

E.T.S. de Ingeniería Industrial,
Informática y de Telecomunicación

Control de posición de dedos de una mano robótica impresa en 3D con Arduino Due



Grado en Ingeniería
en Tecnologías de Telecomunicación

Trabajo Fin de Grado

Daniel Cerezo San Martín

Carlos Aristóteles De La Cruz Blas

Pamplona, 18 de Junio de 2018

RESUMEN

El objetivo de este trabajo de Fin de Grado se ha fijado en el aprendizaje de las diferentes formas de control existentes a la hora de controlar la posición de los movimientos de una mano artificial, que podría utilizarse a forma de prótesis en un futuro. Para cumplir dicho objetivo se ha planteado también la fabricación de un prototipo mecánico con capacidad de movimiento.

Por un lado se estudiarán las múltiples opciones que se pueden implementar en para la realización de un control, tanto en lazo abierto como en lazo cerrado. Tras el debido estudio se realizarán las pruebas para controlar de cada forma estudiada nuestro prototipo.

Por otro lado se ha realizado un prototipo mecánico de una mano robótica mediante la utilización de la técnica de impresión en 3D. A este prototipo se le ha dotado de los elementos necesarios para generar un eficiente sistema de control de movimiento el cual lo gobernará un microcontrolador Arduino DUE.

ABSTRACT

The aim of this Final Degree project has been to learn the different forms of control that exist when controlling the position of the movements of an artificial hand, which could be used as a prosthesis in the future. To achieve this objective, the manufacture of a mechanical prototype with movement capacity has also been considered.

On the one hand, the multiple options that can be implemented in order to carry out a control will be studied, both in open loop and in closed loop. After the due study, the tests will be carried out to control our prototype in each studied form.

On the other hand, a mechanical prototype of a robotic hand has been made by using the 3D printing technique. This prototype has been equipped with the necessary elements to generate an efficient motion control system which will be governed by an Arduino DUE microcontroller.

PALABRAS CLAVE:

Arduino Due, Arduino UNO, mano, prótesis, robótica, impresión 3, control, posición

KEYWORDS:

Arduino Due, Arduino UNO, hand, prosthesis, robotics, impression 3, control, position

Índice

RESUMEN	2
ABSTRACT	2
PALABRAS CLAVE:.....	2
KEYWORDS:	2
Índice de Figuras	5
Índice de Ecuaciones	7
1. Introducción	8
1.1. Motivación	8
1.2. Objetivos	9
1.3. Estructura del documento.....	10
2. Estado del Arte de las Manos Protésicas	11
2.1. Prótesis Antiguas.....	11
2.2. Manos protésicas accionadas por el cuerpo	13
2.2.1. Prótesis a finales del siglo XX	15
2.2.2. Prótesis Modernas: Las secuelas de la Segunda Guerra Mundial.....	17
2.3. Manos Protésicas Alimentadas de Forma Externa.....	18
2.3.1. Manos Protésicas Neumáticas	18
2.3.2. Manos Protésicas Mioeléctricas.....	19
2.4. Prototipos Actuales Destacados.....	20
2.4.1. Recopilación de Proyectos de Bajo Nivel de Complejidad	21
2.4.2. Proyecto Ámbito Empresarial: Shadow Dextrous Hand C6M	22
2.4.3. Proyecto Investigación: Dextra	23
3. Descripción del Sistema Propuesto.....	24
3.1. Descripción General	24
3.2. Mecánica	26
3.3. Arquitectura Electrónica	29
3.3.1. Actuadores	30
3.3.2. Sensor.....	32
3.3.3. Etapa de Potencia.....	33
3.3.4. Controlador	34
4. Electrónica de Control: Control de Posición de un Motor DC.....	36
4.1 Tipos de Control: Lazo Abierto y Lazo Cerrado	36
4.2 Tipos de Control en Lazo Cerrado	38

4.2.1	Control SI/NO (ON/OFF).....	38
4.2.2	Control Proporcional	39
4.2.3	Control Proporcional Integrativo	41
4.2.4	Control Proporcional Derivativo.....	43
4.2.5	Control Proporcional Integrativo Derivativo (PID)	46
4.3	Teoría Aplicada al Prototipo.....	49
5	Resultados Experimentales	50
5.1	Montaje de la electrónica de la Prótesis.....	50
5.2	Lazo abierto.....	54
5.3	Lazo Abierto con Encoder	57
5.3.1	Elección de salida correcta	57
5.3.2	Cálculo de $H(z)$ del Sistema en Lazo Abierto	63
5.4	Lazo Cerrado	67
5.4.1	Explicación del Software y Hardware del Control.....	67
5.4.2	Control ON-OFF	77
5.4.3	Ajustes PID (Control Proporcional).....	82
5.4.4	Control Proporcional Integrativo	86
5.4.5	Control Proporcional Derivativo.....	89
5.4.6	Control Proporcional Integrativo Derivativo (PID)	90
5.4.7	Estudio de Polos y Ceros según las variables PID.....	91
5.5	Control de Prótesis Completa	94
6	Conclusiones.....	96
6.1	Metodología de Diseño	96
6.2	Ventajas/Desventajas.....	97
6.3	Puntos de Mejora.....	98
	Bibliografía	99

Índice de Figuras

Figura 2.1 - Grabado de la mano de Götz von Berlichingen (Izq) y el retrato y firma de su dueño (dcha).....	12
Figura 2.2 - Esquema de Prótesis de Mano Activa realizada por Paré.....	13
Figura 2.3 - Distintos tipos de Arnesees	14
Figura 2.4 - Prótesis de Dorrance a) y b) /Prótesis de Trautman c)	16
Figura 2.5 - Prótesis de Mano Moderna.....	17
Figura 2.6 - Posición de motores en distintos prototipos	21
Figura 2.7- Shadow Dexterous Hand E1 Series [5]	22
Figura 2.8 - (Izq) Proyecto Dextra (Dcha) Detalle del Interior del Prototipo.....	23
Figura 3.1 - Esquema General de Nuestro Sistema	25
Figura 3.2 - Comparación Estructura Ósea con Nuestro Prototipo [8]	27
Figura 3.3 - Vista general del Prototipo.....	28
Figura 3.4 – Arriba, Mecanismo de Movimiento para un Pulgar Oponible. Abajo, Comparación entre el Módulo de Dedo Normal y el Módulo del Pulgar.	29
Figura 3.5 - Motor Pololu escogido para el prototipo [9]	31
Figura 3.6 - Servomotor utilizado en prototipo.....	31
Figura 3.7 - Codificador Pololu escogido para el prototipo [10]	32
Figura 3.8 – (Izq) Driver preseleccionado para el circuito de control [11] (Dcha) Controlador de Circuito L293D [12]	33
Figura 3.9 - Tabla Comparativa entre Arduino UNO y DUE [13]	34
Figura 3.10 - (Izq) Arduino UNO y (Dcha) Arduino DUE [14] [15]	35
Figura 4.1 - Control en Lazo Abierto.....	37
Figura 4.2 - Control en Lazo Cerrado	37
Figura 4.3 - Diagrama de Bloques de Controlador Proporcional [20]	39
Figura 4.4 - Salidas del Sistema con diferentes K_p [21].....	40
Figura 4.5 - Izq: Respuesta del Sistema con una K_p baja//Dcha: Respuesta del Sistema con una K_p alta [22].....	41
Figura 4.6 - Diagrama de Bloques de Controlador Proporcional Integrativo [20]	41
Figura 4.7 - Salidas del Sistema con diferentes K_i [23].....	43

Figura 4.8 - Diagrama de Bloques de un Sistema de Control Proporcional Derivativo [20]	44
Figura 4.9 - Explicación Control Derivativo en un Instante de tiempo.....	45
Figura 4.10 - Salidas de un Sistema de Control Proporcional Derivativo	46
Figura 4.11 - Diagrama de Bloques de un Controlador PID [20]	47
Figura 5.1 - Solución del Bus.....	51
Figura 5.2 - Resultado final del Bus	52
Figura 5.3 - Izq: Funcionamiento Incorrecto del Sistema. Drch: Funcionamiento Deseado del Sistema	52
Figura 5.4 - Error en Bus	53
Figura 5.5 - Funcionamiento Correcto del Sistema	53
Figura 5.6 - Esquemático Experimento 1.....	54
Figura 5.7 - Distribución de Pines Driver	55
Figura 5.8 - Gráfica Contador y Dirección del Lazo Abierto	56
Figura 5.9 – Teoría de Sistemas.....	57
Figura 5.10 - Entrada al Sistema	58
Figura 5.11 - Salida del sistema Sin Mecanismo.....	59
Figura 5.12 - Salida del Sistema con Mecanismo y Gomas Degradadas	60
Figura 5.13 - Salida del Sistema con Mecanismo y Gomas no Degradadas	62
Figura 5.14 – Input (abajo) y Output (arriba) Introducidos en el Programa	64
Figura 5.15 - Información del Sistema Escogido.....	65
Figura 5.16 - Diagrama de Bloques de L293D [26]	68
Figura 5.17 - Esquema del Circuito con L293D	69
Figura 5.18 - Nombre de Pines del Codificador.....	70
Figura 5.19 - Código Decisor de Acción de Motor [27]	72
Figura 5.20 – Explicación Funcionamiento del Encoder.....	73
Figura 5.21 - Código de la Función cuando se provoca una Interrupción [27]	74
Figura 5.22 - Esquema Versión 2	75
Figura 5.23 - Implementación Real Versión 3	75
Figura 5.24 - Cambio Realizado en Fichero	76
Figura 5.25 - Montaje Experimento ON-OFF.....	77
Figura 5.26 - Comparación Señal Digital con Analógica	80

Figura 5.27- Sobreoscilación debido a ajuste PID en tramo A	81
Figura 5.28 - Sobreoscilación debido a ajuste PID en tramo C.....	81
Figura 5.29 - Resultado Experimental Control de Posición	82
Figura 5.30 - Slew-Rate en Control PID [28]	83
Figura 5.31 - Cálculo del Slew-Rate [29].....	84
Figura 5.32 - Resultados Experimentales Control Proporcional Zoom	85
Figura 5.33 - Flexión con Control PI con $K_p=0.5$ y K_i variable	87
Figura 5.34 – Zoom de Flexión con Control PI con $K_p=0.5$ y K_i variable	88
Figura 5.35 - Flexión con Control PD con $K_p=0.5$ y K_d variable.....	89
Figura 5.36 - Funcionamiento Final del PID.....	91
Figura 5.37 - Estudio Polos y Ceros Control Proporcional.....	92
Figura 5.38 - Estudio Polos y Ceros Control Proporcional Integrativo	93
Figura 5.39 - Estudio Polos y Ceros Control Proporcional Derivativo	93
Figura 5.40 - Esquemático de Prótesis Completa.....	94
Figura 5.41 - Modelo Construido Sujetando un Objeto	95

Índice de Ecuaciones

Ecuación 1 – Ecuación de Sistema de Control Proporcional	39
Ecuación 2 – Ecuación de Sistema de Control Integrativo	42
Ecuación 3 - Ecuación de Sistema de Control Proporcional Derivativo	45
Ecuación 4 - Ecuación de la constante Derivativa	45
Ecuación 5 - Ecuación de Sistema Controlador Proporcional Integrativo Derivativo	47
Ecuación 6 - Ecuación de la Transformada Z Unilateral	48
Ecuación 7 - Calculo Slew-Rate del Control Proporcional	84
Ecuación 8 - Cálculo Slew-Rate del Control Proporcional Integrativo	87
Ecuación 9 - Cálculo Slew-Rate del Control Proporcional Derivativo.....	89
Ecuación 10 - Cálculo Slew-Rate del Control Proporcional Integrativo Derivativo.....	90

1. Introducción

1.1. Motivación

Nuestras manos son el principal órgano del cual dispone el ser humano para la manipulación física del medio en el que nos encontramos. Se tratan de nuestras principales herramientas de trabajo, estas son la fuente de información táctil sobre nuestro entorno más importante, llegando a ser una de las zonas de nuestro cuerpo con mayor número de terminaciones nerviosas.

