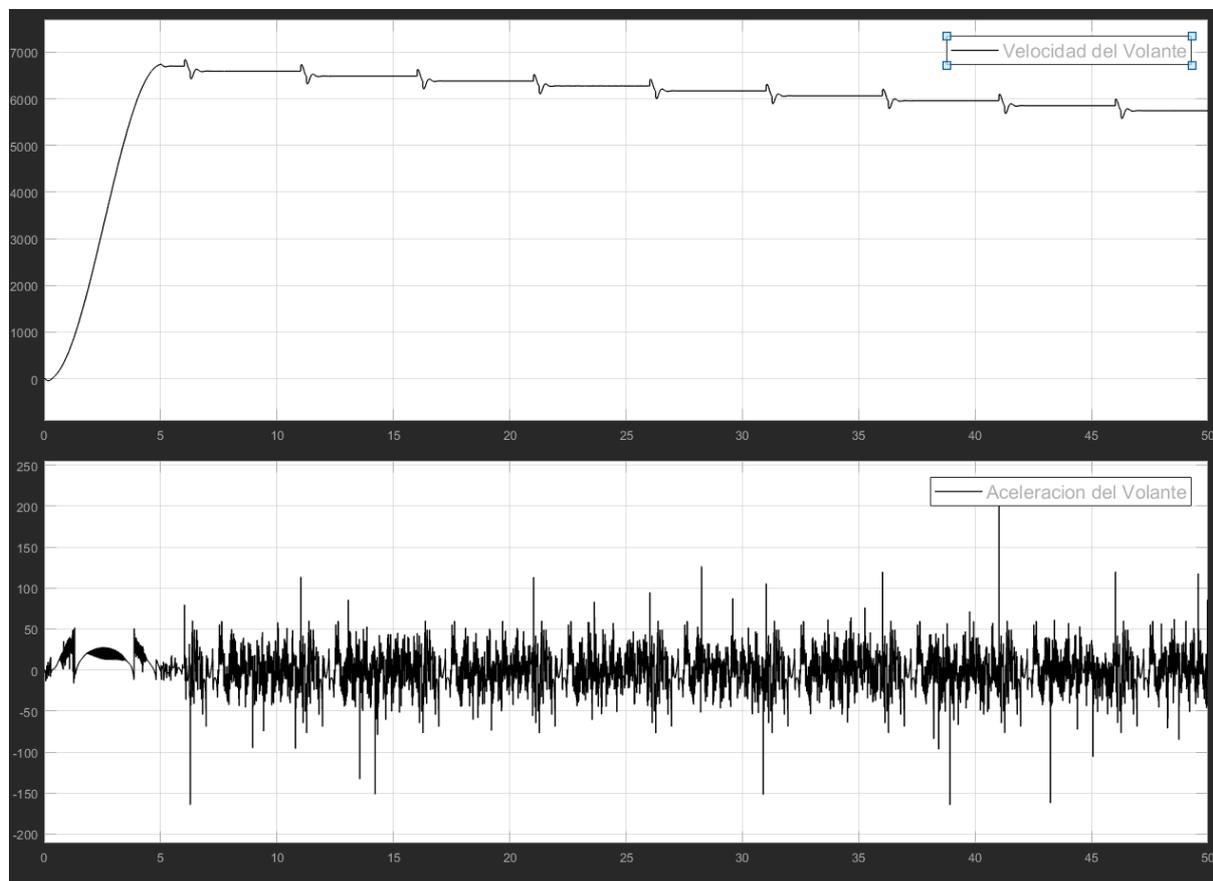
**Figura 217: Variación de la velocidad Absoluta (rojo) RPM vs Variación de la aceleración.**

En la figura anterior podemos apreciar las velocidades máximas desarrolladas por el volante cuando sigue una referencia senoidal el péndulo, y las aceleraciones máximas y mínimas del volante, como podemos apreciar en los momentos en que las velocidades son máximas las aceleraciones son nulas.



**Figura 228: Comportamiento de la trayectoria del péndulo vs perturbaciones.**

Aca podemos ver las señal de referencia vs la señal de lectura del péndulo cuando se van aplicando perturbaciones periódicas. Podemos apreciar como el sistema corrige la posición del volante y lo devuelve a su posición de referencia de  $0^\circ$ .



**Figura 239: Comportamiento de la velocidad y aceleración del pensulo vs las perturbaciones**

En las graficas anteriores se puede ver la lectura de velocidad del sistema simulado y los impulsos periódicos para desestabilizar el sistema. En la grafica inferior vemos como responde la aceleración para generar el momento u una vez estabilizado el volante en su posición vertical, vemos como a cada impulso de pesturbacion como responde la aceleración del volante de inercia.

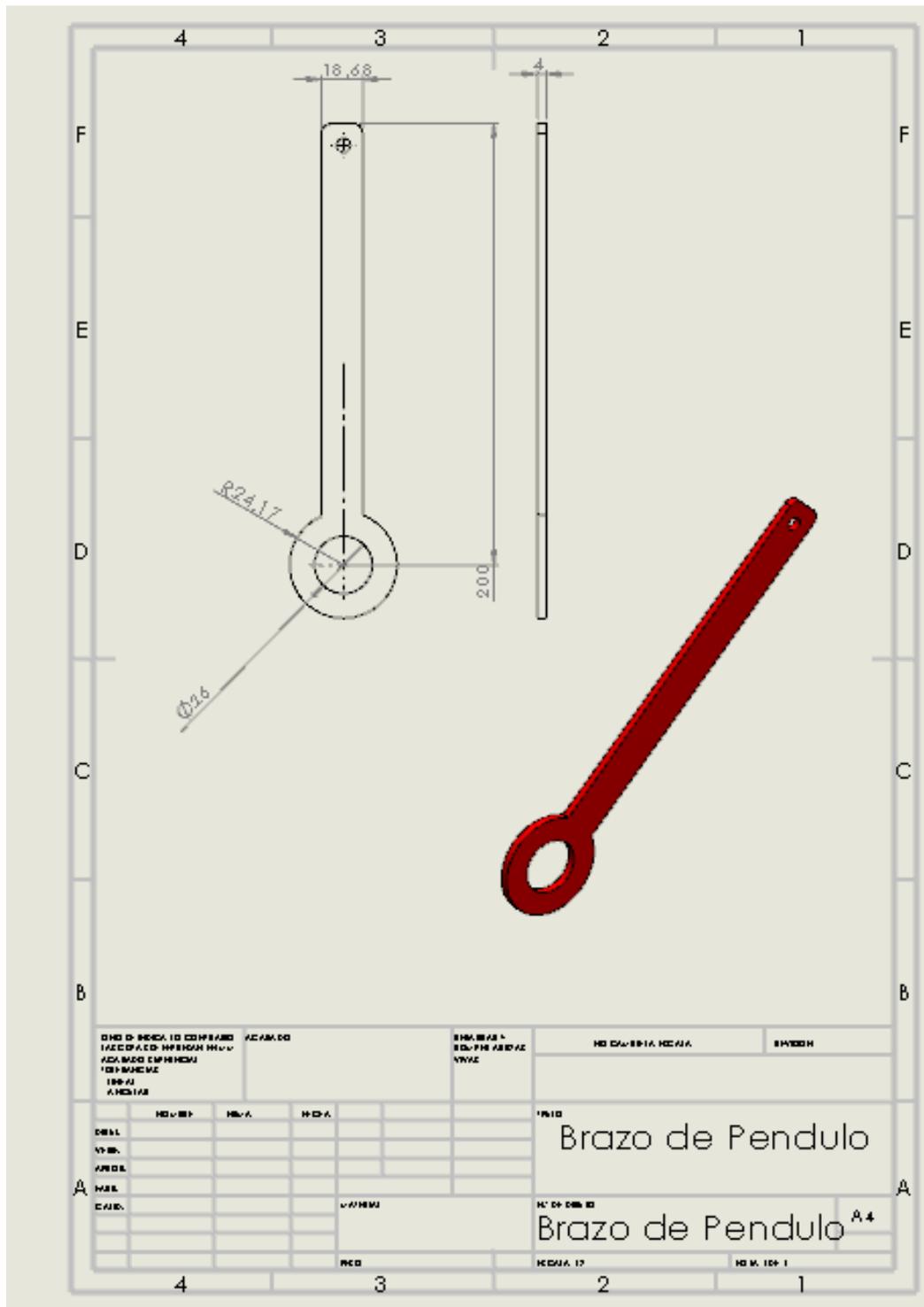
## **4. IMPLEMENTACION DEL PENDULO CON VOLANTE DE INERCIA**

En el siguiente capitulo, se ahondara en el diseño de las piezas del volante, su construcción y armado, su funcionamiento y respuestas.

### **4.1. Componentes**

Se han desarrollado varios modelos y/o prototipos solo mencionaremos las piezas de los modelos funcionales, ya que las piezas y diseños utilizados en lo intentos fallidos no los consideramos relevantes.

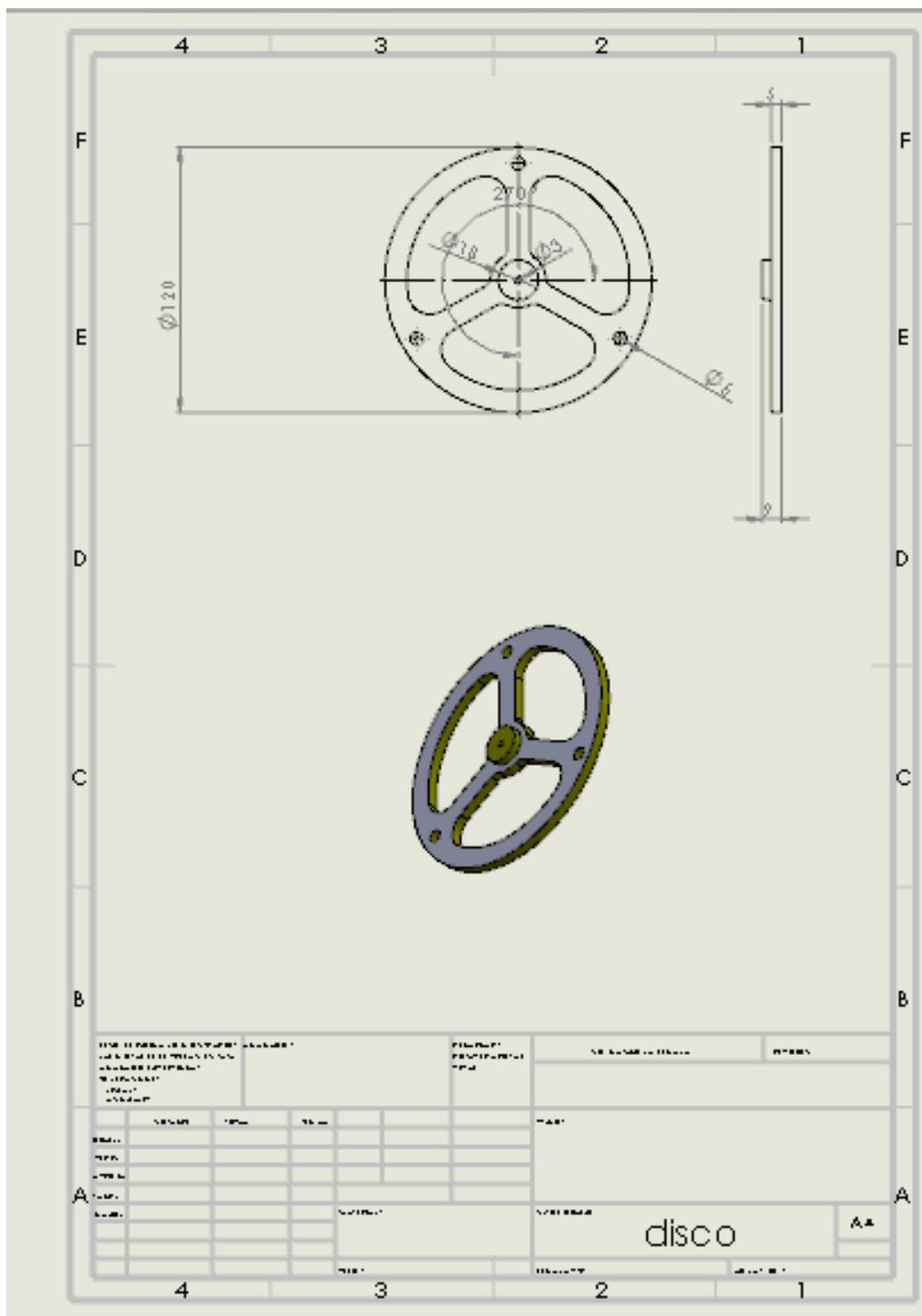
## 4.1.1. Piezas de DM cortadas con laser.