En estos momentos, gracias a la robótica, el ser humano ha sido capaz de crear autómatas que nos facilitan la vida a los seres humanos. Las manos robóticas, sin duda, son uno de estos sistemas destinados a mejorar la vida humana. Bien sea en un entorno industrial como en su uso protésico.

Debido a la importancia de las manos anteriormente explicada, muchas personas que pierden estas extremidades, bien sea por accidentes o deformaciones, sufren una clara desventaja en su vida diaria. Una desventaja que podría ser suplida con una prótesis robótica, no obstante se ha de ser precavido en el diseño de la misma, pues el uso de una prótesis robótica puede volverse tedioso y causar más problemas de los que pretende solucionar si esta posee un diseño y control inadecuado. Es por ello que tanto el diseño como el métodos de control de la misma han de ser cuidadosamente seleccionados y estudiados, para que así la prótesis robótica realice un aporte substancial de valor y pueda llegar a suplir de forma eficaz las desventajas ocasionadas por el uso de una prótesis robótica y, en un futuro no muy lejano, llegar a obtener una ventaja sobre las personas que disponen de una extremidad de forma natural.

Por todo esto, este trabajo se centrará en la investigación de los distintos métodos de control de posición de los dedos de una mano robótica y sus distintas implementaciones.

1.2. Objetivos

- 1) Estudiar los métodos existentes hoy día para el control de posición de una mano robótica.

En concreto se estudiará, tanto prácticamente mediante un prototipo como teóricamente, la forma de sensar y controlar la posición de los dedos en una mano robótica. Para ello se estudiarán las distintas soluciones a la hora de realizar un control de posición:

- Usando lazo abierto y cerrado.
- Aplicando técnicas de control digital basadas en microcontroladores.
- Utilizando un controladores proporcional P, integrativo I y Derivativo en lazo cerrado.

A su vez, en cada uno de estos controles realizaremos una caracterización de la planta de forma que podamos observar la variación de los polos y los ceros en función de los cambios que le realicemos al sistema de control.

- 2) Puesta en funcionamiento de Prototipo

Para ello, gracias a la técnica de impresión 3D, se imprimirán las piezas del prototipo seleccionado y será ensamblará.

Se diseñará el circuito de control utilizando dos controladores de motor diferentes. Uno de ellos será utilizado para esclarecer el funcionamiento del control del motor mientras que el otro controlador lo utilizaremos para que este circuito de control sea más eficiente y más sencillo de implementar.

Se creará el software necesario para que nuestro prototipo funcione de forma correcta y eficiente. Este software se implementará sobre un Arduino DUE, controlador escogido debido a la necesidad de potencia de computación, tanto de salidas y entradas necesarias en el sistema.

1.3. Estructura del documento

Este documento constará de seis capítulos. El primero de ellos será el de introducción, el cual se encuentra leyendo usted en este momento y donde se explica la motivación, los objetivos, el método empleado y la estructura del documento.

En segundo lugar y para contextualizar en el tiempo y el espacio, se encuentra el estado del arte, en el cual se hablará sobre la historia de las prótesis y la fina línea que separa actualmente la distinción entre mano robótica y prótesis, terminando con una comparación de tres prototipos actuales ordenadas en tres niveles distintos de dificultad: Fácil, intermedia y compleja

En el capítulo número tres se expondrá una descripción detallada del modelo que se pretende implementar. Esta descripción se dividirá en dos partes diferenciadas. En la primera de ellas explicaremos la mecánica del modelo, mientras que, en la segunda se explicará la arquitectura electrónica implementada para el control de posición de los dedos.

El capítulo número cuatro se centrará en la explicación de toda la teoría de control necesaria para la comprensión del funcionamiento de la mano y de sus sistemas de control de posición. Se realizará un análisis y explicación matemática de todos los tipos de control distintos explicados en el objetivo 1 del apartado 1.2.

En el capítulo cinco se encontrarán todos los experimentos y pruebas prácticas realizados hasta la fecha. Tratando así los problemas y las soluciones a la hora del montaje del prototipo, el control en lazo abierto y el control en lazo cerrado con el encoder, gracias al cual calcularemos la función de transferencia de la planta y por último todos los experimentos del control en lazo cerrado.

Finalmente en el capítulo seis se expondrán las conclusiones extraídas gracias a la realización de este Trabajo de Fin de Grado y se dará algunas líneas futuras que podrían seguirse para el desarrollo de la mano robótica.

2. Estado del Arte de las Manos Protésicas

Tras la introducción se proseguirá con un resumen de los aspectos generales más destacados en la historia de las manos protésicas, hasta la investigación actual. Hay que tener en cuenta que la línea que hoy en día separa las manos robóticas y las manos protésicas es muy fina y tiende a la desaparición. [1]

Este capítulo está organizado de manera cronológica desde los inicios de las prótesis antiguas, continuando con las manos accionadas por el cuerpo a finales del siglo XX, dando paso a las manos protésicas alimentadas de forma externa tanto forma neumática y mioeléctrico para concluir con los prototipos que destacan actualmente.

2.1. Prótesis Antiguas

Las primeras manos protésicas en la historia se trataban de prótesis pasivas, es decir, no articuladas. La prótesis más icónica y conocida es el gancho, gran estereotipo de los piratas. El gancho pasivo ofrece algunas funciones prácticas limitadas, no obstante no siempre es así, otras prótesis pasivas solamente ofrecen una función estética sin ninguna función práctica. Prótesis destinadas a ocultar lesiones sufridas en batalla o en el trabajo manual para así proporcionar al usuario una cierta sensación de bienestar. Por lo general estas prótesis pasivas estaban construidas con madera, hierro o cobre, con partes de cuero para su fijación a la extremidad residual. En algunos casos las prótesis ejercían una función particular, este es el caso de los caballeros de la Edad Media y Renacimiento que utilizaban manos protésicas a medida en las cuales podían portar escudos. Irónicamente los artesanos que realizaban tales prótesis se trataban de los mismos armeros que proporcionaban las armas y armaduras para la batalla.

Uno de los primeros registros de una mano protésica activa, es decir, una prótesis mecánica articulada es la mano de hierro del mercenario alemán Götz Von Berlichingen, el cual perdió su mano derecha con una bala de cañón durante el asedio de Landshut, Bayern, en 1504 cuando tenía aproximadamente 24 años. Tras haber sobrevivido a dicha

lesión, poseía una mano de hierro mecánica hecha para él por su armero Jagsthausen Württemberg, o puede que por un artesano en Nuremberg. Esta prótesis se adelantaba en el tiempo, se trataba de un sistema completo con cinco dedos articulados que podían moverse con la otra mano. Esta prótesis fue mejorando hasta que a través de un sistema de palancas y muelles se podía articular mediante la pulsación de un botón. Dicha mano protésica se puede observar hoy día en la exposición del castillo de Berlichingen. Se ilustra en la *Figura 2.1* [2]

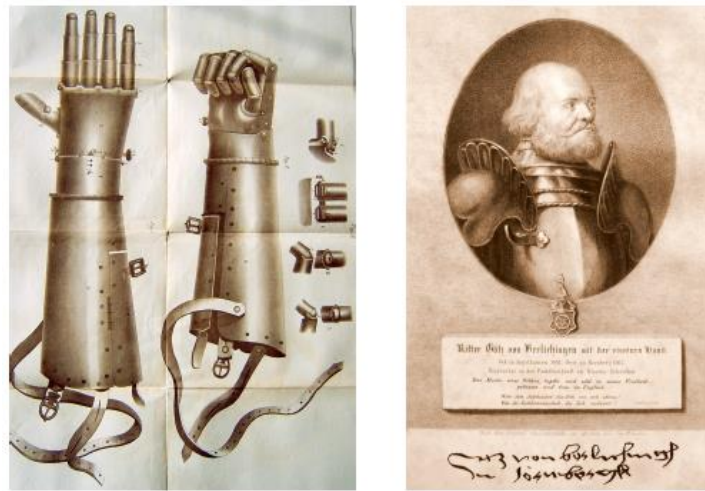


Figura 2.1 - Grabado de la mano de Götz von Berlichingen (Izq) y el retrato y firma de su dueño (dcha)

El cirujano militar Francés Ambroise Paré diseñó varias prótesis ingeniosas para suplir los miembros amputados. En la siguiente imagen se puede observar un esquema de una de las mencionadas prótesis en la *Figura 2.2*. [1]



Figura 2.2 - Esquema de Prótesis de Mano Activa realizada por Paré

2.2. Manos protésicas accionadas por el cuerpo

En 1812, Pierre Ballif, un dentista alemán y técnico quirúrgico en Berlín inventó una prótesis de mano mecánica que podría controlarse utilizando correas conectadas a un arnés de pecho. Tal y como se ilustra en la *Figura 2.3*. El aspecto de esta prótesis es similar a la de Götz Berlichingen, no obstante se trataba de una prótesis tres veces más liviana, estaba hecha de latón y la característica más destacable de la misma es que podía ser accionada sin la ayuda de la otra mano gracias a las correas. En su postura de reposo la mano se mantenía con el puño cerrado, al estirar el brazo, debido a la tensión de los cables, se realizaba una extensión de los dedos. Por el contra, este sistema tenía una escasa fuerza de agarre debido a que la misma solo provenía de los muelles. Aún y todo fue un avance revolucionario en las prótesis de mano. En realidad, este prototipo fue el primero del mundo que utilizaba este mecanismo y sigue siendo la base de la mayoría de las pinzas protésicas actuales.