**Figura 240: Brazo del Pendulo**

Los planos de las piezas en material DM fueron exportados sin avotaciones en formato dxf, para poder ser levantado en el plotter de corte laser en tamaño real. El material DM, Un tablero DM es un aglomerado elaborado con fibras de madera (que previamente se han desfibrado y eliminado la lignina que poseían) aglutinadas con resinas sintéticas mediante fuerte presión y calor, en seco, hasta alcanzar una densidad media. También se le

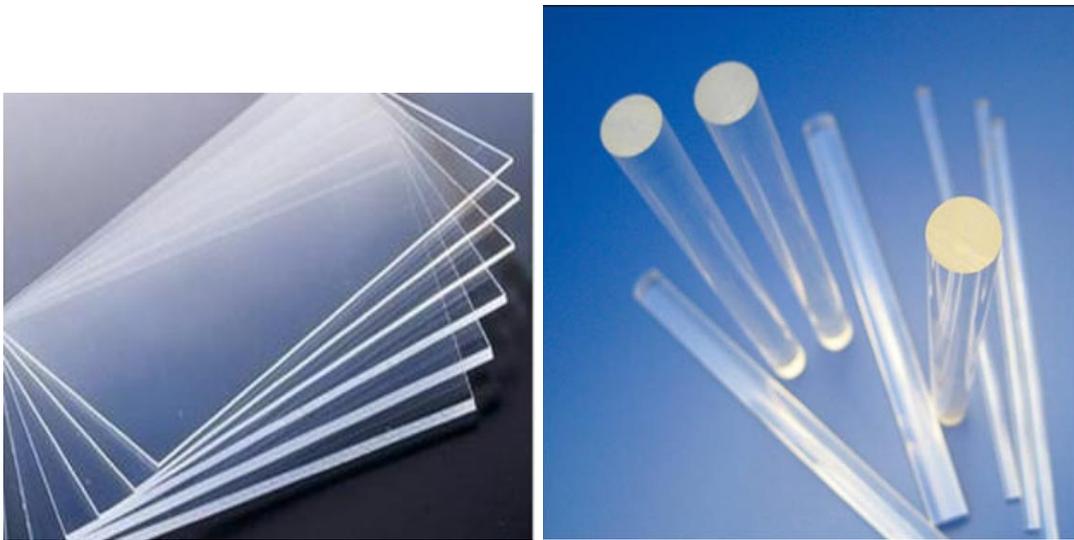
llama DM (densidad media), mdf (sigla en inglés de Medium Density Fibreboard), Trupán (nombre común en Chile derivado de la marca comercial) o Fibrofácil (denominado así en Argentina, también derivado de una marca).



**Figura 251:** Volante de inercia del pendulo

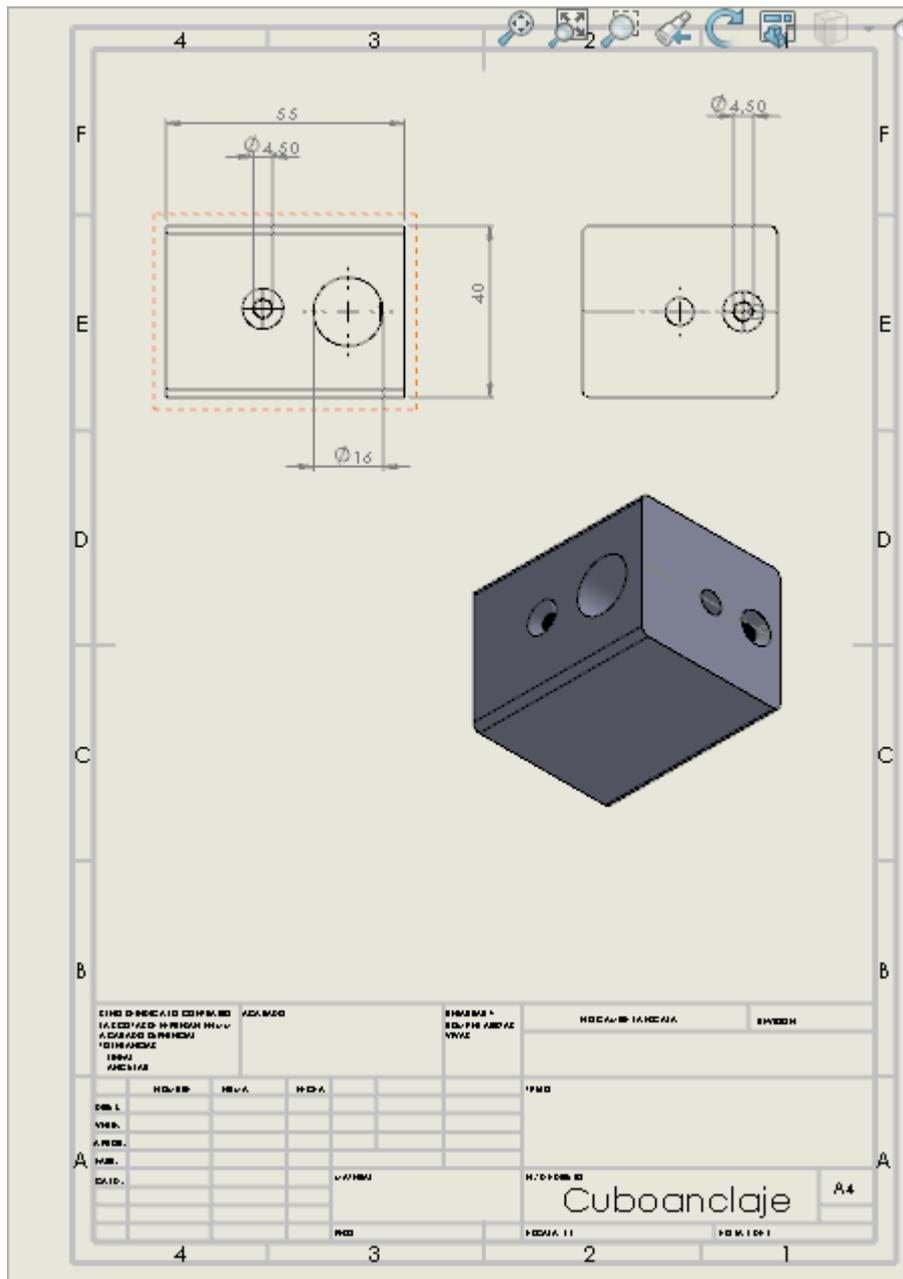
#### 4.1.2. Láminas de metacrilato y Barra de redondo de Metacrilato

El metacrilato o plexiglás es un termoplástico de gran resistencia y rigidez, se caracteriza por ser mucho más transparente que los plásticos en general, y por tener una alta resistencia al impacto comparado con el vidrio (casi 20 veces más resistente) a pesar de tener la mitad de su densidad (aproximadamente  $1190 \text{ Kg/m}^3$ ). También se ha utilizado un redondo en barra de 16mm de diámetro.



**Figura 262: Láminas de metacrilato y Redondo en 16mm**

#### 4.1.3. Cubo Anclaje para el modelo con varrilla de metacrilato.

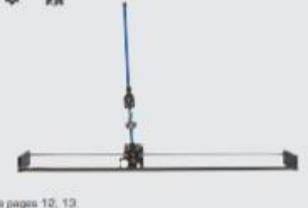
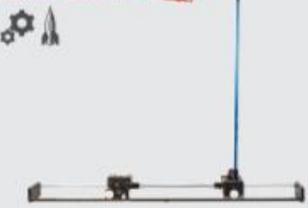


**Figura 273: Cubo de Anclaje**

En el modelo utilizado con brazo de péndulo realizado en varrilla redonda de metacrilato se utilizó el cubo anclaje. Realizado en material de metacrilato.



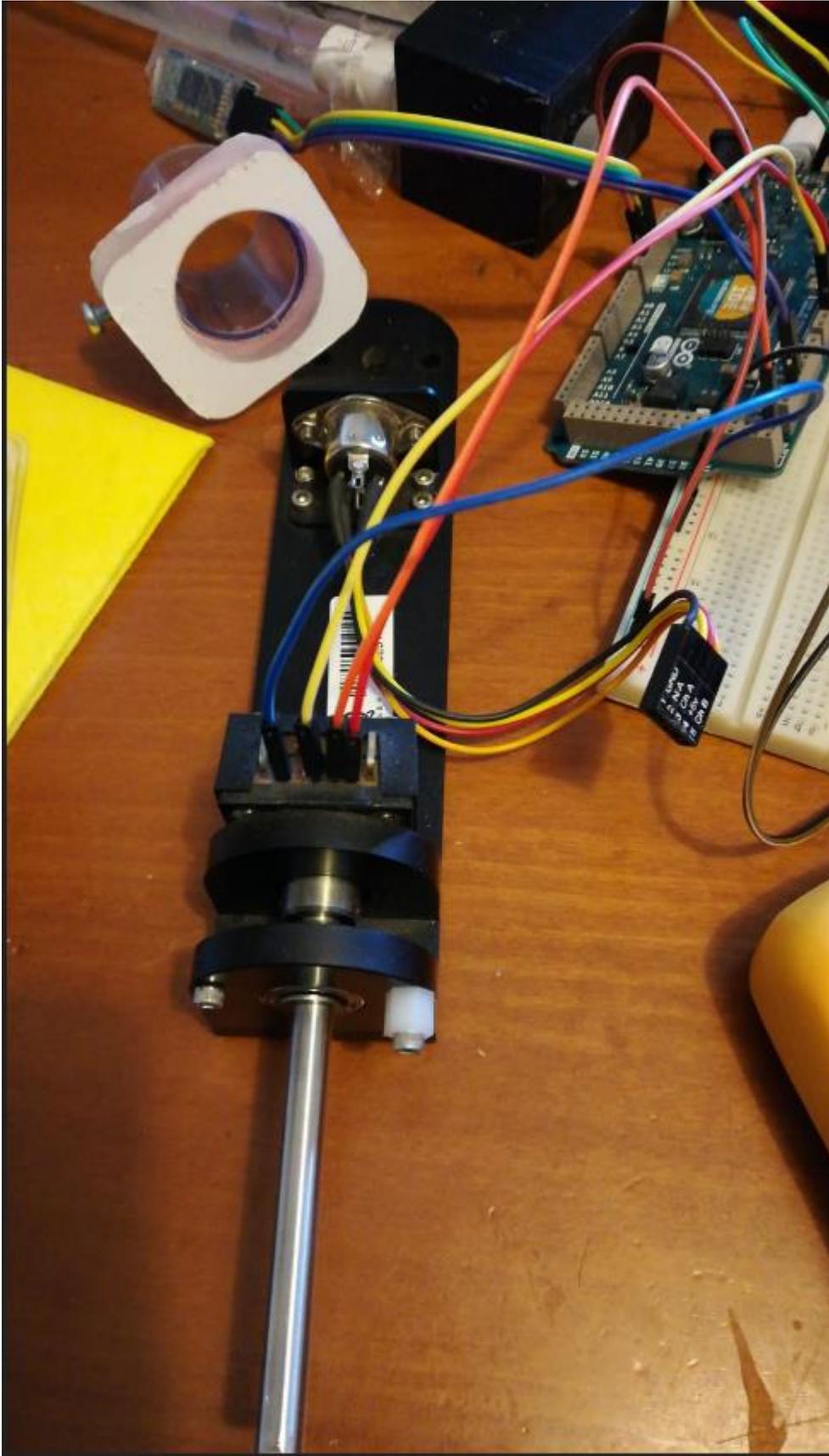
## BASE UNIT WITH ADD-ON MODULES. OVER 14 HANDS-ON LABS.

<p>▶ <b>Linear Inverted Pendulum</b> Intermediate </p>  <p>See pages 8, 9</p>	<p>▶ <b>Linear Flexible Inverted Pendulum</b> Intermediate  Advanced </p>  <p>See pages 10, 11</p>	<p><b>LEGEND</b></p> <p> <b>Electrical and Computer Engineering</b></p> <p>Various Engineering Departments can use Quanser systems to teach or explore control theories. Here are just some examples of the theories you can cover in Introductory, Intermediate and Advanced course levels.</p> <p> <b>Aerospace Engineering</b></p> <p><b>Introductory</b></p> <ul style="list-style-type: none"> <li>• First-order modeling and dynamics</li> <li>• Experimental modeling</li> <li>• Transfer functions</li> <li>• Relay control</li> <li>• PID control</li> <li>• Lead-lag compensation</li> <li>• SISO systems</li> </ul> <p><b>Mechanical Engineering</b></p> <p> <b>Mechanical Engineering</b></p> <p><b>Intermediate</b></p> <ul style="list-style-type: none"> <li>• Kinematics</li> <li>• Linearization</li> <li>• Linear state-space modeling</li> </ul> <p><b>Civil Engineering</b></p> <p> <b>Civil Engineering</b></p> <ul style="list-style-type: none"> <li>• State-feedback control (e.g. LQR)</li> <li>• Feed-forward control</li> <li>• Cascade control</li> <li>• Hybrid control</li> <li>• Structural dynamics</li> </ul> <p><b>Robotics</b></p> <p> <b>Robotics</b></p> <ul style="list-style-type: none"> <li>• Observer design [e.g. Kalman filter]</li> <li>• Nonlinear control</li> <li>• Sensor fusion</li> <li>• Haptic feedback</li> <li>• System identification</li> </ul> <p><b>Mechatronics</b></p> <p> <b>Mechatronics</b></p> <ul style="list-style-type: none"> <li>• Advanced modeling concepts</li> <li>• MIMO systems</li> </ul> <p><b>Courseware Included</b></p> <p>Many systems come complete with courseware that consist of instructor and student workbooks, or laboratory guide. The courseware covers a wide range of control theories. Contact <a href="mailto:info@quanser.com">info@quanser.com</a> for details.</p> <p> <b>Course Materials Available</b></p>
<p>▶ <b>Linear Double Inverted Pendulum</b> Intermediate  Advanced </p>  <p>See pages 12, 13</p>	<p>▶ <b>Seesaw</b> Intermediate </p>  <p>See pages 14, 15</p>	
<p>▶ <b>Seesaw Pendulum*</b> Intermediate  Advanced </p>  <p>See pages 14, 15</p>	<p>▶ <b>Linear Flexible Joint</b> Intermediate </p>  <p>See pages 16, 17</p>	
<p>▶ <b>Linear Flexible Joint on Seesaw</b> Intermediate  Advanced </p>  <p>See pages 16, 17</p>	<p>▶ <b>Linear Flexible Joint with Inverted Pendulum</b> Intermediate  Advanced </p>  <p>See pages 16, 19</p>	

**TO REQUEST A QUOTE, PLEASE EMAIL [INFO@QUANSER.COM](mailto:INFO@QUANSER.COM)**

3

*Figura 295: Ejemplo de Pendulos Invertidos de la firma Quanser que utilizan la base con encoder que fue reutilizada para nuestro modelo.*



*Figura 306: Base de Pendulo comercial reutilizada*

#### 4.1.5. Adaptador de eje en Aluminio



**Figura 317: Adaptadores de eje y características**

Se han usado estas bridas para realizar el anclaje de las piezas como son los volantes de inercia al eje del motor.