La invención de Ballif se mejoró posteriormente gracias a la ingeniera de Berlín Margarethe Karoline Eichler, la cual invirtió el movimiento de la prótesis de la mano,

haciendo la extensión de los dedos cuando se estaba en una postura de reposo y la flexión de los mismos al realizar un movimiento de brazo. Esto conseguirá una fuerza de agarre mayor. A su vez cambió la composición del material de la mano, realizándola con latón y níquel, volviéndola aún más liviana, además se dice que se trataba de una prótesis elegante.

En 1844 el escultor holandés Van Peetersen realizó mejoras en los brazos artificiales para amputados por encima del codo, se le ocurrió el arnés de hombro, menos incómodo que el arnés de pecho. Ver *Figura 2.3 a) y b)*.

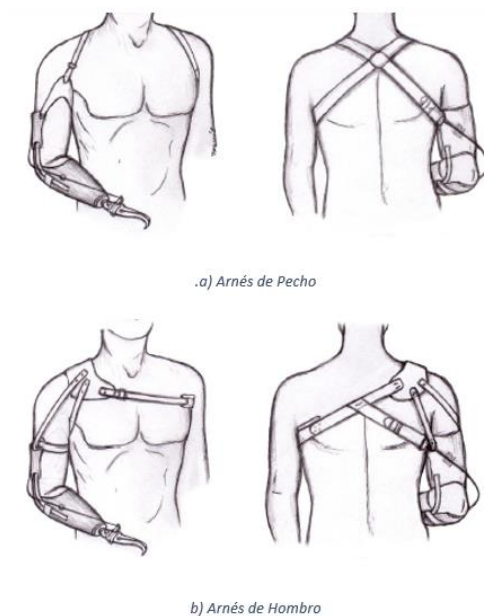


Figura 2.3 - Distintos tipos de Arneses

Durante la década de 1850 y 1860 el conde Amédee de Beaufort, que había sido nombrado “subinspector general de instituciones benéficas” inventó una serie de dispositivos articulados para amputados de las extremidades superiores e inferiores, especialmente para los menos desafortunados de la sociedad. Sobre las ideas de Van Peetersen, diseñó un producto simple, práctico, liviano y de bajo costo para amputados por debajo del codo y por encima del codo. Estos diseños podrían fabricarse en grandes cantidades y sería lo suficientemente robusto como para hacer posible que sus usuarios sean capaces de volver a trabajar.

2.2.1. Prótesis a finales del siglo XX

Durante las últimas décadas del siglo XIX y principios del XX, la tecnología de prótesis avanzó a un ritmo cada vez más rápido, particularmente en América. Fueron muchos los desarrollos que se presentaron en ese momento, no obstante pocos ofrecían ideas innovadoras. Un número significativo de ellos fueron copias o intentos de nuevos desarrollos. Numerosas patentes fueron registradas por todos esos ingeniosos inventores, al final de la Primera Guerra Mundial, en lo que parece un intento de asegurar una parte de lo que se vio como una oportunidad de negocio. Por ejemplo Allward (19989), Minzey (1893), Dorrance (1912), Bosch (1913,1919), Jeffery (1918) y Baehr (1919) que patentó varias pinzas accionadas por un cable. Las manos patentadas por cable de Caron (1917, 1918a, b, c, 1920) y E. Robinson (1919) actuaban a través de un arnés para hombros. S. Lucas (1890), Patton (1903) y Z. Taylor (1905) diseños patentados simples de pinzas o manos "tipo Götz", es decir, tuvo que ser operado por la otra mano. Nelson (1909), Shackelford y Alexander (1911), Carnes (1904, 1911, 1912, 1913), Cronemiller (1917) y Rohrmann (1919) construcciones complejas de mano patentadas, con numerosas partes, engranajes, cables, resortes, etc.

Es curioso que todas estas manos artificiales hayan llegado más allá de la etapa de prototipo, y también podemos preguntarnos sobre su usabilidad real. No obstante uno de ellos fue un hito en la historia de las prótesis. El gancho de Dorrance, patentado en 1912 por su inventor David Dorrance, él mismo amputado de la mano derecha y fundador de una compañía de prótesis, la ahora llamada, Hosmer Dorrance Corporation.

El gancho de Dorrance, también conocido como el gancho de Hosmer, es una simple, robusta, económica pinza de dos garras. Es duradera y fácil de operar. Se mantiene cerrada por una banda de goma. Se logra abrir mediante un movimiento respecto al hombro. En la *Figura 2.4* se puede observar en detalle. Se realizaron múltiples variaciones de él (por ejemplo, Trautman 1919, Kosek y Trautman 1928; Dorrance 1930, 1935, 1936; David 1940, 1942), y ha seguido siendo una prótesis de elección para los amputados desde entonces. Hoy día es el dispositivo terminal más común para amputados de miembros superiores y continúa siendo fabricado por Hosmer Dorrance Corporation, y también forma la línea base de otras dos grandes compañías de prótesis,

la alemana Otto Bock Healthcare y la británica RSL Steeper (Hosmer Dorrance Corporation 2010; Otto Bock 2010; RSL Steeper 2010).

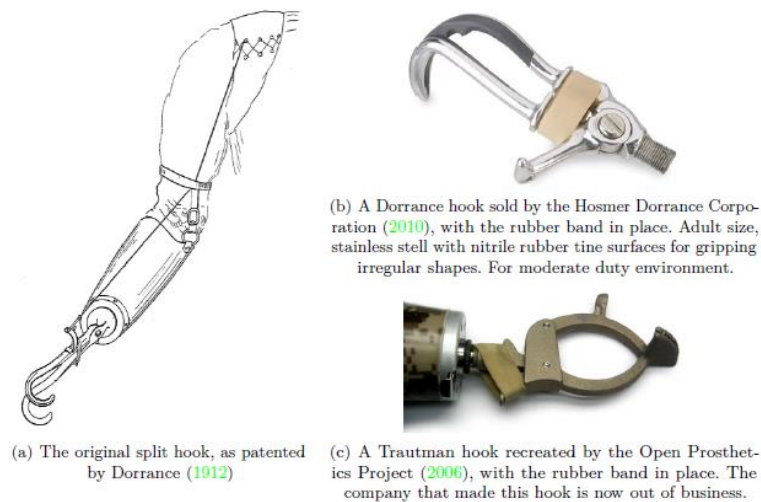


Figura 2.4 - Prótesis de Dorrance a) y b) /Prótesis de Trautman c)

Otra famosa prótesis de la época es el brazo antropomórfico de Carnes que se puede observar en la *Figura 2.5*. William Carnes, un estadounidense, había perdido su brazo derecho por encima del codo en 1902 en un accidente en una tienda de máquinas. Insatisfecho con las prótesis que probó pasó a hacer un modelo propio (Carnes 1904). La exitosa venta a otros amputados similares a él lo llevó a la creación de la Carnes Artificial Limb Company 1901. Tras varias modificaciones de la misma se fue mejorando y ampliando el mercado. Sus partes externas estaban realizadas con fibra vulcanizada y encerraba un complejo mecanismo de más de 200 partes de acero accionadas por tres cuerdas controladas por el movimiento de los hombros y del tocón amputado. La prótesis permitió a su usuario final realizar movimientos más complejos y detallados.

Entre la Primera Guerra Mundial y la Segunda Guerra Mundial, el desarrollo de manos protésicas se ralentizó. Sobre el año 1920 se incorporan nuevos materiales, incluyendo acero inoxidable, aluminio y aleaciones de aluminio ligero, fueron cada vez más utilizados. Una mano accionada por cable con muelles en espiral para los dedos fue desarrollado por Laherty y D. Becker (1933), y luego mejorado y comercializado durante varias décadas por D. Becker. Los diseños resultantes, hoy son conocidos como "Becker

Lockgrip Hand" y "Becker Imperial Hand", que todavía son fabricados por el negocio familiar del inventor en Saint Paul, Minnesota.

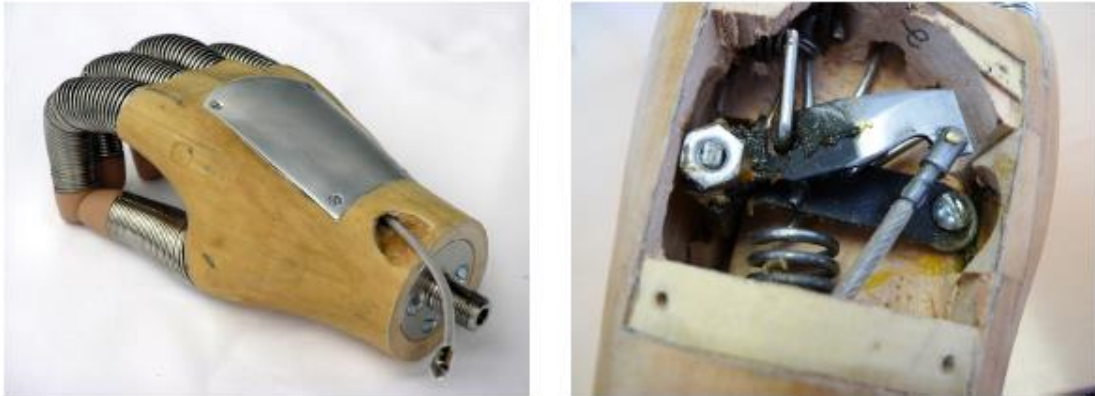


Figura 2.5 - Prótesis de Mano Moderna

2.2.2. Prótesis Modernas: Las secuelas de la Segunda Guerra Mundial

Al final de la Segunda Guerra Mundial, en respuesta a la movilización de veteranos de guerra y en la solicitud del cirujano general del ejército, el gobierno de los Estados Unidos aumentó los fondos para la investigación y desarrollo de prótesis. Lanzando una investigación ambiciosa. Se trataba de un programa a través de la Academia Nacional de Ciencias. El programa dio como resultado una nueva rama de conocimiento, la biomecánica. Cambiando así en la práctica la idea que existía de prótesis elevándola a un nivel nuevo. Durante la década de 1950 con estas nuevas técnicas y la sustitución del material de fabricación por plásticos y aleaciones de titanio.

Y así, después de la Segunda Guerra Mundial y la década de 1950 se marcó el nacimiento de la prótesis moderna así como el comienzo de EEUU en la participación de los estudios de la prótesis, esta participación continua hoy. [1]

2.3. Manos Protésicas Alimentadas de Forma Externa

A mediados del siglo XX comenzó la investigación centrada en las prótesis alimentadas de forma externa y el uso de la electromiografía para controlar dichas prótesis. A diferencia de los ganchos y manos mecánicas accionadas por cable que obtienen su poder y control desde el cuerpo de la persona amputada, las prótesis alimentadas externamente, como su propio nombre indica, obtienen la potencia de otra fuente, utilizando desde baterías eléctricas hasta bombonas de gas, usualmente de dióxido de carbono líquido para aquellos que optaban por el uso de actuadores mecánicos. En los siguientes párrafos se explica cómo trabajan este tipo de prótesis.

2.3.1. Manos Protésicas Neumáticas

Las primeras prótesis de gas se desarrollaron justo después de la Segunda Guerra Mundial en las universidades clínicas en Heidelberg, Alemania y Vaduz, respectivamente, por Otto Häfner y Sigmund Weil y por Edmund Wilms. La construcción de Edmund Wilms fue una mano de cierre voluntario activada por un músculo. Se colocó una almohadilla llena de aire entre el muñón y el alvéolo de la prótesis, conectado gracias a algún tipo de interruptor por un tubo flexible, de modo que cuando el amputado contrajera los músculos de su muñón, el cambio de presión en la almohadilla accionase el interruptor que llevaba a la entrada de control de la prótesis. El desarrollo de estas prótesis de gas continuó durante la década del 1950 en Alemania, no obstante hasta finales de la década de 1960 no se aceleró realmente, extendiéndose a otros países. Se desarrollaron varios brazos y manos protésicas usando interruptores o válvulas con sus manos y dedos vestigiales o bien con movimientos de los hombros.

2.3.2. Manos Protésicas Mioeléctricas

La electromiografía se define como la prueba diagnóstica practicada con la finalidad de evaluar la actividad muscular tras un estímulo eléctrico, el estado de los nervios encargados de activar el músculo evaluado y el funcionamiento de la unión neuromuscular. Para ello se pueden utilizar electrodos de superficie colocados sobre la piel del músculo (electromiografía de superficie) o mediante electrodos de aguja (Electromiografía intramuscular). Desde el siglo XIX se sabe que la contracción de las fibras musculares se asocia con la actividad eléctrica, no obstante la capacidad de detectar con precisión estas señales ha dependido de los avances en electrónica realizados durante la primera mitad del siglo XX.

Es importante destacar que las señales no provienen de las células nerviosas sino de las musculares, aunque ambas están ampliamente relacionadas. Las fibras musculares están organizadas en pequeños grupos llamados “unidades motoras únicas”, correspondiente al grupo contráctil más pequeño. De forma que un solo potencial activo en una sola unidad motora estará asociado a una contracción nerviosa, concentraciones sostenidas requerirán disparos repetidos de la unidad de control individual, lo que resulta en una cadena de potenciales de acción y a mayor fuerza necesitada para realizar dicha contracción, mayor será el número de unidades motoras individuales que emitan ese potencial.

La electromiografía intramuscular registra la actividad eléctrica de unas pocas unidades motoras individuales, mientras que la electromiografía superficial da una imagen más general. Ambas poseen muchas aplicaciones, por ejemplo, en hospitales a la hora de diagnosticar ciertas condiciones neuromusculares, o en laboratorios de investigación involucrados en biomecánica o fisiología neuromuscular, o en prótesis.

El control electrónico integrado de los sistemas de manos protésicas mioeléctricas puede ser relativamente sofisticado, compensando el hecho de que el pequeño número de electrodos limita la destreza de la mano. En primer lugar, la mayoría, si no todos, los sistemas de control mioeléctrico contemporáneos permiten algún tipo de control proporcional desde el amputado a la prótesis, es decir que el grado de apertura o cierre

de la mano, o la velocidad de las articulaciones, o la fuerza de la prensión, son proporcionales a la intensidad de la contracción muscular. Por lo tanto, hay retroalimentación de algunos sensores a la prótesis, para el control de bucle cerrado de la fuerza prensil, la posición de la articulación y la velocidad de la articulación.

En las manos protésicas más nuevas, los microprocesadores incorporados proporcionan un control avanzado de lazo cerrado, en un intento de hacer que la mano sea inteligente y semiautónoma. La idea detrás de este objetivo es proporcionar al usuario una mano artificial que se preocupa por ciertos aspectos por sí misma, de la misma manera que la mano biológica parece cuidar esos aspectos en sí misma, debido a que su control por el sistema nervioso es casi automático y subconsciente. Esta tendencia hacia las prótesis semiautónomas se encuentran claramente más cercanas a la robótica con una serie de problemas compartidos respecto a sensores y control: dos áreas de investigación que son reconocidas como retos y críticas para un futuro potencial. [1]

2.4. Prototipos Actuales Destacados

Pese a existir muchos proyectos distintos de fabricación casera, de código libre y del ámbito empresarial, todos se basan en los mismos mecanismos básicos de funcionamiento con diferentes niveles de complejidad, durabilidad, costes, etc.

Por todo ello distinguiremos tres proyectos distintos, primero hablaremos de un proyecto recopilación de todos los tipos de manos robóticas de fácil construcción que se puede encontrar en la web. Tras esto el siguiente apartado tratará de un proyecto del ámbito privado y para terminar expondremos el prototipo que proponemos, basado en el ámbito de la investigación y el aprendizaje.

Tanto el primero como el segundo incorporan técnicas de impresión en 3D. Debido a su bajo coste y a su efectividad y fácil reemplazamiento.

2.4.1. Recopilación de Proyectos de Bajo Nivel de Complejidad

Este apartado es una recopilación de proyectos con el mismo método de control de posición de los dedos.

Solamente se ha encontrado información en sencillos videos, no obstante es suficiente para poder observar como el método de control de posición de los dedos es distinto.

El método de control constará de 5 servomotores, en algunas ocasiones colocados en el interior de la palma de la mano y otras veces en el exterior. Esto se debe a que los servomotores ocupan un mayor volumen que los motores de DC y su integración física es más compleja.

En el caso en que los servomotores se encuentren en el exterior de la palma de la mano según se puede observar en la *Figura 2.6 Izquierda*, los hilos se han de reconducir hasta el lugar donde están situados los servomotores, los cuales generarán el movimiento, cosa que es más sencilla cuando los servomotores se encuentran integrados en la palma de la mano, *Figura 2.6 Derecha*. [3] [4] [5]



Figura 2.6 - Posición de motores en distintos prototipos

2.4.2. Proyecto Ámbito Empresarial: Shadow Dextrous Hand C6M

Shadow Dextrous Hand es un sistema de mano robot humanoide avanzado que proporciona 24 movimientos para reproducir lo más cercanamente posible los grados de libertad de la mano humana. Ha sido diseñado para proporcionar salida de fuerza y sensibilidad de movimiento comparables a la mano humana.

El modelo C6M Hand utiliza el sistema de accionamiento eléctrico "Smart Motor" de Shadow, en lugar del sistema neumático de activación de Air Muscle de otros sistemas. El "Motor inteligente" integra la electrónica de control de posición y de la fuerza, la electrónica de accionamiento del motor, el motor, la caja de cambios, la detección de fuerza y las comunicaciones en una unidad compacta.

Se trata de un sistema autónomo: toda la actuación y la detección requeridas están integradas en la Mano, a su vez incorpora todos los sistemas de control necesarios y documentación para propósitos de investigación y enseñanza. Se puede observar en la *Figura 2.7* [6]



Figura 2.7- Shadow Dexterous Hand E1 Series [5]

2.4.3. Proyecto Investigación: Dextra

Dextra es el proyecto desarrollado por Álvaro Villoslada que se ha basado este Trabajo de Fin de Grado. Se trata de una mano protésica cuyo objetivo es el de hacer una prótesis robótica de bajo precio y de código abierto, lo cual lo logró gracias a su diseño modular y a la impresión 3D.



Figura 2.8 - (Izq) Proyecto Dextra (Dcha) Detalle del Interior del Prototipo

El elemento principal de Dextra es el módulo de dedo. Como ya he mencionado anteriormente la mano es modular: los cuatro dedos son intercambiables, y el pulgar es una variación del módulo de dedo.

El módulo de dedo comprende, como se puede observar en la *Figura 2.8.(Dcha)*, el dedo mecánico imprimible, el actuador y un codificador. El actuador utiliza un motor de DC para girar un carrete y así enrollar un hilo que atraviesa el interior de todo el módulo del dedo, convirtiendo el movimiento de rotación del motor de DC en un movimiento lineal que logrará hacer que el dedo se flexione y se extienda. [7]

Tras esta breve introducción de la evolución de las manos protésicas, se continuará con la explicación del prototipo desarrollado.

3. Descripción del Sistema Propuesto

Una vez revisados los distintos prototipos que se han ido desarrollando a lo largo de la historia, llegando así al prototipo en el cual se basa este trabajo, se explicará en el capítulo actual cómo se ha desarrollado nuestro prototipo, cuál es su mecánica y como es su arquitectura de control.

3.1. Descripción General

Para comenzar, se realizará una descripción general del sistema propuesto. Donde se verán todos y cada uno de los bloques que harán que nuestro prototipo funcione. Tal y como se puede observar en la *Figura 3.1* el sistema desarrollado se puede dividir en los siguientes componentes:

En primer lugar se necesitará un ordenador, este será el encargado de mantener el contacto con el controlador. Se encargará en un principio de introducir en el controlador la programación deseada y tras esto será el canal a través del cual nuestro microcontrolador recibirá las órdenes para realizar los distintos movimientos de nuestra mano. A parte será también el medio de salida de datos del Arduino DUE para realizar la recogida de información para los resultados experimentales.

El controlador anteriormente mencionado se encargará de gestionar las operaciones y de comparar la salida de la mano con la referencia deseada y actuar en consecuencia a través de la etapa de potencia para la mano. En él se implementará de forma digital un controlador PID (véase el Capítulo 4) para cada dedo así como el control del servo. De forma que la realimentación en el sistema será la información que le envíen los sensores situados en cada dedo, estos indicarán el estado actual de cada control, se pueden observar recuadrados en color morado en el Sistema Mecánico de la *Figura 3.1*.