## 4.2. Componentes Electrónicos

En el péndulo hemos utilizado, encoders incrementales ópticos, etapas de potencia, placa de sistemas embebidos arduino due etc. Encoder Incremental Rotatorio en cuadratura.

¿Que es un encoder?

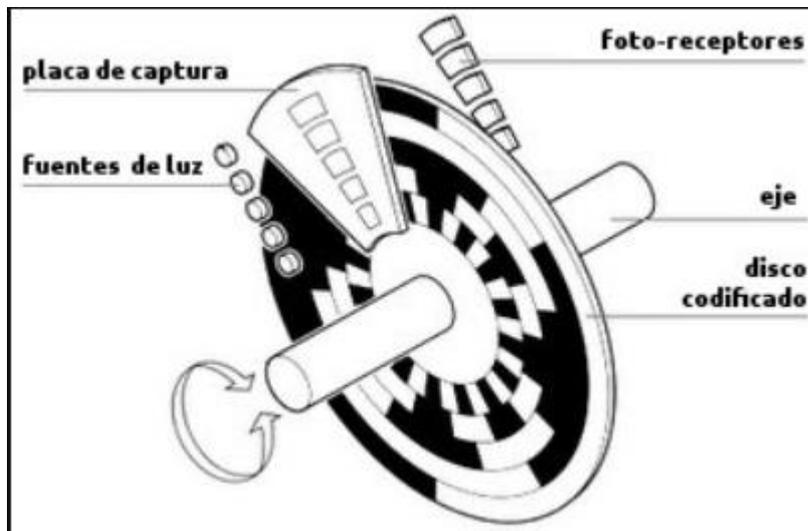
El encoder es un transductor rotativo, que mediante una señal eléctrica sirve para indicar la posición angular de un eje, velocidad y aceleración del rotor de un motor.

¿Como funciona un encoder?

Un encoder se compone básicamente de un disco conectado a un eje giratorio. El disco esta hecho de vidrio o plástico y se encuentra “codificado” con unas partes transparentes y otras opacas que bloquean el paso de la luz emitida por la fuente de luz (típicamente emisores infrarrojos). En la mayoría de los casos, estas áreas bloqueadas (codificadas) están arregladas en forma radial.

A medida que el eje rota, el emisor infrarojo emite luz que es recibida por el sensor óptico (o foto-transistor) generando los pulsos digitales a medida que la luz cruza a través del disco o es

bloqueada en diferentes secciones de este. Esto produce una secuencia que puede ser usada para controlar el radio de giro, la dirección del movimiento e incluso la velocidad.



*Figura 328: figura de un disco de encoder incremental*

Esto quiere decir que el encoder incremental provee una posición estratégica desde donde siempre comenzará la cuenta. La posición actual del encoder es incremental cuando es comparada con la última posición registrada por el sensor. Los encoders incrementales son un tipo de encoder óptico y este en este tipo de encoder cada posición es completamente única. Dentro de los encoders incrementales, se encuentran los encoders de cuadratura. Corresponde a un tipo de encoder incremental que utiliza dos sensores ópticos posicionados con un desplazamiento de  $1/4$  de ranura el uno del otro, generando dos señales de pulsos digitales desfasada en  $90^\circ$  o en cuadratura. A estas señales de salida, se les llama comúnmente A y B. Mediante ellas es posible suministrar los datos de posición, velocidad y dirección de rotación del eje. Si se incluye la señal de referencia, se le denomina I (índice). Usualmente, si la señal A adelanta a la señal B (la señal A toma valor lógico "1" antes que la señal B, por ejemplo), se establece el convenio de que el eje está rotando en sentido horario, mientras que si B adelanta a A, el sentido será antihorario.



Información general del producto	
Número de pieza de Digi-Key	516-3031-ND
Cantidad disponible	0 <a href="#">Controlar el plazo.</a>
Fabricante	<a href="#">Broadcom Limited</a>
Número de pieza del fabricante	HEDS-9100#J00
Descripción	ROTARY ENCODER OPTICAL 1024PPR
Plazo estándar del fabricante	8 semanas

Documentos y medios	
Hojas de datos	<a href="#">HEDS-9000/9100</a>

Atributos del producto		Seleccionar todos
Categorías	<a href="#">Sensores y transductores</a> <a href="#">Codificadores</a>	<input type="radio"/>
Fabricante	Broadcom Limited	<input checked="" type="radio"/>
Serie	<a href="#">HEDS</a>	<input type="radio"/>
Estado de la pieza	Activo	<input type="radio"/>
Tipo de codificador	Óptico	<input type="radio"/>
Tipo de salida	Cuadratura (por pasos)	<input type="radio"/>
Pulsos por revolución	1024	<input type="radio"/>
Voltaje de la fuente	5 V	<input type="radio"/>
Tipo de accionador	Disco decodificador no incluido	<input type="radio"/>
Tope	No	<input type="radio"/>
Interruptor incorporado	No	<input type="radio"/>
Tipo de montaje	Montaje de chasis	<input type="radio"/>
Orientación	Vertical	<input type="radio"/>
Tipo de terminación	Clavijas de terminal	<input type="radio"/>
Vida rotativa (ciclos por minutos)	-	<input type="radio"/>

[Informar un error](#)

[Buscar](#)

*Figura 339: Lector Optico Caracteristicas*



*Figura 340: Empaque Encoder*

### Motor Maxon:

En nuestro prototipo tuvimos que emplear un motor maxon especial de alta eficiencia puesto que necesitábamos un torque superiores a 0.170Nm, 7000 RPM o superior y un peso reducido. A continuación anexamos imágenes de las características técnicas eléctricas, mecánicas, geométricas y físicas del mismo.

**Motor - DC-MAX26S GB KL 24V**  
**Sensor - ENX16 EASY 1024IMP**

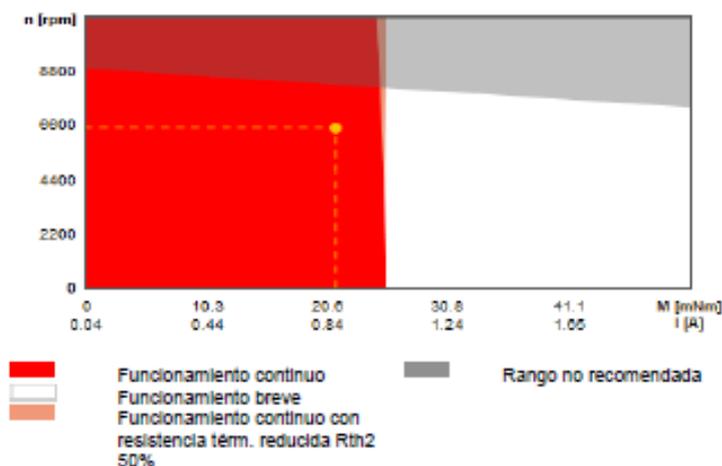
Peso total del motor 120 g

<b>Motor: DC-MAX26S GB KL 24V</b>	
Conmutación	Escobillas de grafito
Tensión nominal	24 V
Rodamientos motor	Rodamiento de bolas
<b>Sensor ENX16 EASY 1024IMP</b>	
<b>Número de pulsos</b>	
Número de pulsos	1024

Máx. temperatura del bobinado	100 °C
<b>Datos mecánicos</b>	
Máx. velocidad permitida	11000 rpm
Mín. juego axial	0.1 mm
Máx. juego axial	0.2 mm
Juego axial	0.025
Carga axial máx. (dinámica)	5 N
Máx. fuerza de montaje a presión (estática)	75 N
Máx. carga radial 5 mm desde la brida	20 N
<b>Más especificaciones</b>	
Número de pares de polos	1
Número de delgas del colector	13
Peso	120 g
Número de ciclos de esterilización	0

*Figura 351: Motor Maxon Características*

### Funcionamiento continuo



### Motor - DC-MAX26S GB KL 24V



#### Valores a tensión nominal

Tensión nominal	24 V
Velocidad en vacío	8930 rpm
Corriente en vacío	39.8 mA
Velocidad nominal	6100 rpm
Par nominal (máx. par permanente)	26.6 mNm
Corriente nominal (máx. corriente en continuo)	1.08 A
Par de arranque	287 mNm
Corriente de arranque	11.2 A
Máx. rendimiento	88.6 %

#### Datos característicos

Máx. potencia de salida	28.6 W
Resistencia en bornes	2.14 Ohm
Inductancia en bornes	0.278 mH
Constante de par	25.6 mNm/A
Constante de velocidad	373 rpm/V
Relación velocidad/par	31.2 rpm/mNm
Constante de tiempo mecánica	4.89 ms
Inercia del rotor	14.9 gcm <sup>2</sup>

#### Datos térmicos

Resistencia térmica carcasa/ambiente	13.2 K/W
Resistencia térmica bobinado/carcasa	3.2 K/W
Constante de tiempo térmica del bobinado	17.8 s
Constante de tiempo térmica del motor	350 s
Temperatura ambiente	-30..85 °C

21.03.2018 / Reservado el derecho a cambios / Su configuración: B789F8A96040 / Versión n°: 1

Página 5 de 7

*Figura 362: Motor Maxon Características*

Nº de artículo\*: B789F8A96040 Revision 1

driven by precision

Motor - DC-MAX26S GB KL 24V  
Sensor - ENX16 EASY 1024IMP

maxon motor en todo el mundo

[http://www.maxonmotor.es/maxon/view/content/contact\\_page](http://www.maxonmotor.es/maxon/view/content/contact_page)

E-mail: [e-shop@maxonmotor.com](mailto:e-shop@maxonmotor.com)  
Internet: [www.maxonmotor.com](http://www.maxonmotor.com)

Una vez recibido el pedido a través de la tienda electrónica, el producto se elaborará en Suiza y se entregará franco fábrica en un máximo de 11 días laborables.

Condiciones generales de contrato: [http://www.maxonmotor.es/maxon/view/content/terms\\_and\\_conditions\\_page](http://www.maxonmotor.es/maxon/view/content/terms_and_conditions_page)

#### Modelo 3D interactivo

Active el modelo 3D haciendo clic en la imagen



Aquí podrá ver su configuración:

<http://www.maxonmotor.es/maxon/view/configurator/?ConfigID=B789F8A96040>

\*Si el enlace no funciona, podrá Ud. volver a abrir el producto mediante el número del artículo en el catálogo online.

#### Archivos adjuntos / Archivos CAD

Haga doble clic en la chincheta para abrir un archivo. También puede guardar el archivo haciendo clic con el botón derecho del ratón.

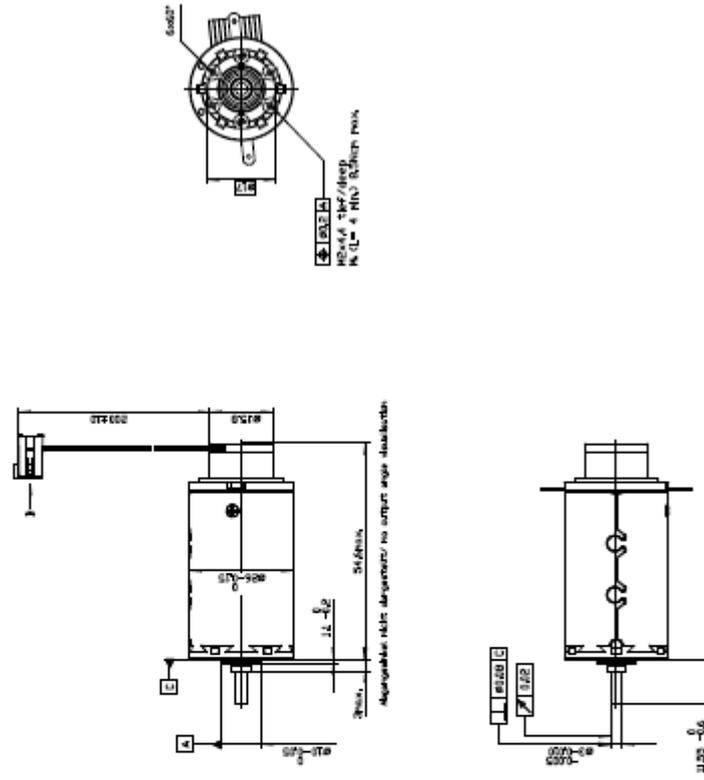
 B789F8A96040.stp (STP AP 214)

### *Figura 373: Motor Maxon Características*

Se optó por un motor Maxon, puesto que posee una gama de velocidades que llegan hasta las 20000 rpm, en nuestro caso según cálculos y simulación necesitábamos aproximadamente 9000 rpm, bajo peso del motor (este motor solo pesa 120grs) y un par de alrededor de 0.28Nm. Debido a la calidad de la fabricación de los imanes especiales del mismo se logra las características necesarias para el sistema de control empleado en el prototipo simulado. También trabajamos en un rango de corrientes de 200mA hasta 2 A.

Motor - DC-MAX26S GB KL 24V  
 Sensor - ENX16 EASY 1024IMP

(Dibujo no a escala!)