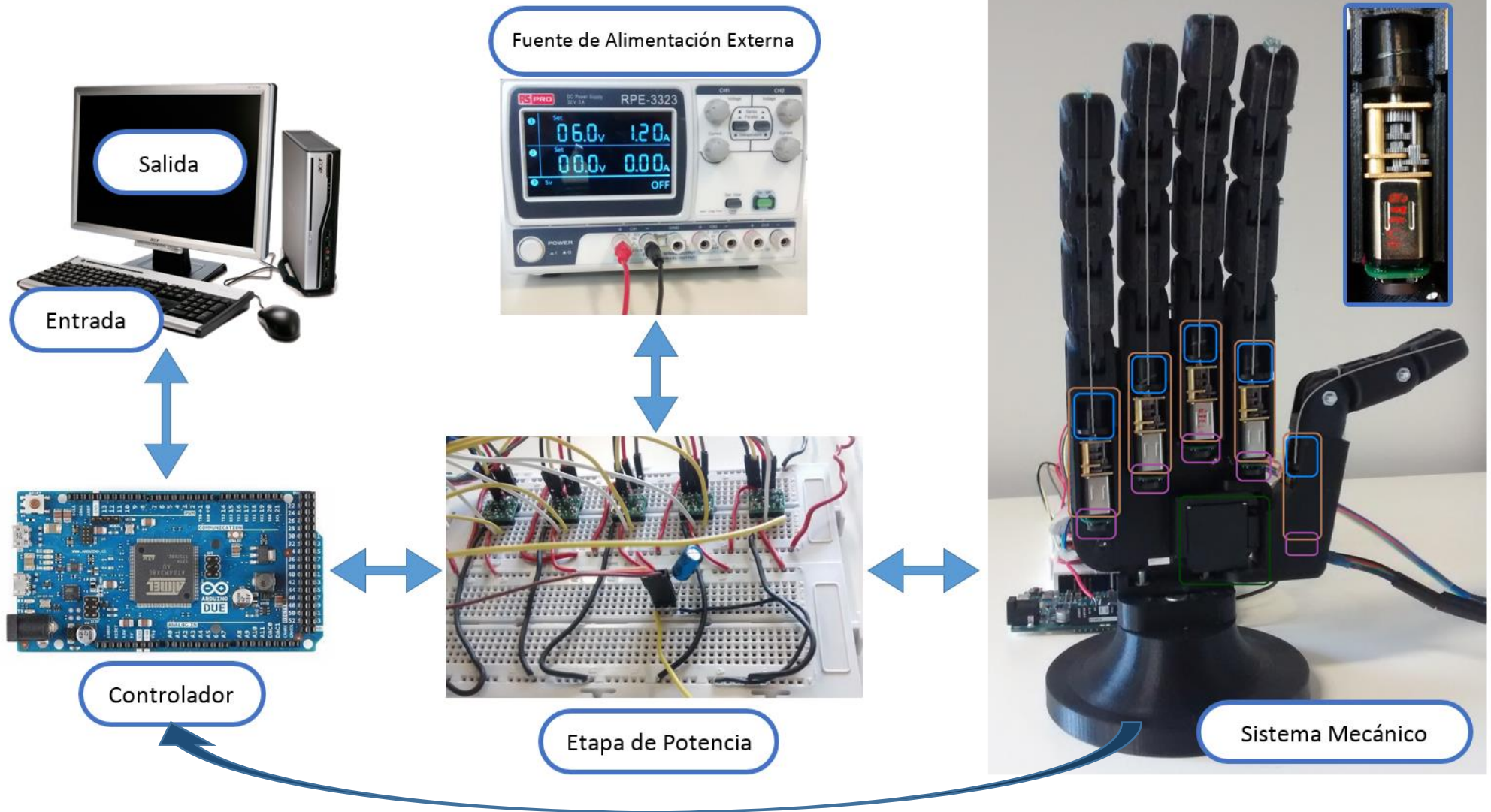


Figura 3.1 - Esquema General de Nuestro Sistema

Al ser la alimentación que proporciona el microcontrolador insuficiente para alimentar los motores y dar el torque necesario se necesita una etapa de potencia que los alimente de forma adecuada y a su vez traduzca las órdenes proporcionadas por el microcontrolador a las señales necesarias para el giro de los motores.

Para la ya mencionada etapa de potencia se necesita una alimentación que será proporcionada por una fuente de alimentación externa. Actualmente el sistema se encuentra alimentado con una fuente de alimentación que necesita estar conectada a la corriente, no obstante en un futuro se pretende sustituirla por una batería.

Todos estos elementos anteriores serán los necesarios para el correcto funcionamiento de nuestro Sistema Mecánico. El cual incorporará en su interior cinco motores de DC así como un servomotor encargado de manejar el pulgar oponible. Bajo cada motor de DC se colocará el sensor magnético anteriormente mencionado para proporcionar a nuestro sistema de una realimentación con la cual controlar la posición de cada dedo.

En las secciones siguientes se detallan con más detalle cada una de estas partes.

3.2. Mecánica

La parte mecánica de nuestro prototipo es exactamente idéntica a la del prototipo de Álvaro Villoslada, debido a que el interés de este proyecto se centraba sobre la parte de control del sistema, dejando en un plano secundario a la parte mecánica.

Como ya explicamos en el capítulo anterior, la parte mecánica se centra en el módulo del dedo. Este mismo basa su estructura en la estructura ósea del cuerpo humano y por tanto se divide en cuatro secciones, tal y como se puede observar en la *Figura 3.2*.

Observando las piezas de nuestro modelo, parte derecha de la *Figura 3.2*, de abajo a arriba se puede comprobar como el esquema es idéntico al de la estructura ósea real de una mano. Primero nos encontraremos con los metacarpianos, los cuales en nuestro prototipo los utilizaremos para la colocación del motor de DC que generará el movimiento. Le seguirá las falanges proximales, las falanges medias y las falanges distales.

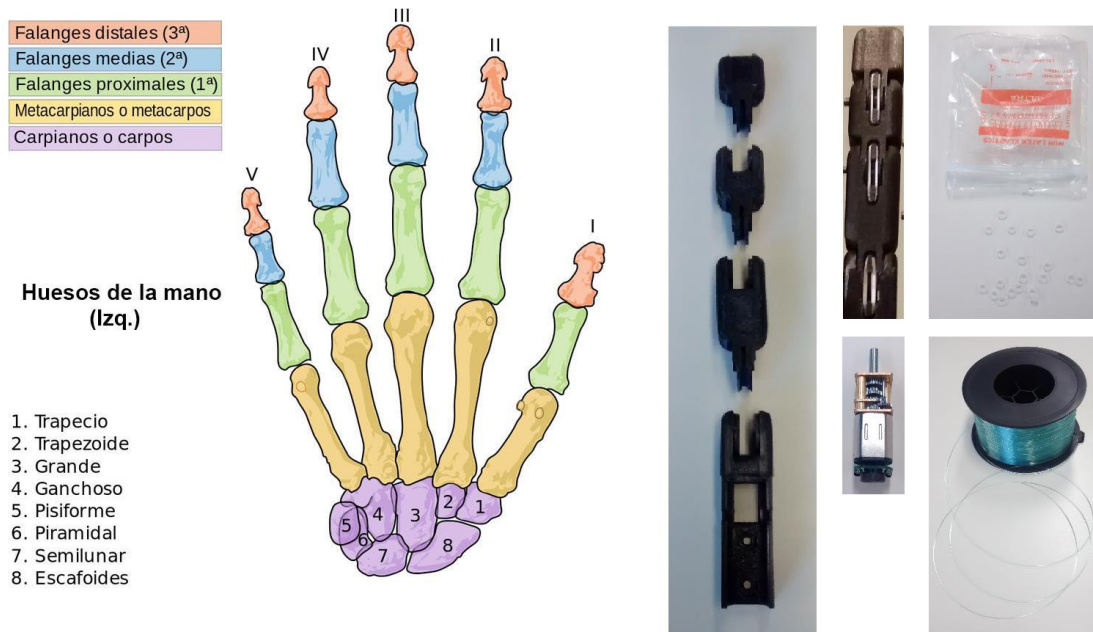


Figura 3.2 - Comparación Estructura Ósea con Nuestro Prototipo [8]

El funcionamiento de la mano será el siguiente: El motor en DC será colocado en el interior del metacarpiano. Las gomas serán colocadas en las diferentes uniones entre las falanges a modo de tensores (*Figura 3.2, sobre el motor*), por lo que existirá siempre una fuerza con tendencia a tener el dedo siempre extendido. A modo de tendón, todo el módulo del dedo es atravesado por un cable, puede observarse en la *Figura 3.1* de color gris. Este cable será del que se valga el motor para ejercer la fuerza que se encargará de hacer el movimiento de flexión de los dedos. Para ello, el motor constará de una caja reductora de engranajes que aumentará la fuerza del mismo y permitirá un control exacto de la posición del motor y por tanto del dedo, todo esto se puede ver con claridad en el Sistema Mecánico de la *Figura 3.1*. Al rotor saliente de la caja reductora se le incorporará una bobina que facilite el recogido del cable de una forma correcta (ver *Figura 3.1* Sistema Mecánico, recuadro azul). El movimiento de extensión de los dedos se producirá al girar el motor en sentido contrario y liberar el cable, pues gracias a los tensores anteriormente mencionados, se realizará una fuerza opuesta a la del motor, tal y como se puede observar en la *Figura 3.3*.



Figura 3.3 - Vista general del Prototipo

El funcionamiento del dedo gordo será gracias a un servomotor que se encuentra situado en la palma de la mano. Este servomotor será el encargado de realizar el movimiento para colocar el pulgar de forma oponible para que así el prototipo sea capaz de realizar movimientos de agarre. Todo esto se puede observar en la *Figura 3.4*, para una mayor comprensión del mecanismo se han desmontado los dedos y la palma de la mano de forma que se pueda observar claramente el interior de nuestra prótesis. A su vez también en la parte de debajo de la misma figura se observa una comparativa entre el módulo del dedo y el módulo del dedo pulgar. Como se puede observar también está basado en la estructura ósea, teniendo aparte del metacarpiano, solamente dos falanges en vez de tres, la falange distal y la proximal.



Figura 3.4 – Arriba, Mecanismo de Movimiento para un Pulgar Oponible. Abajo, Comparación entre el Módulo de Dedo Normal y el Módulo del Pulgar.

3.3. Arquitectura Electrónica

Una vez conocida la mecánica de nuestro prototipo se explicará a continuación la electrónica que hará que esta prótesis cobre vida y pueda realizar los movimientos que deseemos.

El mecanismo de movimiento constará de los siguientes componentes electrónicos ya presentados en el primer apartado de este capítulo:

- Actuador: Motor DC
- Sensor: Encoder
- Etapa de Potencia: Drivers
- Controlador: Arduino

Pasaremos ahora a explicar cada uno de estos apartados. Todos estos elementos están basados en la parte mecánica por el diseñador del modelo, Alvaro Villoslada, sin

embargo el tipo de controlador basado en Arduino y la codificación del PID lo hacen diferente y es donde haremos una aportación.

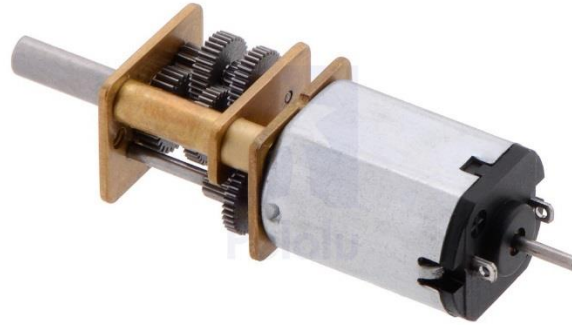
3.3.1. Actuadores

Motor DC

El actuador que se encargará de generar el movimiento en nuestra prótesis será un motor en DC, no obstante dicho motor tendrá que ser de pequeño tamaño para encaje con el tamaño de una prótesis normal pero a la vez tendrá que tener el suficiente torque como para poder vencer a la fuerza que los tensores del dedo realizan de forma opuesta a su movimiento y un control preciso. Es por todo esto por lo que se escogió el modelo:

Pololu 1000:1 Micro Metal Gearmotor HP 6V with Extended Motor Shaft

Este modelo que se puede observar en la *Figura 3.5*, aparte de ser de un tamaño adecuado, se trata de un motor de gran robustez, gran torque y de alta potencia que contiene todas estas características. Gracias a su caja reductora convertirá 1000 vueltas del rotor principal en una sola vuelta, gracias a lo cual se obtendrá una gran resolución a la hora de realizar el control. Por otro lado se trata de un motor de alta potencia (HP, High Potency) para que así el torque pueda ser el suficiente para recoger el cable en la bobina venciendo a los tensores que hacen la fuerza opuesta. Por último posee un eje trasero, gracias al cual podremos colocar un sensor magnético y realizar el conteo de vueltas mediante un encoder.



www.pololu.com

Figura 3.5 - Motor Pololu escogido para el prototipo [9]

Servomotor

Se utilizará el servomotor:

Turnigy TGY-EX5252MG Twin BB Digital Micro Servo

Este motor posee el torque adecuado para manejar nuestro pulgar. A su vez las dimensiones están calculadas en torno a él. No tiene características especiales que explicar. Se puede observar en la *Figura 3.6*.



Figura 3.6 - Servomotor utilizado en prototipo

3.3.2. Sensor

El sensor será el encargado de llevar la cuenta de las vueltas que realice nuestro motor para así poder realizar un control de la posición. Este encoder está condicionado completamente a las dimensiones del motor y es por ello que se buscó un dentro del mismo fabricante. Se escogió el codificador: (Vease *Figura 3.7*)

Magnetic Encoder Pair Kit for Micro Metal Gearmotors, 12 CPR, 2.7-18V

Como se puede observar en el nombre del componente el sensor escogido fue un encoder específicamente diseñado para nuestro tamaño de motor. Este codificador posee un voltaje máximo de 18V y un voltaje mínimo de 2.7V, lo cual es adecuado para nuestro motor dado que el voltaje máximo que soporta es de 6V. A su vez cuenta con un CPR (Count Per Revolution) de 12, lo cual mejorará notablemente nuestra resolución de movimiento debido a que si ya poseíamos una caja reductora que cada 1000 vueltas de rotor interno el rotor externo daba una vuelta, ahora nos encontramos que cada vuelta que gire el rotor estará codificada por 12 pulsos, lo cual genera una resolución de 12 multiplicado por 1000, un total de 12.000 posiciones por vuelta. Teniendo en cuenta que cada vuelta posee 360°, si dividimos 360 entre 12.000 nos dará una precisión de 0,03 grados por posición o de forma inversa 33,3 posiciones por cada grado.

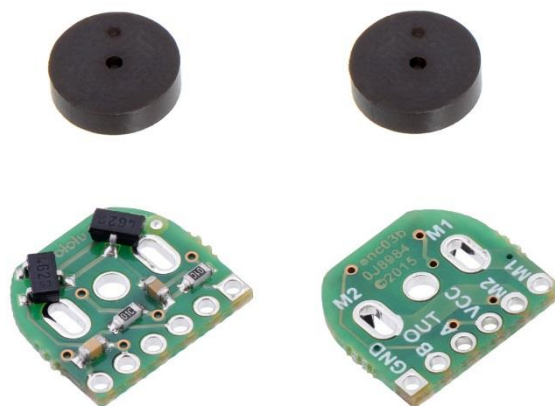


Figura 3.7 - Codificador Pololu escogido para el prototipo [10]

3.3.3. Etapa de Potencia

En el transcurso de este proyecto se han trabajado con dos etapas de potencia distintas. Esto se debe a las ventajas que ofrecen cada uno de ellos. En primer lugar el driver L293D nos proporcionará una claridad a la hora de comprender el funcionamiento del circuito mientras que una vez ya comprendido el DRV8838 nos proporcionará una simplificación de circuito que utilizaremos para realizar así una implementación física más sencilla y limpia.

DRV8838 Single Brushed DC Motor Driver Carrier

Este driver fue el preseleccionado en el proyecto. Esto se debe a que simplifica en gran medida el circuito necesario para llevar a cabo el control de un motor no obstante aporta un mayor complejidad a la hora de la comprensión del circuito. Se puede observar en la *Figura 3.8 Izq.*

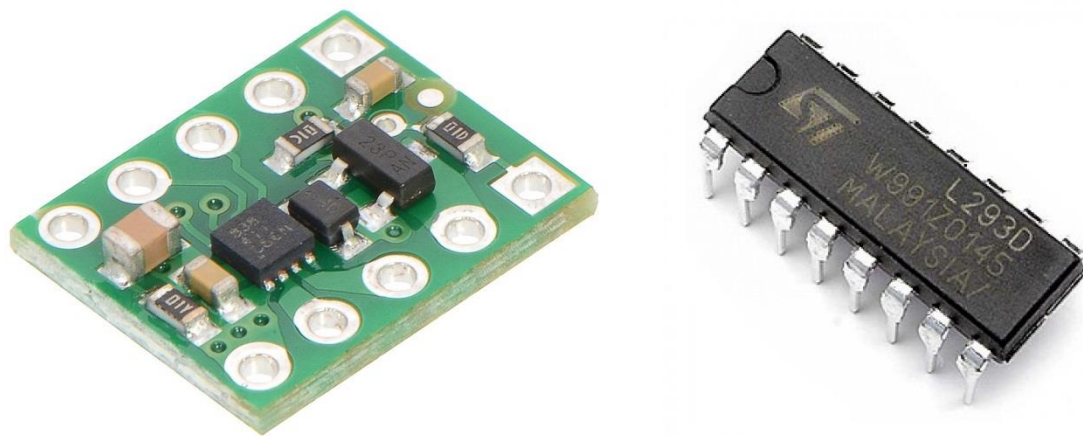


Figura 3.8 – (Izq) Driver preseleccionado para el circuito de control [11] (Dcha) Controlador de Circuito L293D [12]

3.3.3.1. L293D

Este controlador fue el finalmente escogido para realizar el control del motor. Aporta una complejidad al circuito, no obstante da una visión clara y sencilla de lo que ocurre en el circuito de control. Se puede observar en la *Figura 3.8 Dcha.*

3.3.4. Controlador

En este proyecto se han utilizado dos controladores distintos. Ambos se tratan de placas Arduino, concretamente el encargado de realizar las pruebas sencillas de control era el Arduino UNO mientras que el encargado de mover y manejar toda la prótesis será un Arduino DUE. En la *Figura 3.9* se especifican las principales diferencias entre ambos.

Name	Processor	Operating/Input Voltage	CPU Speed	Analog In/Out	Digital IO/PWM	EEPROM [kB]	SRAM [kB]	Flash [kB]	USB	UART
Uno	ATmega328P	5 V / 7-12 V	16 MHz	6/0	14/6	1	2	32	Regular	1
Due	ATSAM3X8E	3.3 V / 7-12 V	84 MHz	12/2	54/12	-	96	512	2 Micro	4

Figura 3.9 - Tabla Comparativa entre Arduino UNO y DUE [13]

El principal motivo por el que no utilizamos Arduino UNO para controlar por completo la prótesis es debido a que su frecuencia de operación no es suficiente como para llevar de forma adecuada el control de los dedos y tampoco consta de salidas suficientes como para controlar 5 motores DC y un servomotor.

En la *Figura 3.10* se puede observar las diferencias físicas entre un modelo y otro.

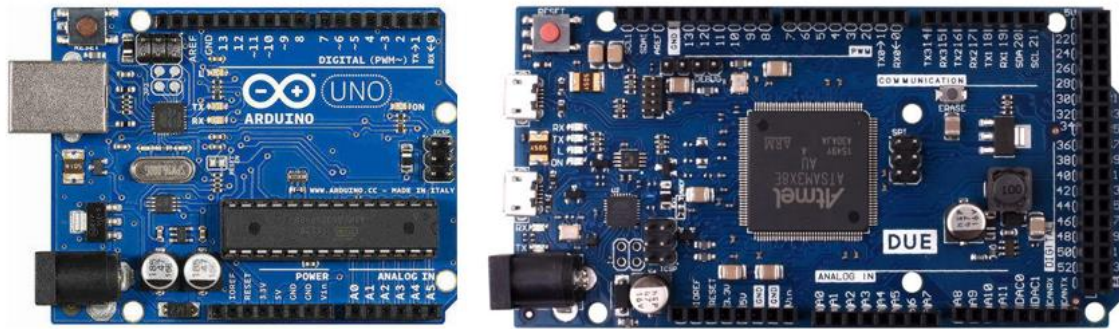


Figura 3.10 - (Izq) Arduino UNO y (Dcha) Arduino DUE [14] [15]

Ambos controladores utilizan el lenguaje de Arduino, que se trata de un lenguaje basado en C++. Para introducir la programación en el microcontrolador es necesario utilizar el IDE (Entorno de Desarrollo Integrado) de Arduino. Esto se realiza mediante una conexión USB al ordenador en el que se encuentra instalado el IDE.

El código que le introduciremos para controlar nuestro prototipo constará de un bucle de control que será el encargado de decidir que tarea realizar en función a la salida que se obtiene de nuestro sistema y el punto deseado. A su vez nuestro controlador estará constantemente comprobando si se ha escrito alguna orden en el monitor serial. Por último también estará atento a las interrupciones y a incrementar o decrementar el valor de posiciones en función del sentido de giro del motor. Todo esto se detallará con profundidad en los siguientes capítulos.