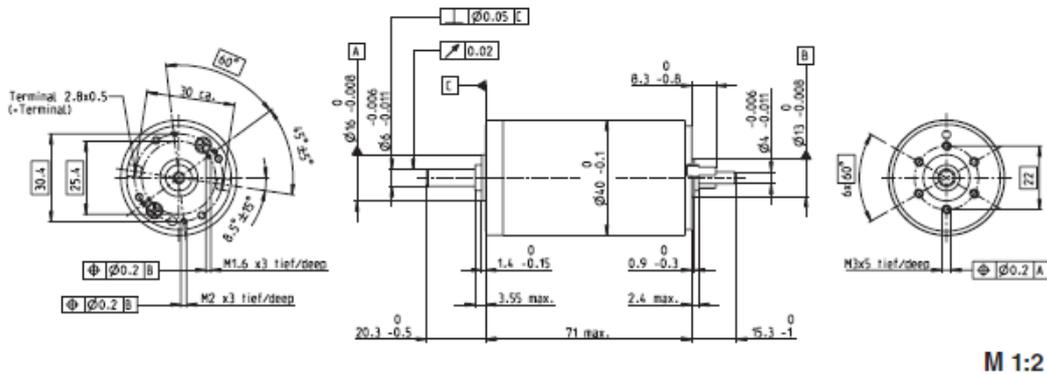


Axial play Motor: 0.1..0.2  
 axial play motor: 0.1..0.2

Figura 384: Motor Maxon Caracteristicas

maxon DC motor

## RE 40 Ø40 mm, Graphite Brushes, 150 Watt



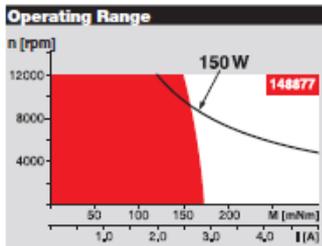
- Stock program
- Standard program
- Special program (on request)

### Article Numbers

148866 148967 148877 218008 218009 218010 218011 218012 218013 218014

Motor Data	148866	148967	148877	218008	218009	218010	218011	218012	218013	218014
<b>Values at nominal voltage</b>										
1 Nominal voltage	V	12	24	48	48	48	48	48	48	48
2 No load speed	rpm	6920	7580	7590	6420	5560	3330	2690	2130	1720
3 No load current	mA	241	137	68.6	53.6	43.7	21.9	16.6	12.5	9.66
4 Nominal speed	rpm	6380	6940	7000	5810	4930	2710	2060	1510	1080
5 Nominal torque (max. continuous torque)	mNm	94.9	177	187	186	180	189	190	192	190
6 Nominal current (max. continuous current)	A	6	6	3.17	2.66	2.23	1.4	1.13	0.909	0.73
7 Stall torque	mNm	1720	2420	2560	2040	1620	1020	814	655	523
8 Starting current	A	105	80.2	42.4	28.6	19.7	7.43	4.79	3.06	1.97
9 Max. efficiency	%	87	91	92	91	91	89	89	88	87
<b>Characteristics</b>										
10 Terminal resistance	Ω	0.115	0.299	1.13	1.68	2.44	6.46	10	15.7	24.4
11 Terminal inductance	mH	0.0245	0.0823	0.329	0.46	0.612	1.7	2.62	4.14	6.4
12 Torque constant	mNm/A	16.4	30.2	60.3	71.3	82.2	137	170	214	266
13 Speed constant	rpm/V	581	317	158	134	116	69.7	56.2	44.7	35.9
14 Speed / torque gradient	rpm/mNm	4.05	3.14	2.97	3.16	3.45	3.29	3.31	3.27	3.29
15 Mechanical time constant	ms	5.89	4.67	4.28	4.2	4.19	4.16	4.15	4.15	4.16
16 Rotor Inertia	gcm <sup>2</sup>	139	142	137	127	116	121	120	121	118

Specifications	
<b>Thermal data</b>	
17 Thermal resistance housing-ambient	4.7 K/W
18 Thermal resistance winding-housing	1.9 K/W
19 Thermal time constant winding	41.5 s
20 Thermal time constant motor	726 s
21 Ambient temperature	-30...+100°C
22 Max. permissible winding temperature	+155°C
<b>Mechanical data (ball bearings)</b>	
23 Max. permissible speed	12000 rpm
24 Axial play	0.05 - 0.15 mm
25 Radial play	0.025 mm
26 Max. axial load (dynamic)	5.6 N
27 Max. force for press fits (static) (static, shaft supported)	110 N
28 Max. radial loading, 5 mm from flange	1200 N



**Comments**

- Continuous operation**  
In observation of above listed thermal resistance (lines 17 and 18) the maximum permissible winding temperature will be reached during continuous operation at 25°C ambient.  
- Thermal limit.
- Short term operation**  
The motor may be briefly overloaded (recurring).
- Assigned power rating**

Other specifications	
29 Number of pole pairs	1
30 Number of commutator segments	13
31 Weight of motor	480 g

Values listed in the table are nominal.  
Explanation of the figures on page 45.

**Option**  
Preloaded ball bearings

**maxon Modular System**

**Planetary Gearhead**  
Ø42 mm  
3 - 15 Nm  
Page 242

**Planetary Gearhead**  
Ø52 mm  
4 - 30 Nm  
Page 245

**Overview on page 16 - 21**

- Encoder MR 256 - 1024 Imp., 3 channels Page 273
- Encoder HEDL\_5540 500 CPT, 3 channels Page 278/278
- Brake AB 28 24 VDC 0.4 Nm Page 330
- Industrial Version Encoder HEDL 9140 Page 281
- Brake AB 28 Page 331
- End cap Page 335

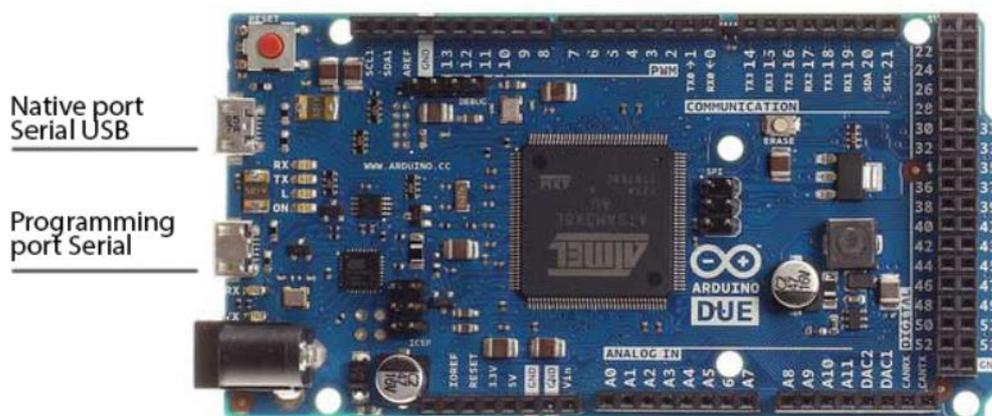
**Recommended Electronics:**

- ESCON 50/5 Page 292
- EPOS2 24/5 Page 313
- EPOS2 50/5 Page 313
- EPOS2 70/10 Page 313
- EPOS2 P 24/5 Page 316
- EPOS3 70/10 EtherCAT Page 319
- ADS 50/10 Page 373
- ADS\_E 50/10 Page 373
- Notes Page 18

Figura 395: Motor Maxon Caracteristicas

## PLACA ARDUINO DUE:

Serial ports on the Due



Características: Arduino Due	Características: AT91SAM3X8E
Microcontrolador: AT91SAM3X8E	Voltaje de Operación: 3.3V
Voltaje de funcionamiento: 3.3 V	Pines de E/S Digital: 54 (de los cuales 12 proveen salida PWM)
Pines I/O digitales: 54	Pines de Entrada Análoga: 12
Pines de entradas análogas: 12	Pines de Salida Análoga: 2 (DAC)
Corriente DC total en todos los pines I/O: 130 mA	Corriente DC Total de Salida en todas las líneas de E/S: 130 mA
Corriente DC en el pin de 5 V: 800 mA	Corriente DC para pin 3.3V: 800 mA
Corriente DC en el pin de 3.3 V: 800 mA	Corriente DC para pin 5V: 800 mA
Memoria Flash: 512 KB disponibles para las aplicaciones de usuario.	Memoria Flash: 512 KB
SRAM: 96 KB (dos bancos: 64KB Y 32 KB)	SRAM: 96 KB (dos bancos: 64KB y 32KB)
Velocidad de reloj: 84 MHz	Velocidad de Reloj: 84 MHz

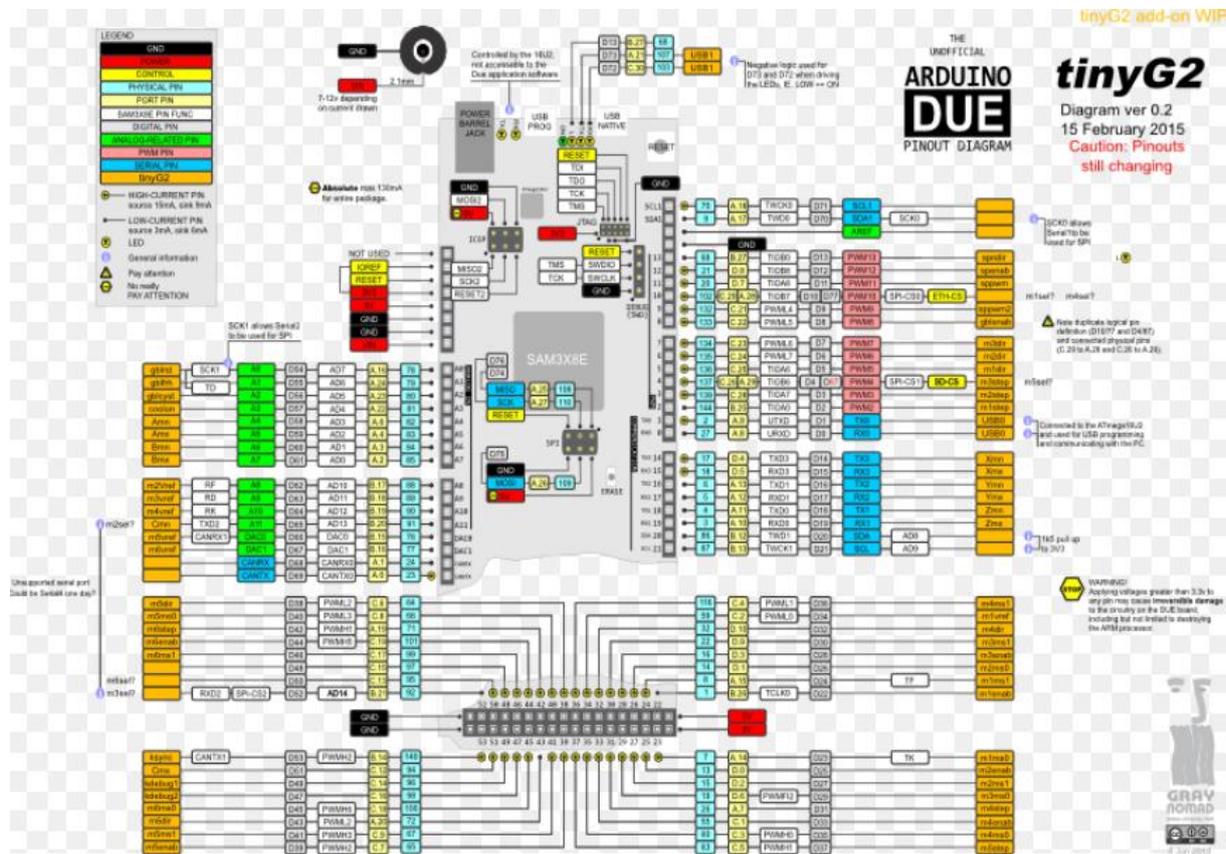
*Figura 406: Arduino Due Imagen Y Características.*

En nuestro proyecto hemos utilizado arduino due puesto que necesitamos cumplir ciclos de trabajo de 10ms, y con esta placa la cual es la mas potente de la familia podemos trabajar a frecuencias de 84MHZ.

EL control del portotipo se programo en el sistema embebido, y desde la pc solo se envio las señales de referencia para su seguimiento, y se recibió la lectura de las señales para graficar y análisis. El control del prototipo no se podía manejar desde MATLAB SIMULINK puesto que logramos velocidades de ciclo de 50ms, perdiendo gran cantidad del paquete de datos, y como

el control de este sistema es de dinámica rápida por debajo de los 30 ms, se optó por directamente programar el control en la placa y controlarlo desde allí.

EL problema de no poder cumplir con la velocidad de ciclo de control es debido principalmente a la velocidad de comunicación del puerto serie.



**Figura 417: Arduino Due disposición de pines y funciones**

El Arduino Due es el más poderoso controlador de la familia Arduino, está desarrollado basado en un microcontrolador Atmel SAM3X8E que tiene un procesador ARM Cortex-M3 de 32 bits. Con esta placa tus proyectos tendrán una super capacidad de procesamiento y recursos de entradas/salidas similares a los de un Arduino Mega 2560.

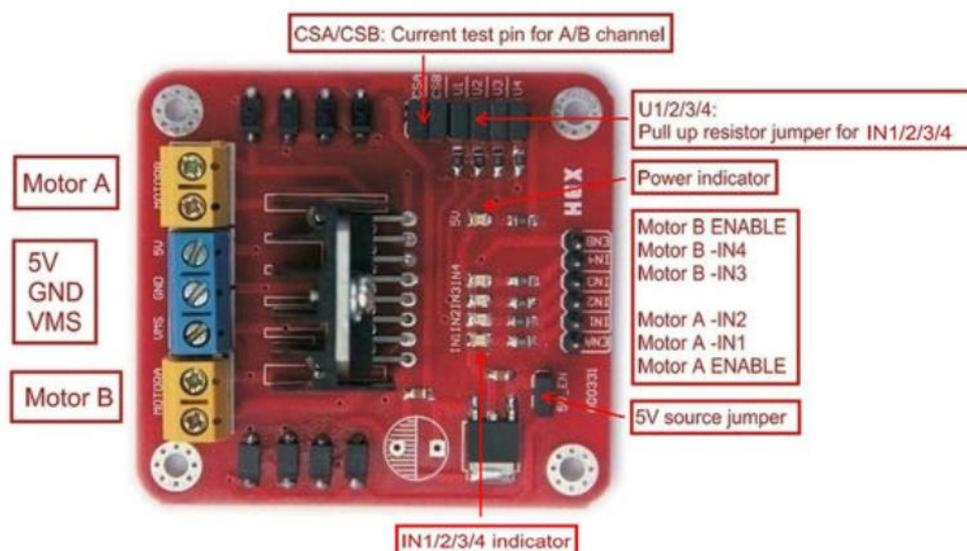
El Due tiene un total de 54 entradas/salidas (12 de las cuales pueden ser usadas como salidas de PWM), 12 entradas análogas, 4 UARTs (puertos seriales), un reloj de 84MHz, una conexión USB-OTG, 2 DAC (convertidores análogos-digitales), 2 TWI, un conector SPI, un conector JTAG, un botón de reset y un botón de borrar. Adicionalmente a lo anterior otras características como DACs, Audio, DMA, y una librería experimental de multitarea.