Todo el código que se haya generado a lo largo del transcurso de este Trabajo de Fin de Grado se guardará a modo de Anexo en una carpeta que acompañará a este documento.

4. Electrónica de Control: Control de Posición de un Motor DC

Los sistemas de control son ampliamente utilizados en todo el mundo, se pueden observar dentro del ámbito de las telecomunicaciones así como en los procesos industriales, la aviación, etc. Dentro del amplio campo que abarca la electrónica de control este capítulo se centrará en indagar sobre el control de posición de un motor DC. Estos actuadores eléctricos son ampliamente utilizados en la robótica hoy en día y constituyen una parte fundamental dentro de este Trabajo de fin de Grado.

Dentro del campo de control existen muchas clasificaciones para estos sistemas, bien sea por su naturaleza analógica o digital o bien mixta, por su número de entradas y salidas, etc. No obstante la clasificación principal separa a los sistemas de control en dos grandes grupos: Sistemas de lazo abierto y Sistemas de lazo cerrado.

Cada uno de estos es interesante en función de la aplicación que se quiera obtener. Por ejemplo, si un sistema no da la respuesta deseada (lazo abierto), puede modificarse su comportamiento dejando intacto el sistema y agregando un lazo de control usando generalmente retroalimentación negativa.

En este capítulo se explicarán los conceptos de lazo abierto y lazo cerrado para luego centrarnos en los distintos sistemas de control de lazo cerrado debido a que el control que se ha realizado se trata de un control PID. Este control se realizará de forma discreta gracias a nuestro controlador, por lo que a la hora de observar su respuesta en frecuencia tendremos que utilizar la Transformada Z, herramienta que será a su vez brevemente explicada.

4.1 Tipos de Control: Lazo Abierto y Lazo Cerrado

Se denominan sistemas de control de lazo abierto (“open loop”) a aquellos sistemas en los que la salida no tiene efecto sobre la acción de control. Es decir, la salida ni se mide ni se realimenta para compararla con la entrada y así poder hacer un seguimiento de la misma tal y como se puede observar en la *Figura 4.1*.

En un sistema de control de lazo abierto, no se compara la salida con la entrada de referencia, es decir, no se realimenta la entrada. Por tanto, para cada entrada corresponderá una condición de operación fija. Gracias a esto la precisión de un sistema en lazo abierto dependerá por entero de una calibración previa. [16]

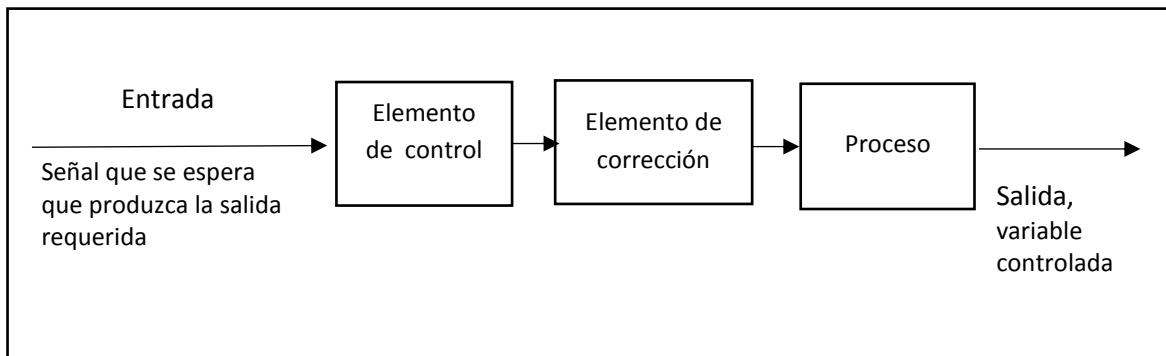


Figura 4.1 - Control en Lazo Abierto

Los sistemas de control de lazo cerrado se caracterizan por tener sensores que vigilan el proceso automatizado. La realimentación nos marcará el estado actual, de forma que gracias a nuestro elemento de control podamos calcular la diferencia con el estado deseado para que así el controlador actúe en consecuencia tal y como se puede ver en la Figura 4.2. Nuestro sistema de control está construido en base al mismo concepto.

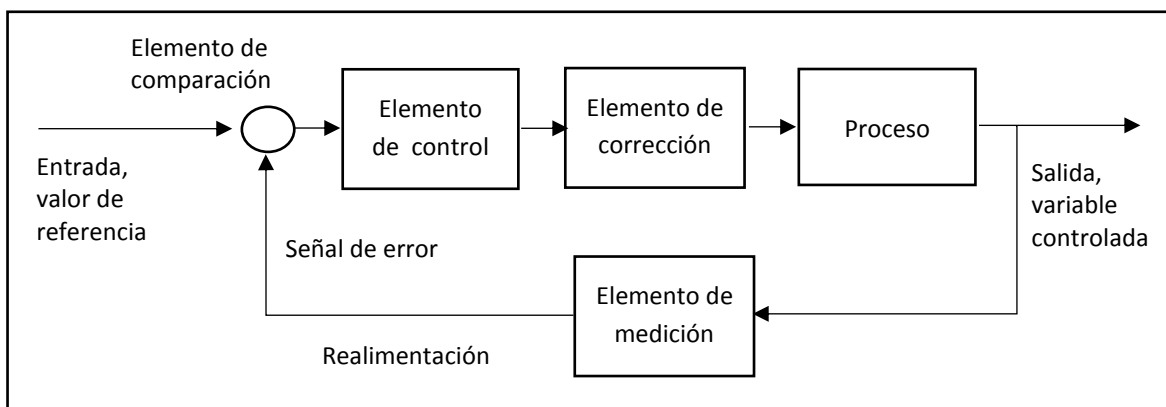


Figura 4.2 - Control en Lazo Cerrado

Lo que se puede observar en la Figura 4.1 y en la Figura 4.2 son las dos estructuras de control principales ya mencionadas representadas mediante diagramas de bloques. Cada caja representa una parte esencial del sistema el cual siempre recibirá una señal de entrada que será procesada y nos devolverá una señal de salida.

En el caso de sistema en lazo abierto solamente existirá una entrada al elemento de control, del cual realizará una función que creará una señal de salida que será la que el actuador recoja para realizar una función. Este actuador desarrollará un proceso que tendrá un efecto, el cual se determina como: Señal de salida del sistema. [17] [18]

4.2 Tipos de Control en Lazo Cerrado

Dentro del control en lazo cerrado existen múltiples metodologías con el propósito de llevarlo a cabo. Las que cuentan con una mayor presencia en el ámbito tecnológico actual son las llamadas:

- Control SI/NO (ON/OFF)
- Control Proporcional
- Control Proporcional Integrativo (PI)
- Control Proporcional Derivativo (PD)
- Control Proporcional Integrativo Derivativo (PID)

4.2.1 Control SI/NO (ON/OFF)

Dentro de los sistemas de control en lazo cerrado se trata del que menor complejidad entraña. Se basa en un control por realimentación el cual tan solo dispondría de dos posiciones, o encendido o apagado.

Los sistemas de control SI/NO presentan las siguientes características:

- La variable controlada tendrá una variación cíclica
- El controlador no tiene la capacidad de producir un valor exacto en la variable controlada
- Tiene un sencillo mecanismo de construcción

Una de las ventajas que aporta este sistema de control es que es muy simple, de bajo precio y de fácil instalación y mantenimiento. Todo esto lo convierten en los sistemas más utilizados en procesos en los cuales se requiere poca precisión.

En cuanto a las desventajas, la más notable de ellas es una mínima precisión que la hace no recomendable para procesos críticos. [19]

4.2.2 Control Proporcional

Se le llama sistema de control proporcional a todos aquellos sistemas que se realimentan de forma lineal. Es decir, la salida del controlador será proporcional a la magnitud del error.

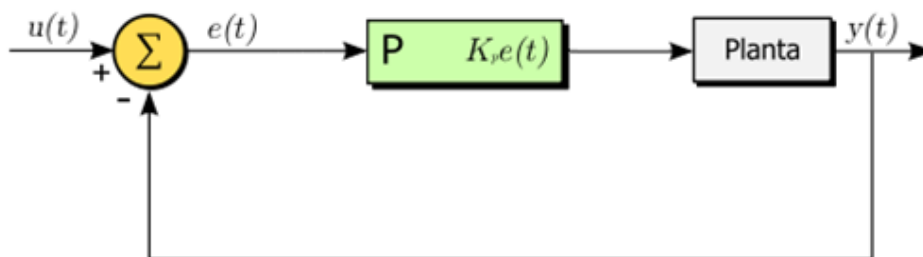


Figura 4.3 - Diagrama de Bloques de Controlador Proporcional [20]

El principio matemático que se esconde detrás del funcionamiento de un sistema de control proporcional es simple y se basa en *Ecuación 1*:

$$y(t) = Kp * e(t) + b$$

Ecuación 1 – Ecuación de Sistema de Control Proporcional

Siendo:

$y(t)$: Salida de nuestro sistema

Kp : Constante Proporcional

$e(t)$: Error de nuestro sistema

b : Condiciones Iniciales del sistema

El error de nuestro sistema se calcula sencillamente hallando la diferencia entre la señal de referencia (a donde queremos llegar) y nuestra salida del sistema (donde nos encontramos actualmente).

El funcionamiento teórico de este método de control se puede observar en la *Figura 4.4*. Donde se observan las diferentes salidas teóricas en función a las distintas K_p .

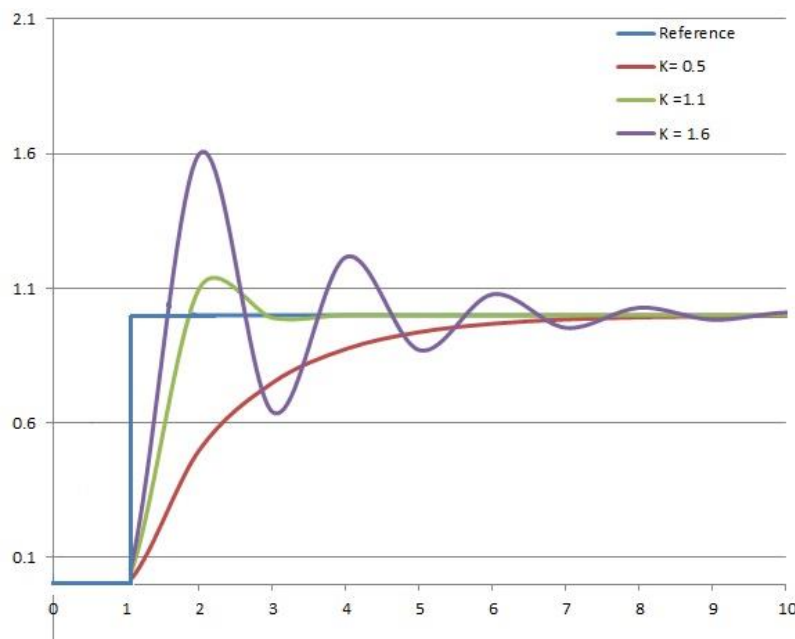


Figura 4.4 - Salidas del Sistema con diferentes K_p [21]

La constante proporcional (K_p) será la que se encargue de la sensibilidad de nuestro sistema. El valor de dicha constante será la que se encargue de la respuesta de nuestro sistema. Si le damos un valor muy bajo la respuesta al proceso será ampliamente estable en estado estacionario, no obstante la salida será “lenta” ante cambios del nivel de referencia, o como nos referiremos a él a partir de ahora, SetPoint. Con un bajo valor, el controlador no tendrá la suficiente agresividad como para realizar cambios rápidos en el proceso. Sin embargo, en contra posición, si la ganancia es demasiado alta la respuesta del sistema será más rápida pero provocará oscilaciones en la salida. Ambos efectos se pueden observar en la *Figura 4.4* [21] y en la *Figura 4.5* [22].