La placa contiene todo lo que se necesita para hacer funcionar el microcontrolador, simplemente se debe conectar al computador con el cable micro-USB o alimentarlo por la alimentación DC

o una batería para iniciar. El Due es compatible con todos los shield de Arduino que funcionen a 3.3V y que sea compatibles con la asignación de pines del Arduino 1.0.

**Advertencia:** A diferencia de todas las otras placas Arduino, el Arduino Due funciona a 3.3V. El máximo voltaje que los pines de entradas/salidas pueden soportar es de 3.3V. Alimentar con voltajes superiores puede dañar la placa.

#### 4.2.1. Driver de potencia L298N



*Figura 428: Imagen Driver o etapa de potencia L298N*

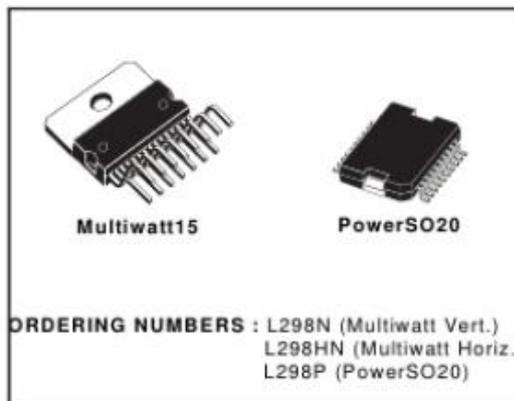
Imagen del Driver de etapa de potencia utilizado en el proyecto anterior con el fin de trabajar regulando la fuente de tensión a 9 v y suministrar hasta 2 amper de corriente sin dañarse.

## DUAL FULL-BRIDGE DRIVER

- OPERATING SUPPLY VOLTAGE UP TO 46 V
- TOTAL DC CURRENT UP TO 4 A
- LOW SATURATION VOLTAGE
- OVERTEMPERATURE PROTECTION
- LOGICAL "0" INPUT VOLTAGE UP TO 1.5 V (HIGH NOISE IMMUNITY)

### DESCRIPTION

The L298 is an integrated monolithic circuit in a 15-lead Multiwatt and PowerSO20 packages. It is a high voltage, high current dual full-bridge driver designed to accept standard TTL logic levels and drive inductive loads such as relays, solenoids, DC and stepping motors. Two enable inputs are provided to enable or disable the device independently of the input signals. The emitters of the lower transistors of each bridge are connected together and the corresponding external terminal can be used for the con-



nection of an external sensing resistor. An additional supply input is provided so that the logic works at a lower voltage.

### ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Value	Unit
$V_S$	Power Supply	50	V
$V_{SS}$	Logic Supply Voltage	7	V
$V_i, V_{en}$	Input and Enable Voltage	-0.3 to 7	V
$I_O$	Peak Output Current (each Channel)		
	- Non Repetitive ( $t = 100\mu s$ )	3	A
	- Repetitive (80% on -20% off; $t_{on} = 10ms$ )	2.5	A
	-DC Operation	2	A
$V_{sens}$	Sensing Voltage	-1 to 2.3	V
$P_{tot}$	Total Power Dissipation ( $T_{case} = 75^\circ C$ )	25	W
$T_{op}$	Junction Operating Temperature	-25 to 130	$^\circ C$
$T_{stg}, T_j$	Storage and Junction Temperature	-40 to 150	$^\circ C$

Figura 439: Características técnicas de la etapa de potencia L298N

**ELECTRICAL CHARACTERISTICS** ( $V_S = 42V$ ;  $V_{SS} = 5V$ ,  $T_J = 25^\circ C$ ; unless otherwise specified)

Symbol	Parameter	Test Conditions	Min.	Typ.	Max.	Unit
$V_S$	Supply Voltage (pin 4)	Operative Condition	$V_{IH} + 2.5$		46	V
$V_{SS}$	Logic Supply Voltage (pin 9)		4.5	5	7	V
$I_S$	Quiescent Supply Current (pin 4)	$V_{en} = H$ ; $I_L = 0$ $V_i = L$		13	22	mA
				50	70	mA
$I_{SS}$	Quiescent Current from $V_{SS}$ (pin 9)	$V_{en} = L$ $V_i = X$			4	mA
		$V_{en} = H$ ; $I_L = 0$ $V_i = L$		24	36	mA
		$V_i = H$		7	12	mA
		$V_{en} = L$ $V_i = X$			6	mA
$V_L$	Input Low Voltage (pins 5, 7, 10, 12)		-0.3		1.5	V
$V_{IH}$	Input High Voltage (pins 5, 7, 10, 12)		2.3		$V_{SS}$	V
$I_L$	Low Voltage Input Current (pins 5, 7, 10, 12)	$V_i = L$			-10	$\mu A$
$I_{IH}$	High Voltage Input Current (pins 5, 7, 10, 12)	$V_i = H \leq V_{SS} - 0.6V$		30	100	$\mu A$
$V_{en} = L$	Enable Low Voltage (pins 6, 11)		-0.3		1.5	V
$V_{en} = H$	Enable High Voltage (pins 6, 11)		2.3		$V_{SS}$	V
$I_{en} = L$	Low Voltage Enable Current (pins 6, 11)	$V_{en} = L$			-10	$\mu A$
$I_{en} = H$	High Voltage Enable Current (pins 6, 11)	$V_{en} = H \leq V_{SS} - 0.6V$		30	100	$\mu A$
$V_{CEsat(H)}$	Source Saturation Voltage	$I_L = 1A$	0.95	1.35	1.7	V
		$I_L = 2A$		2	2.7	V
$V_{CEsat(L)}$	Sink Saturation Voltage	$I_L = 1A$ (5)	0.85	1.2	1.6	V
		$I_L = 2A$ (5)		1.7	2.3	V
$V_{CEsat}$	Total Drop	$I_L = 1A$ (5)	1.80		3.2	V
		$I_L = 2A$ (5)			4.9	V
$V_{sens}$	Sensing Voltage (pins 1, 15)		-1 (1)		2	V

*Figura 440: Características técnicas de la etapa de potencia L298N*

## L298

## ELECTRICAL CHARACTERISTICS (continued)

Symbol	Parameter	Test Conditions	Min.	Typ.	Max.	Unit
$T_1$ (V <sub>I</sub> )	Source Current Turn-off Delay	0.5 V <sub>I</sub> to 0.9 I <sub>L</sub> (2); (4)		1.5		μs
$T_2$ (V <sub>I</sub> )	Source Current Fall Time	0.9 I <sub>L</sub> to 0.1 I <sub>L</sub> (2); (4)		0.2		μs
$T_3$ (V <sub>I</sub> )	Source Current Turn-on Delay	0.5 V <sub>I</sub> to 0.1 I <sub>L</sub> (2); (4)		2		μs
$T_4$ (V <sub>I</sub> )	Source Current Rise Time	0.1 I <sub>L</sub> to 0.9 I <sub>L</sub> (2); (4)		0.7		μs
$T_5$ (V <sub>I</sub> )	Sink Current Turn-off Delay	0.5 V <sub>I</sub> to 0.9 I <sub>L</sub> (3); (4)		0.7		μs
$T_6$ (V <sub>I</sub> )	Sink Current Fall Time	0.9 I <sub>L</sub> to 0.1 I <sub>L</sub> (3); (4)		0.25		μs
$T_7$ (V <sub>I</sub> )	Sink Current Turn-on Delay	0.5 V <sub>I</sub> to 0.9 I <sub>L</sub> (3); (4)		1.6		μs
$T_8$ (V <sub>I</sub> )	Sink Current Rise Time	0.1 I <sub>L</sub> to 0.9 I <sub>L</sub> (3); (4)		0.2		μs
f <sub>c</sub> (V)	Commutation Frequency	I <sub>L</sub> = 2A		25	40	KHz
$T_1$ (V <sub>en</sub> )	Source Current Turn-off Delay	0.5 V <sub>en</sub> to 0.9 I <sub>L</sub> (2); (4)		3		μs
$T_2$ (V <sub>en</sub> )	Source Current Fall Time	0.9 I <sub>L</sub> to 0.1 I <sub>L</sub> (2); (4)		1		μs
$T_3$ (V <sub>en</sub> )	Source Current Turn-on Delay	0.5 V <sub>en</sub> to 0.1 I <sub>L</sub> (2); (4)		0.3		μs
$T_4$ (V <sub>en</sub> )	Source Current Rise Time	0.1 I <sub>L</sub> to 0.9 I <sub>L</sub> (2); (4)		0.4		μs
$T_5$ (V <sub>en</sub> )	Sink Current Turn-off Delay	0.5 V <sub>en</sub> to 0.9 I <sub>L</sub> (3); (4)		2.2		μs
$T_6$ (V <sub>en</sub> )	Sink Current Fall Time	0.9 I <sub>L</sub> to 0.1 I <sub>L</sub> (3); (4)		0.35		μs
$T_7$ (V <sub>en</sub> )	Sink Current Turn-on Delay	0.5 V <sub>en</sub> to 0.9 I <sub>L</sub> (3); (4)		0.25		μs
$T_8$ (V <sub>en</sub> )	Sink Current Rise Time	0.1 I <sub>L</sub> to 0.9 I <sub>L</sub> (3); (4)		0.1		μs

- 1) Sensing voltage can be -1 V for  $t \leq 50 \mu\text{sec}$ ; in steady state  $V_{\text{enmin}} \geq -0.5 \text{ V}$ .
- 2) See fig. 2.
- 3) See fig. 4.
- 4) The load must be a pure resistor.

Figure 1 : Typical Saturation Voltage vs. Output Current.

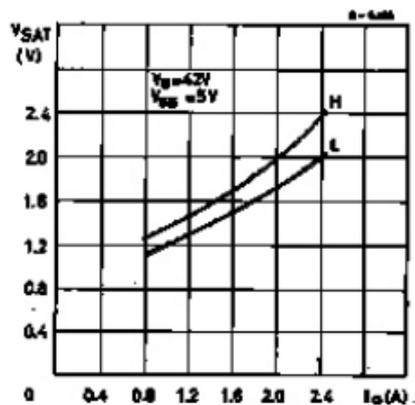


Figure 2 : Switching Times Test Circuits.

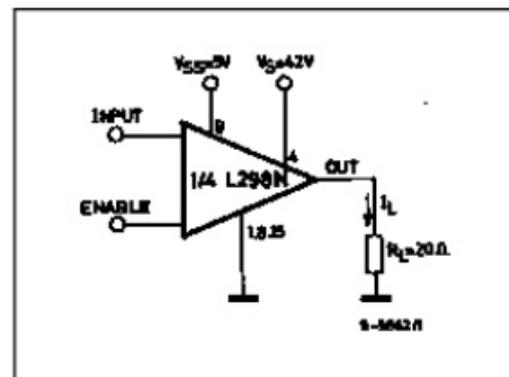
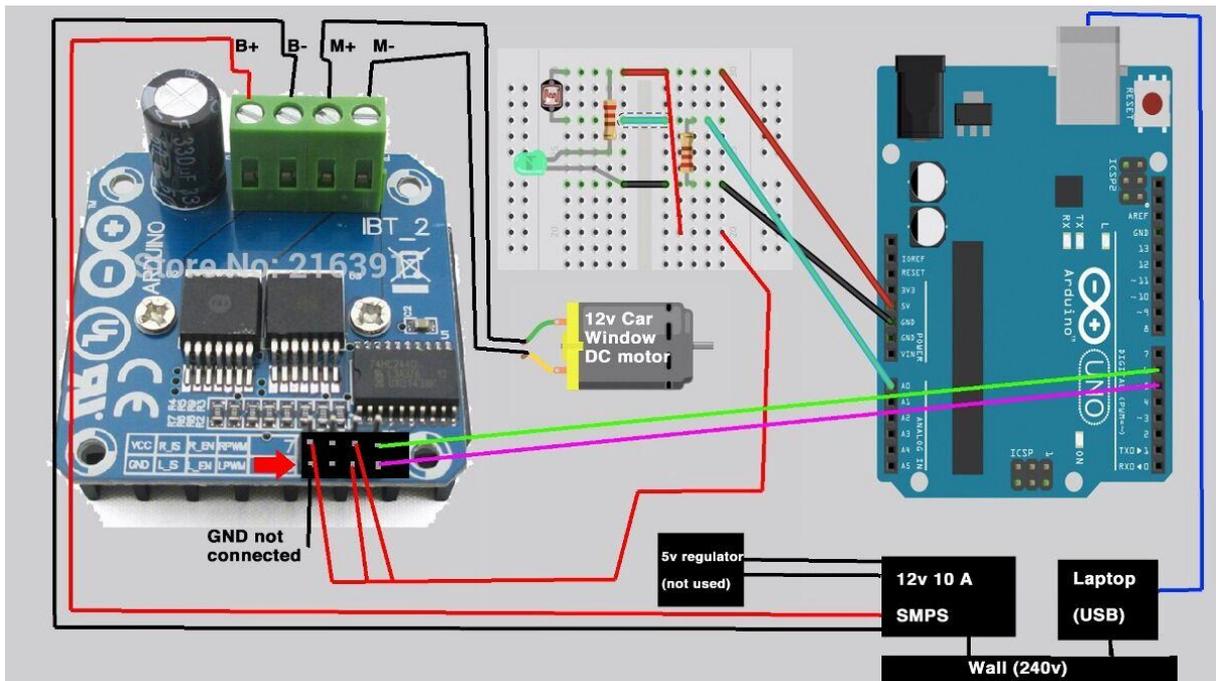


Figura 452: Características técnicas de la etapa de potencia L298N

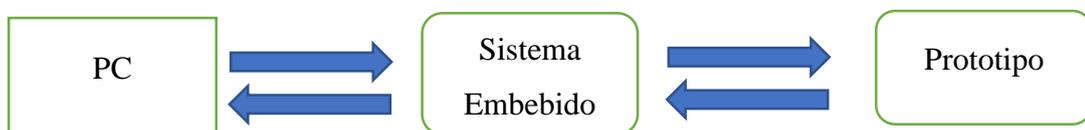


*Figura 463: Esquemático de ejemplo de un conexionado entre arduino, etapa de potencia y motor.*

En el diagrama esquemático anterior se muestra un ejemplo de conexionado de la etapa de potencia con Arduino y Motor.

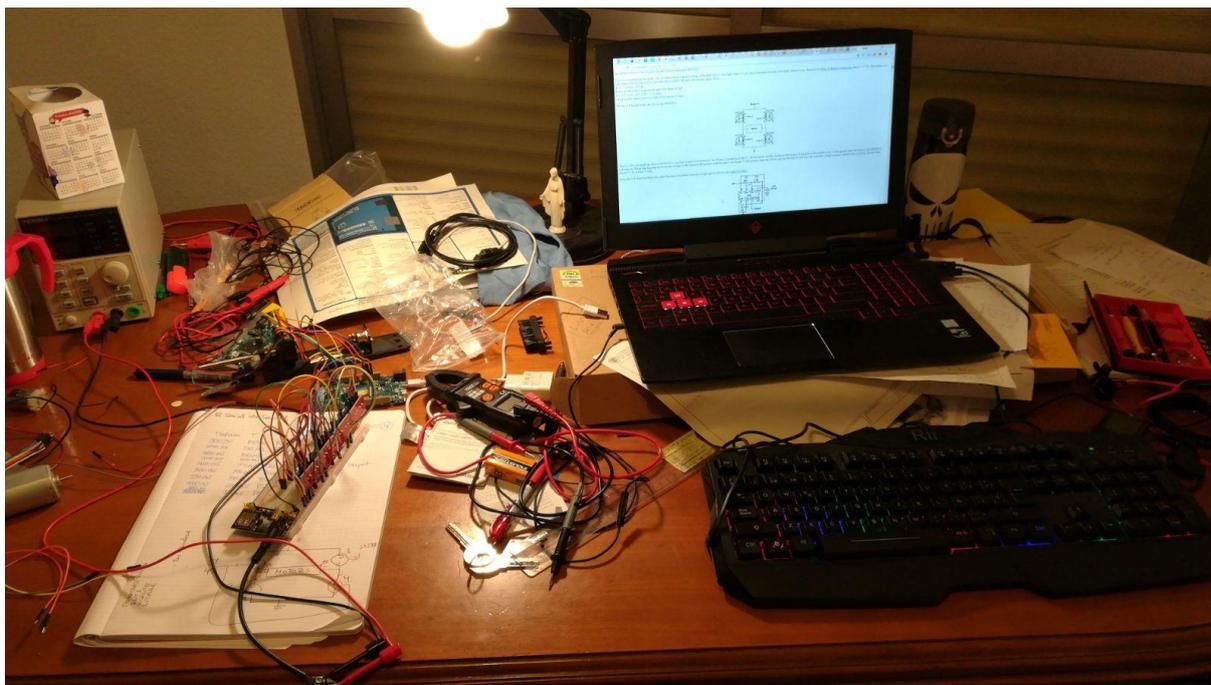
## 5. IMPLEMENTACIÓN DEL PENDULO CON VOLANTE DE INERCIA. CODIGO DE PROGRAMACION.

En nuestro sistema hemos realizado que el control del prototipo por la necesidad de velocidad de computo por debajo de 20 ms en el sistema embebido, y desde la computadora solamente enviamos las ordenes de tipo de trabajo y recibimos y graficamos los resultados obtenidos.

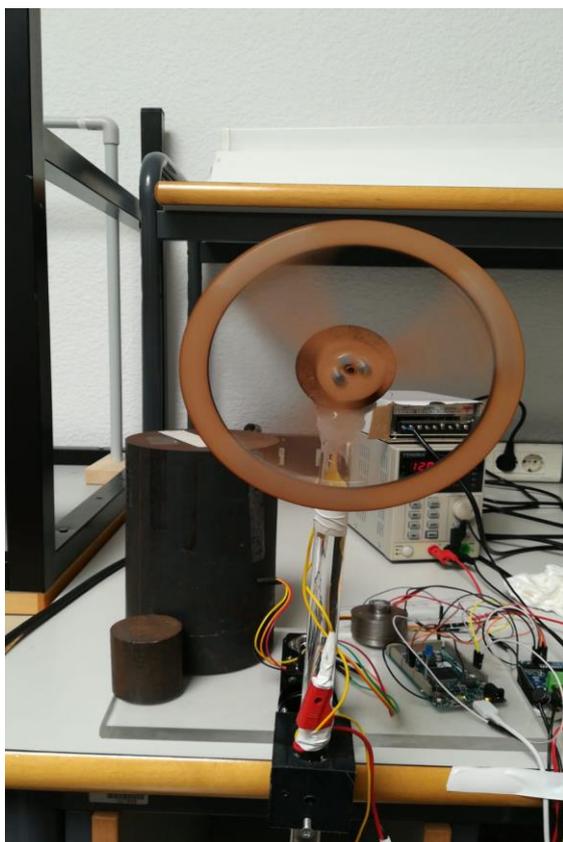


*Figura 474: Diagrama de interacción de nuestro sistema.*

En esta capítulo incluiremos el código completo de programación de nuestro microcontrolador para la lectura del encoder y control del motor de cc.

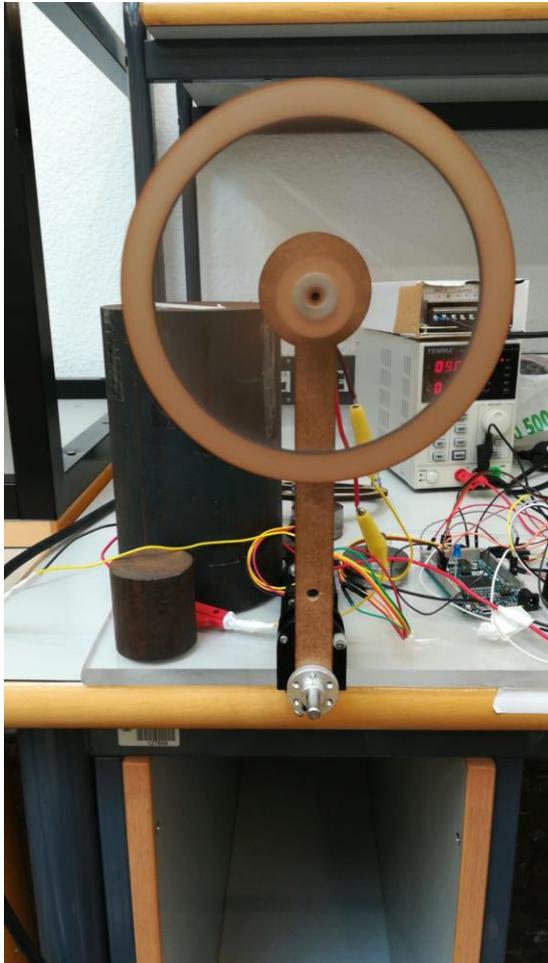


*Figura 485: Etapa de comienzo de construcción de prototipo físico.*



*Figura 496: Prototipo en funcionamiento*

Prototipo en funcionamiento en la etapa de seguimiento de referencia senoidal.



*Figura 507: Prototipo en calibración y pruebas.*

Imagen del prototipo en funcionamiento en la etapa de referencia cero.

### **5.1. Programa Utilizado.**

Programa Implementado en el sistema embebido.

```
//Este archivo mide el angulo y lo envia por bluetooth al que se conecte al htc05
//Este programa lo que hace es darme tanto en un sentido como en otro los grados del encoder.
//Los trasmite por bluetooth
//con este programa tenemos una resolucion de 12 bits velocidad/1024 y es la instruccion
analogWriteResolution(12);

#define MIN1 -2.5
#define MAX1 2.5

// Variables para Contro PID
int t1 = 0 ;
int t2 = 0 ;
```

```

int T = 20 ; //Este es el tiempo del bucle que hice mas bajo para que dure 10 ms, si supera prende el led //
Periodo de muestreo (milisegundos)
float Tm = T/1000.0 ; // Periodo de muestreo (segundos)

float Kp = 0.0 ; // Ganancia proporcional
float Kd = 0.0 ; // Ganancia derivada
float Ki = 0.0 ;

float ff=0.1 ;//frecuencia
//float amp=2.5 ;//amplitud
float off = 0.0 ;//offset
float ss=0.0 ;//senoidal

float ultimo_grado=180.0; // variable penduleo
float grados_pendolo; // los grados en el penduleo

float t = 0.0 ;
float R = 0.0 ; // referencia, el angulo 0° a conseguir

float Ei_1 = 0.0 ; // error anterior
float E = 0.0 ; // error actual
float Up = 0.0 ; // Control P
float Ud = 0.0 ; // Control derivativo
float U = 0.0 ; // control PD
float Ui = 0.0 ;
float Ui_1 = 0.0 ;

byte ang1=0 ;
byte ang2=0 ;

int PWMout;

// Variables BTS7960. Control de motor
#define cw 0 // giro a derechas
#define ccw 1 // giro a izquierdas
int sentido= ccw;
#define R_PWM 5 // pines
#define L_PWM 4 // pines
int limite_speed= (int) 255;

// Encoder
#define encoder0PinA 2 // Definimos los pines del pin A y PinB del Chanel del encoedrs.
#define encoder0PinB 3
volatile int encoder0Pos = 0; //esta variable esta en RAM, es requisito en las interrupciones si esta variable se
cambia.
volatile int ultimovalor = 0; //Este es el ultimo valor que mide
float grados=0;
//float grados0; //esta variable es la que se inicaliza apenas conecte el encoder es el angulo inicial va a ser el
correspondiente a 180 grados
//float gradMov=2.5;

void setup() {
  pinMode(encoder0PinA, INPUT_PULLUP); //definos los pines como entradas
  pinMode(encoder0PinB, INPUT_PULLUP);

  pinMode(R_PWM, OUTPUT);
  pinMode(L_PWM, OUTPUT);

```

```

//pinMode(5,INPUT_PULLUP);// Entrada potenciometro

// encoder pin on interrupt 0 (pin 2)//Interrupcion cero
attachInterrupt(digitalPinToInterrupt(encoder0PinA), doEncoderA, CHANGE);//Activamos las dos
interrepciones ver hoja

// encoder pin on interrupt 1 (pin 3)//Interrupcion uno
attachInterrupt(digitalPinToInterrupt(encoder0PinB), doEncoderB, CHANGE);//La interrupcion la haremos al
cambio

// Serial PORT
Serial1.begin (115200); // aca inicializo el monitor o puerto serie con el que voy a trabajar.monitor serial IDE
arduino

motorRun(255,ccw);

}

void loop() {

while (Serial1.available()>=2) {
  ss=(Serial1.read()*(MAX1-MIN1)/255.0)+MIN1 ;//Aca recibo si quieta o senoidal.Osea sin amplitud.
  //ss=-1*ss;
  off=(Serial1.read()*(MAX1-MIN1)/255.0)+MIN1 ;//Aca recibo el offset
  Serial1.flush() ;
}
//ss=amp*sin(2*3.14*ff*(t1/1000.0))+off ;
ss=ss+off ;

grados=(((encoder0Pos)*360.0/4096.0))+180;
// corregimos angulo pasando a negativo en sentido antihorario
if (grados>180) {grados= grados-360;
}

//utilizamos una segunda variable para el penduleo
grados_pendolo=grados;
if (grados_pendolo<0) {
  grados_pendolo=grados_pendolo+360;
}

ang1=(byte)((grados+180.0)*100/256) ;//Porque aca se le pone byte aca se pone la parte superior del dato en 8
bits
ang2=(byte)((grados+180.0)*100-ang1*256) ;//aca se coloca la parte inferior del dato en 8 bits
Serial1.write(ang1) ;
Serial1.write(ang2) ;
Serial1.flush() ;

// Calculo Accion de Control
Kp=0.90; //original en 0.90
Kd=0.05;//Cuando Use Kd este lo comento, originalmente en 0.05
Ki=0.40;//Estaba originakmente en 0.40

// CONTROL PID
E=ss-grados ; // Señal error

```

```

Up=E*Kp ; // Acción proporcional

Ui=(E*Tm+Ui_1) ; // Acción integral
if (Ui>2) { Ui=2 ; }
if (Ui<-2) { Ui=-2 ; }
Ui_1=Ui ;
Ui=Ui*Ki ;

Ud=(E-Ei_1)/Tm ; //E=error actual Ei_1 = Error anterior tm=periodo muestreo en segundos
Ei_1=E ; // Guardamos el valor anterior
Ud=Ud*Kd ; // Control derivativo

U=Up+Ud+Ui; //U=Up+Ui+Ud ; // Señal control

// If de control o penduleo
if (abs(grados)<26)//aca notifico en el angulo que quiero que empiece a trabajar el pid original en 26
{

    if (U>1.0) { //Serial.println("SAT +++") ;
        U=1.0;}
    if (U<(-1.0)) {
        // Serial.println("SAT ---") ;
        U=-1.0 ;}
    PWMout=map(abs(U)*100,0,100,0,255);

    if (U<0) {
        motorRun(PWMout,cw); // giro a izquierdas
    }

    if (U>0) {
        motorRun(PWMout,ccw); // giro a derechas
    }

    // Bluetooth
    //Serial1.print("*G");
    //Serial1.print((float)grados,4);//Imprimir con retorno de carro y cuatro decimales despues de la coma.y en
float(este muejstra los valores de los grados).
    //Serial1.print(",");
    //Serial1.print(Kp);
    //Serial1.println("*");

    // Serial USB-lora
    // Serial1.print("G=");
    // Serial1.print(grados);
    // Serial1.print(" ; E=");
    // Serial1.print(E);
    // Serial1.print(" ; U=");
    // Serial1.print(U);
    // Serial1.print(" ; P=");
    // Serial1.println(PWMout);
} else
{
    // Columpiar
    int velocidad=map(abs(grados),0,180,0,126);//Aca mapeo que a mayor angulo mayor velocidad y viceversa y
empiezo en 180-110(pero 126 paro)
    //7Serial.print("velocidad ");
    //Serial.print(velocidad);

    if (grados_pendulo>ultimo_grado){ // angulo decreciente
        motorRun(velocidad,ccw); // giro a derechas
    }
}

```

```

        ultimo_grado=grados_pendolo;
        // Serial.print("mayor ");
        // Serial.println(velocidad);
        // Serial.println(grados);
    }
else {
    motorStop();
}
    if (grados_pendolo<ultimo_grado){ // angulo decreciente
        motorRun(velocidad,cw); // giro a izquierdas
        ultimo_grado=grados_pendolo;
        // Serial.print("menor ");
        // Serial.println(velocidad);
        // Serial.println(grados_pendolo);
    }
    else {
        motorStop();
    }
}

// enviar grados por el puerto Serie
//Serial.println(grados);Este es el que leia en mi version
//Serial.print(",");
//Serial.println(U);

// Temporización del bucle de control para que realice en control en 10 ms y si no es que no controla y prende
el led 13
t2=millis() ;
if (t2-t1>T) {
    digitalWrite(13,HIGH) ;
    //Serial.print("*****");
}
else {digitalWrite(13,LOW) ;}
while (t2-t1<T) {t2=millis() ;}
t1=millis() ;

} // end loop

// Funciones
//////////

void doEncoderA() {
    // look for a low-to-high on channel A
    if (digitalRead(encoder0PinA) == HIGH) {

        // check channel B to see which way encoder is turning
        if (digitalRead(encoder0PinB) == LOW) {
            encoder0Pos = encoder0Pos + 1;    // CW//sentido horario
        }
        else {
            encoder0Pos = encoder0Pos - 1;    // CCW// Sentido antihorario
        }
    }
}

else // must be a high-to-low edge on channel A
{

```

```

// check channel B to see which way encoder is turning
if (digitalRead(encoder0PinB) == HIGH) {
  encoder0Pos = encoder0Pos + 1;    // CW
}
else {
  encoder0Pos = encoder0Pos - 1;    // CCW
}
}
//Reiniciamos el contador de pulsos una vez a dado una vuelta completa 4096
if ((encoder0Pos == 4096)|| (encoder0Pos == -4096))
{
  encoder0Pos=0;
}
ultimovalor=encoder0Pos;
} //Fin de la funcion doencoder A

void doEncoderB() {
  // look for a low-to-high on channel B
  if (digitalRead(encoder0PinB) == HIGH) {

    // check channel A to see which way encoder is turning
    if (digitalRead(encoder0PinA) == HIGH) {
      encoder0Pos = encoder0Pos + 1;    // CW
    }
    else {
      encoder0Pos = encoder0Pos - 1;    // CCW
    }
  }

  // Look for a high-to-low on channel B

  else {
    // check channel B to see which way encoder is turning
    if (digitalRead(encoder0PinA) == LOW) {
      encoder0Pos = encoder0Pos + 1;    // CW
    }
    else {
      encoder0Pos = encoder0Pos - 1;    // CCW
    }
  }
  //Reiniciamos el contador de pulsos una vez a dado una vuelta completa 4096
  if ((encoder0Pos == 4096)|| (encoder0Pos == -4096))
  {
    encoder0Pos=0;
  }
  ultimovalor=encoder0Pos;
}

void motorRun(int speed,int direction)
{
  if (speed>limite_speed) {
    speed=limite_speed;
  }
  if (direction==cw) {
    analogWrite(L_PWM,0);
    analogWrite(R_PWM,speed);
  }

  if (direction==ccw) {
    analogWrite(R_PWM,0);
  }
}

```

```

    analogWrite(L_PWM,speed);
  }
}

void motorStop()
{
  analogWrite(L_PWM,0);
  analogWrite(R_PWM,0);
}

```

## 5.2. Graficas y resultados obtenidos.

En la grafica a continuación podemos ver dos etapas en el comportamiento del prototipo, la linea roja es la referencia del mismo.

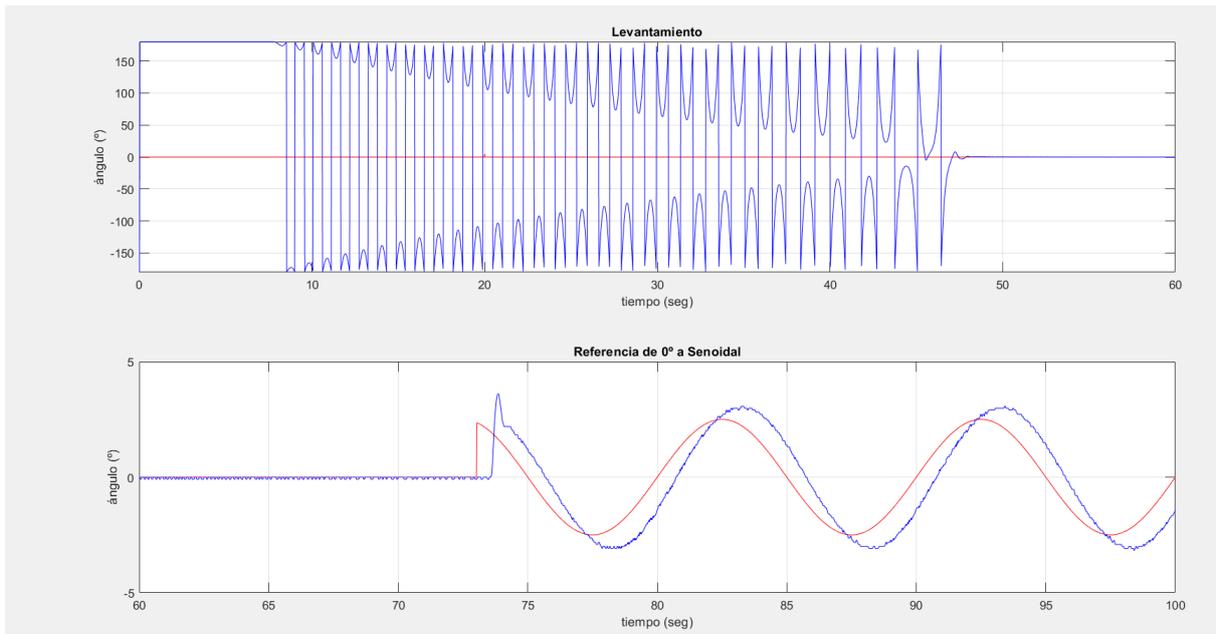
EN la primer grafica observamos el levantamiento del prototipo puesto que en el inicio se encuentra en equilibrio estable en reposo en su posición bas baja, y a medida que comenzamos a mover el volante y siguiendo su precuencia natura el pensulo se comienza a balancear para conseguir llegar a la señal de referencia de  $0^\circ$  posición de control del péndulo en equilibrio inestable.

EN la grafica podemos ver como el péndulo se balancea aumentando su angulo, esto es debido a que hamacamos o provocamos el movimiento del péndulo controlando la aceleración y desaceleración hacia un sentido y otro del motor, y por ende del volante de inercia.

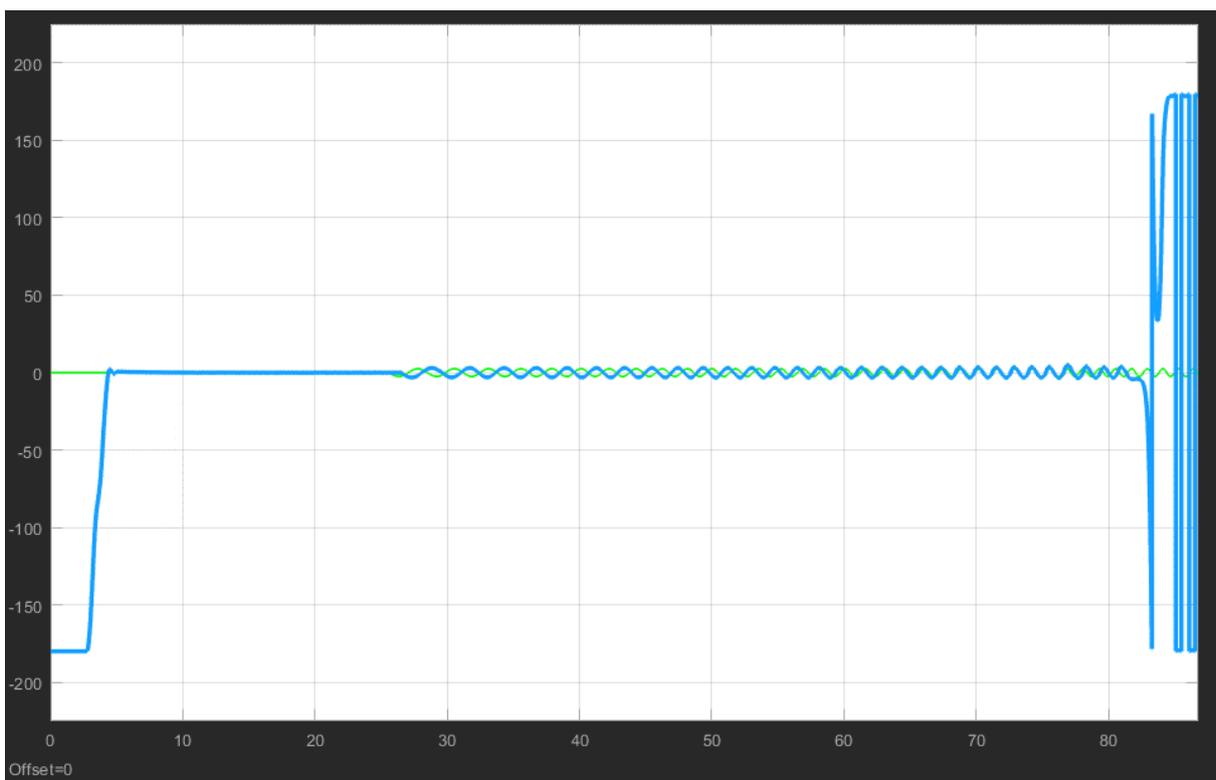
A medida que el péndulo va acercandoe al angulo de referencia  $0^\circ$ , el contro se conmuta pasando ahora a ser el control PID el encargado de mantener el equilibrio inestable del sistema.

De este comportamiento podemos ver la grafica del mismo.

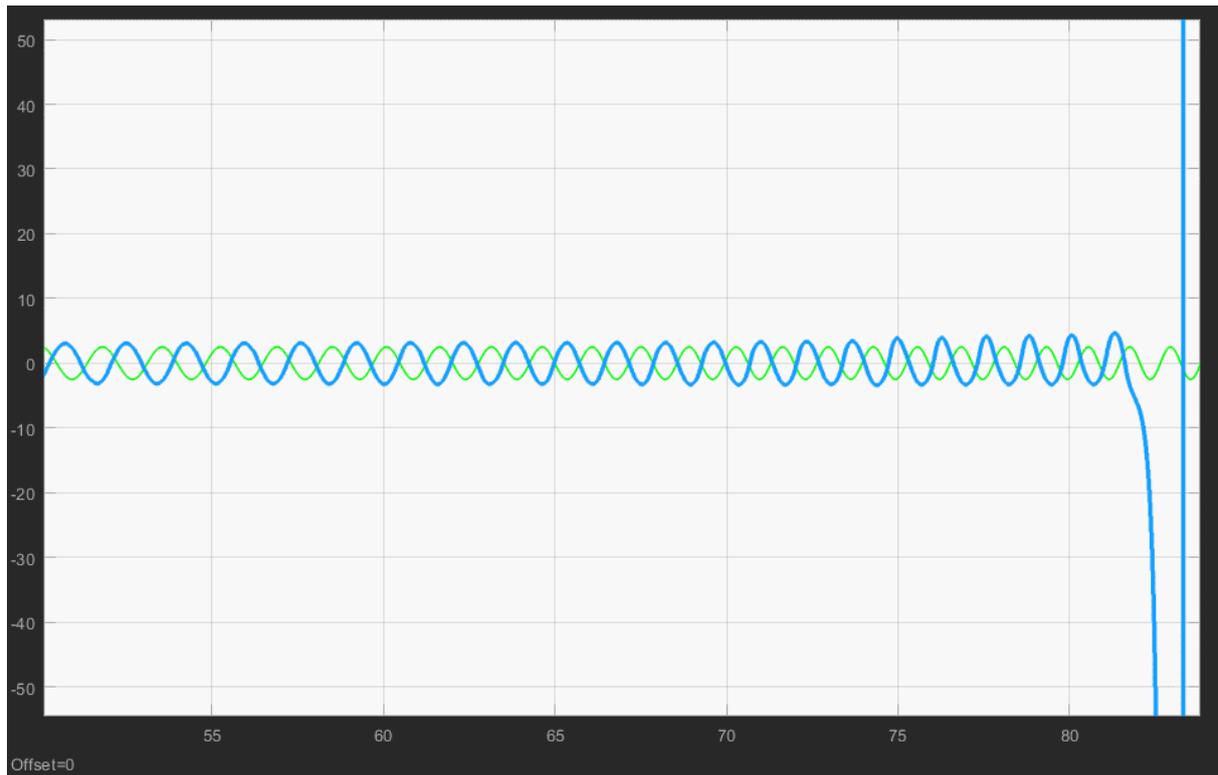
En la grafica ingerior vemos cuando conmutamos el sistema cuando se encuentra en equilibrio inestable con referencia en  $0^\circ$  y posición en  $0^\circ$  también, y provocamos en la referencia una señal senoidal de 10 segundo de periodo, entonces podemos observar como la línea azul correspondiente al comportamiento del prototipo comienza a seguirlo, teniendo una diferencia 0.5 segundos de retraso con respecto a la referencia y una diferencia en su amplitud de  $0.5^\circ$ , esto es debido a la inercia del sistema.



**Figura 518: Comportamiento del Levantamiento hasta referencia  $0^\circ$  y posteriormente seguimiento de senoidal.**



En esta grafica se puede observar como el sistema una vez que se encuentra estabilizado comienza a seguir una referencia senoidal que va incrementando su frecuencia.



**Figura 529: Seguimiento de referencia sinusoidal variable hasta perder la referencia.**

En la grafica anterior se puede observar, don de la línea amarilla es una señal sinusoidal de 10 segundos de periodo, que va desde 2,5 s de periodo hasta 1.25 seg de periodo, acelerando hasta el doble de su frecuencia inicial. La amplitud de esta referencia es de 2,5°.

Vemos que la lectura de nuestro prototipo evidencia que comienza a seguir la referencia sinusoidal con un retraso debido a la inercia del sistema y que a medida que vamos aumentando la frecuencia se va retrasando puesto que le es imposible copiar el comportamiento de la señal de referencia y vemos como a medida que aumenta su retraso también va aumentando su amplitud, hasta que el sistema pierde completamente el contro se desestabiliza y saliendo de su equilibrio inestable se cae por efecto de la gravedad.

### 5.3 Ajuste de parámetros del regulador

Con el fin de sintonizar el control del sistema es normal que se inicie el proceso de sintonía con la acción integral y derivativa en cero. Ajustar el nivel de P lentamente hasta que el pendulo empiece a oscilar alrededor de su posición de balance, P no debe ser excesivamente grande.

Luego se debe incrementar la acción integral lentamente hasta que el robot empiece a oscilar nuevamente y se acelere cuando pierda el equilibrio. Finalmente, con los dos parámetros ya ajustados, incrementar lentamente la acción derivativa hasta que el sistema se estabilice y no exista un grado considerable de oscilación.

De este modo, se han obtenido las siguientes constantes:  $K_p=0.90$ ; //original en 0.90  
 $K_d=0.05$ ; //Cuando Use  $K_d$  este lo comento, originalmente en 0.05  
 $K_i=0.40$ ; //Estaba originakmente en 0.40

## 6. CONCLUSIONES Y PROPUESTAS A FUTURO.

Al final de esta etapa de proyecto se puede destacar, la modalidad de trabajo empleada, la cual consistio en aplicar los conocimientos adquiridos a lo largo del Master.

Los conocimientos adquiridos a destacar fueron:

- Cálculo y Selección de componentes electrónicos y mecánicos
- Diseño en CAD en Siemen Nx y Solid Works
- Diseño de sistema de control y simulación en Matlab Simulink.
- Programacion de Arduino.
- Conexión de Arduino con PC para extraer datos y obtener graficas.

Propuestas a futuro:

Se diseño un prototipo con dos volantes de inercia, además se ulizo un modulo electrónico con giroscopio,acelerómetro para determinar orientacio de una barra en posición vertical y mantener su equilibrio.Este problema se planteo como una segunda etapa, sibien se construyo el prototipo, con sus componentes, se simulo, no se pudo lograr mantener en equilibrio del mismo por falta de tiempo.

## 7.BIBLIOGRAFIA

[1]-J. Á. Acosta, “Furuta’s Pendulum: A Conservative Nonlinear Model for Theory validation and Practise,” Math. Probl. Eng., vol. 2010, pp. 1-29, 2010.

[2]- D. Pagano, L. Pizarro, and J. Aracil, "Local bifurcation analysis in the Furuta Pendulum via normal forms," Int. J. Bifurcation Chaos, vol. 10, no. 5, pp. 981-995, May. 2000.

[3]- Control de un pendulo invertido, Cristian Guajardo Yevenes, Alumno Departamento de Electronica, Universidad Tecnica Federico Santa Maria, 05 de mayo de 2004.

[4] SPONG, Mark W. et al. "Nonlinear Control of the Inertia Wheel Pendulum". University of Illinois. Estados Unidos, 2000.

### **Páginas web de los componentes empleados:**

Regulador DC/DC: <https://www.pololu.com/product/2122>

Bluetooth: <http://www.electronicaestudio.com/docs/istd016A.pdf>

Motores: <https://www.pololu.com/product/2823>

Brackets: <https://www.pololu.com/product/1084>

Ruedas: <https://www.pololu.com/product/1435>

Hub: <https://www.pololu.com/product/1083>

Metacrilato: <https://www.mwmaterialsworld.com/es/plancha-de-metacrilato-transparente-cristal.html>

### **Herramientas en la Programación:**

Lauszus, Kristian. 2012. Kalman Filter. Available at: <https://github.com/TKJElectronics/KalmanFilter> [Accessed 20 Mars 2015]

## 8. ANEXOS

### 8.1. Presupuestos:

Combinaciones				
<b>1. Combinación</b>				
Unidad motriz	Precio unitario	Cantidad	Precio total	
<b>Motor</b> DC-MAX26S GB KL 24V <b>Sensor</b> ENX16 EASY 1024IMP N° de artículo: B789C0B0FD95  Certificado de Conformidad: <b>No</b>	€186,00	1	<b>€186,00</b>	
<a href="#">Realizar pedido nuevamente</a>				
Importe parcial (montado)	<b>€186,00</b>	1	<b>€186,00</b>	
 <p>El producto no está disponible como tal en el almacén, pero sus componentes sí están disponibles. Procederemos al envío del paquete en el plazo de 11 días laborables. Plazo de entrega previsto: hasta 15 días laborables.</p>				
Importe parcial 1. Combinación				<b>€186,00</b>
<b>Total carro de la</b>			<b>€186,00</b>	
<b>compra</b>				
Costes de envío y embalaje			€25,36	
<b>Total</b>			<b>€211,36</b>	

Presupuesto Motor Maxon.

**A:****Para:** FABIO ANDRES ROCCHI**Asunto:** Aviso de llegada sobre la importacion procedente de: SUIZA**Conocimiento aereo / Albaran (AWB):** 789956586170**Proveedor:** MAXON MOTOR AG

Estimado cliente,

le informamos que una vez realizado el despacho de la mercancia declarada en este envío, los impuestos, tasas y gastos administrativos originados por el mismo son los siguientes:

Partida Arancelaria:	8501101090
Arancel:	
IVA: 21,00	44,31
Recargo Equivalencia:	
IIIE y Otros:	
Gastos Administrativos:	16,34
<b>Total:</b>	<b>60,65</b>

\* Si usted es un importador acogido en el procedimiento previsto en el artículo 167.bis segundo párrafo, de la Ley 37/1992 del IVA, la cantidad a aceptar en esta preliquidación es de [0,00] (IVA Devengado no incluido)

\*\* En casos excepcionales la factura pro-forma podría variar ligeramente debido a los cambios de divisa aplicados generalmente los días 1 de cada mes. \*\*

## Gastos de importacion de motor maxon

Para: frocchi77@hotmail.com

Nota: por favor no responda a este mensaje. Si tiene alguna consulta sobre su pedido, contáctenos a [atencion.cliente@rs-components.com](mailto:atencion.cliente@rs-components.com)

**Confirmación de Pedido** Su número de pedido: **144197855**

Estimado Fabio Andres Rocchi,

A través de este email le confirmamos los detalles de su pedido.

Puede consultar el estado de su pedido en la sección [Mis Pedidos](#) en nuestra página [es.rs-online.com](http://es.rs-online.com).

Artículos comprados:		Nº de pedido RS: 144197855
HEDS-9040J00,Optical encoder,3ch,1024CPR Código RS: 7967874 Cantidad: 1 Sus números de referencia: Centro de coste: Almacén de origen: GB	<b>Fecha de entrega:</b> 1 Entrega el día 16/05/2018	27,61 €
Total del Pedido:		27,61 €
Gastos de Envío		5,00 €
<b>Total del Pedido (IVA excl.):</b>		<b>32,61 €</b>

Datos de la entrega	
Dirección de Envío ROCCHI FABIO ANDRES C/ Esteban Ballester, 22 - Puerta 8 Valencia 46022 ESPAÑA	Suministrador: RS Components Avenida de Europa,19 28224 Pozuelo de Alarcón Madrid España
Código de Empresa: 14620034	A la atención de: Fabio Rocchi

	701 Brooks Ave South, PO Box 677 Thief River Falls, MN 56701-0677	www.digikey.es FREEPHONE 900-983-183 FAX +31 53-484-9583	Invoice # 62909956 Completed Salesorder EURO €			
<b>Bill To:</b> FABIO ANDRES ROCCHI ESTEBAN BALLESTER N°22 PUERTA N°8 VALENCIA, VALEN 46022 SPAIN	<b>Ship To:</b> FABIO ANDRES ROCCHI ESTEBAN BALLESTER N°22 PUERTA N°8 VALENCIA, VALEN 46022 SPAIN	<b>Salesorder / Packlist:</b> 54992755 / DL1 <b>Customer:</b> 10993546 <b>Payment Terms:</b> MASTERCARD / 1005 <b>Shipping Method:</b> USAV <b>Tracking #:</b> 1Z5674320405933596 <b>Order Source:</b> INTERNET <b>Order Date:</b> 19-May-2018 <b>Invoice Date:</b> 20-May-2018 <b>Ship Date:</b> 21-May-2018 <b>Document Date:</b> 21-May-2018/AUTO				
<b>Buyer:</b> FABIO ANDRES ROCCHI ESTEBAN BALLESTER N°22 PUERTA N°8 VALENCIA, VALEN 46022 SPAIN	<b>Ship From:</b> DIGI-KEY 701 BROOKS AVE. SOUTH P.O. BOX 677 THIEF RIVER FALLS MN 56701-0677					
<b>Line Item</b>	<b>Ordered</b>	<b>Cancelled</b>	<b>Shipped</b>	<b>Item Number/ Description</b>	<b>Unit Price EURO €</b>	<b>Amount EURO €</b>
1	1	0	1	PART: 516-3507-ND MFG : Broadcom Limited / HEDM-5600#J06 COO : MALAYSIA LEAD FREE ROHS COMP REACH UNAFFECTED Jul-2017	DESC: KIT ENCODER 2CH 1024CPR 1/4* HTSUS: 8543.70.4500	46.72000 46.72
Total Invoiced 46.72 Shipping charges applied 18.00 ** Charges subtotal ** 64.72 Total charged to credit card 64.72 EURO €						
Incoterm 2010: DDP VALENCIA, VALEN, SPAIN						
* NO EEI 30.37(a) * No license required * These items are controlled by the U.S. Government and authorized for export only to the country of ultimate destination for use by the ultimate consignee or end-user(s) herein identified. They may not be resold, transferred, or otherwise disposed of, to any other country or to any person other than the authorized ultimate consignee or end-user(s), either in their original form or after being incorporated into other items, without first obtaining approval from the U.S. Government or as otherwise authorized by U.S. law and regulations.						
<b>Box(es):</b>						
1	USAV	1	1	1Z5674320405933596 516-3507-ND	0.17 kg / 6 oz	

**Precio encoder incremental optico.**

### Información de envío

---



JZK BTS7960B 43A Alto Voltaje Doble motor paso a paso DC  
Tarjeta del módulo del conductor Puente H PWM Para Smart  
Car **Arduino**  
Vendido por: JZK Express Network

**EUR  
11,79**

---

Importe de los productos: EUR 11,79  
Envío y manipulación: EUR 0,00

**Importe total del envío (IVA incluido si corresponde): EUR  
11,79**

Pagado por Mastercard: EUR 11,79

**Etapas de Potencia**

Pedido: **475285** realizado el 02/05/2018 21:37:07

Forma de Pago: **Redsys**

Referencia	Producto	Precio Unidad	Cantidad	Precio Total
ROB-0035	Doble acople giratorio con rodamiento	8,20€	2	16,40€
ROB-0029	Acople de tornillo - 5mm	5,30€	1	5,30€
ROB-0028	Acople de tornillo - 1/4"	5,20€	1	5,20€
DCM-0025	Soporte de eje 3 mm / M3	6,90€	1	6,90€
DCM-0026	Soporte de eje 5 mm / M3	6,90€	1	6,90€
DCM-0027	Soporte de eje 6 mm / M3	7,50€	2	15,00€
ROB-0071	Acople de presión - 5mm	7,95€	1	7,95€
ARD-0116	Arduino Industrial 101	35,00€	1	35,00€
Productos				119,36€
Descuentos				0,00€
Envíos				5,99€
<b>TOTAL</b>				<b>125,35€</b>

**Método de envío**

## Acoples y Arduino Due.

Pedido: **475134** realizado el 26/04/2018 18:44:35

Forma de Pago: **Redsys**

Referencia	Producto	Precio Unidad	Cantidad	Precio Total
DCM-0025	Soporte de eje 3 mm / M3	6,90€	2	13,80€
DCM-0026	Soporte de eje 5 mm / M3	6,90€	1	6,90€
DCM-0027	Soporte de eje 6 mm / M3	7,50€	1	7,50€
Productos				34,13€
Descuentos				0,00€
Envíos				5,99€
<b>TOTAL</b>				<b>40,12€</b>

**Método de envío**

**Computo Y Presupuesto del modelo**

<b>Item</b>	<b>Precio</b>	<b>Cantidad</b>		<b>Total</b>
<b>Cortes de DM segun Planos</b>	<b>40</b>	<b>1</b>		<b>40</b>
<b>Metacrilato</b>	<b>80</b>	<b>1</b>		<b>80</b>
<b>Motor Maxon</b>	<b>270</b>	<b>1</b>		<b>270</b>
<b>Bridas de aluminio</b>	<b>40</b>	<b>1</b>		<b>40</b>
<b>Arduino Due</b>	<b>35</b>	<b>1</b>		<b>35</b>
<b>Driver</b>	<b>12</b>	<b>1</b>		<b>12</b>
<b>Tornillos</b>	<b>15</b>			<b>15</b>
<b>Cables de coneccion</b>	<b>5</b>			<b>5</b>
<b>Encoder</b>				<b>65</b>
<b>Total</b>				<b>562</b>