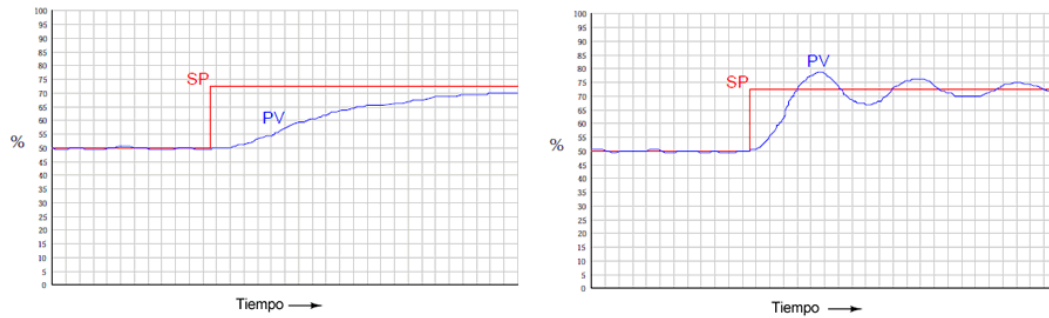


Figura 4.5 - Izq: Respuesta del Sistema con una K_p baja//Dcha: Respuesta del Sistema con una K_p alta [22]

4.2.3 Control Proporcional Integrativo

Se denomina sistema de control proporcional integrativo a aquellos sistemas que su realimentación tiene el objetivo de disminuir y eliminar el error en un estado estacionario. Bien sea por el funcionamiento común del mismo sistema o por oscilaciones provocadas por estímulos externos al mismo.

El control proporcional integrativo se basa en la suma de dos bloques como se puede observar en la Figura 4.6, el control proporcional anteriormente explicado y el control integrativo que explicaremos a continuación.

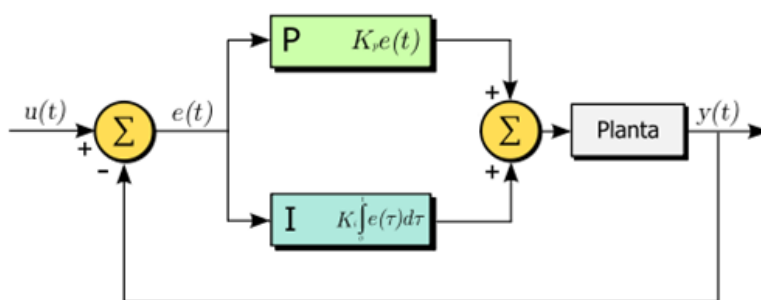


Figura 4.6 - Diagrama de Bloques de Controlador Proporcional Integrativo [20]

De forma que su ecuación será la ecuación del control proporcional ya explicada sumada a la ecuación del control integral explicada a continuación.

Control Integral

El control integral será el encargado de producir el ajuste del proceso basado en el error acumulativo. Es decir, se basará en la acumulación del error en el pasado para así poder cancelar cualquier perturbación del estado estacionario.

La ecuación que rige este proceso de control es la *Ecuación 2*:

$$y_i(t) = K_i * \int_0^t e(\tau) d\tau$$

Ecuación 2 – Ecuación de Sistema de Control Integrativo

Siendo:

$y_i(t)$: Salida del Control Integral

K_i : Constante Integrativa

$e(t)$: Función del error

El funcionamiento teórico de este sistema se puede observar en la *Figura 4.7* donde se observan las salidas para ciertos valores de las constantes. En azul tendremos el escalón, encargado de poner en marcha el movimiento dado el error que provoca al crear un error distinto de cero. En color negro tenemos la salida con una constante integral igual a dos, tal y como se puede observar se producen sobreoscilaciones que hacen que el sistema no sea el más recomendable. También en rojo observamos la salida con una baja constante de integración, en la cual no existen sobreoscilaciones no obstante el tiempo de respuesta se el más lento. Por último en color verde tenemos una señal con un control PID adecuado, este tipo de control se explicará más adelante.

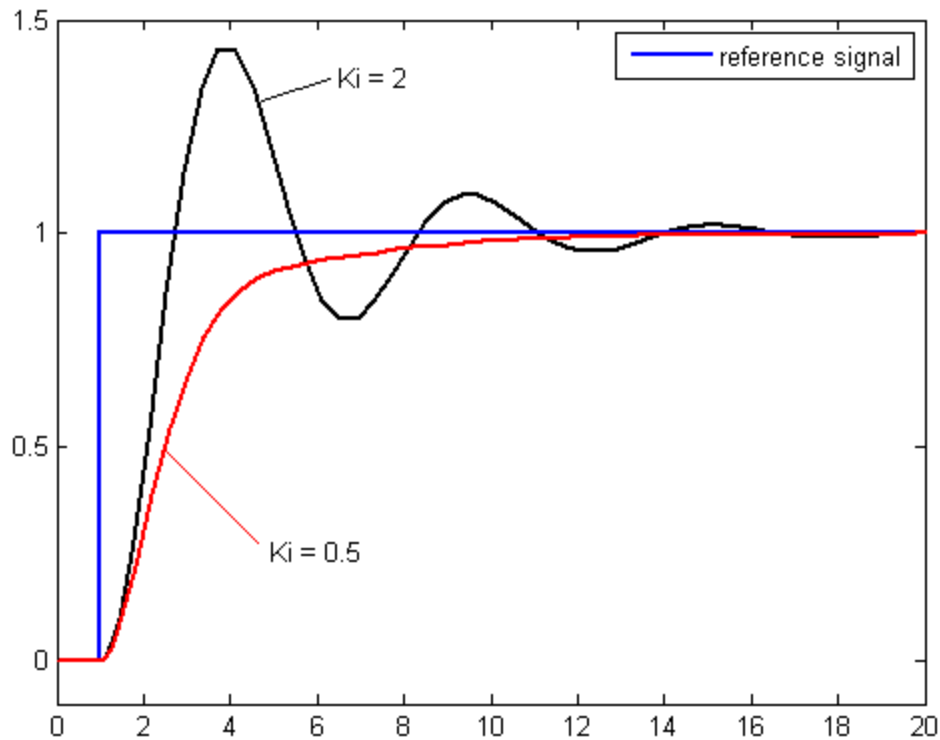


Figura 4.7 - Salidas del Sistema con diferentes K_i [23]

El control integral actuará cuando exista una desviación entre la variable y la señal de referencia. El funcionamiento del mismo se explica debido a la naturaleza de la función integral, esta misma consiste en el cálculo del área bajo una curva delimitada por los extremos, por lo tanto de esta forma, lo que realizará el control integral será calcular el área de la curva correspondiente al interior de la curva del error desde el inicio del proceso hasta el tiempo actual y tras esto será multiplicada por una constante, que denominaremos Constante Integral (K_i). El controlador se encargará de hacer que el error acumulado tienda a cero, llegando a cancelar así las oscilaciones, y si estas no existieran, evitarlas. [24]

4.2.4 Control Proporcional Derivativo

Se denominan sistemas de control proporcional derivativo a aquellos sistemas que tienen como función la corrección del error de forma proporcional con la misma velocidad que se produce, evitando así que el error se incremente.

El control proporcional derivativo se basa en la suma de dos bloques como se puede observar en la *Figura 4.8*, el control proporcional anteriormente explicado y el control derivativo que explicaremos a continuación.

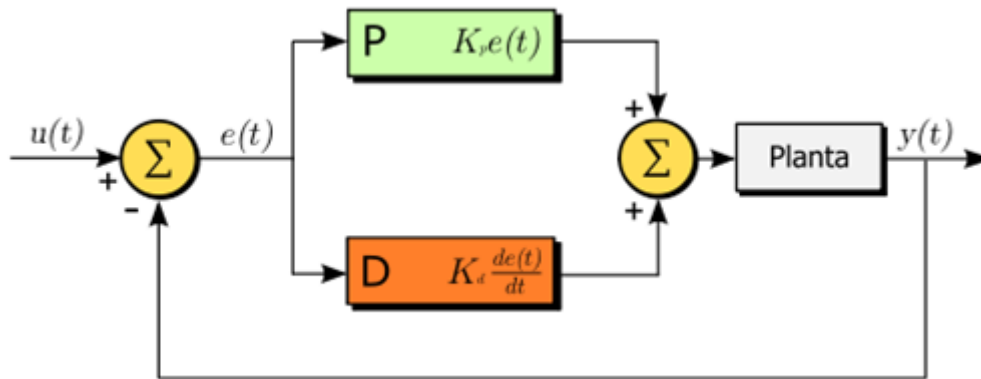


Figura 4.8 - Diagrama de Bloques de un Sistema de Control Proporcional Derivativo [20]

De forma que su ecuación será la ecuación del control proporcional ya explicada sumada a la ecuación del control derivativo explicada a continuación.

Control Derivativo

El control derivativo será el encargado de mantener el error al mínimo corrigiéndolo de forma proporcional con la velocidad a la que se produce. Es decir, se basará en la derivada con respecto al tiempo.

Explicado de una forma más sencilla, el control derivativo se basa en una predicción del error en un punto de un tiempo futuro. No obstante al ser un punto futuro, obviamente, no sabemos su valor exacto y se recurrirá a una estimación. Se realizará la derivada en el instante de tiempo actual de forma que obtengamos la tangente en ese instante de tiempo y luego se tomará el valor que esa recta tome en ese instante de tiempo futuro como estimación tal y como se observa en la *Figura 4.9*.

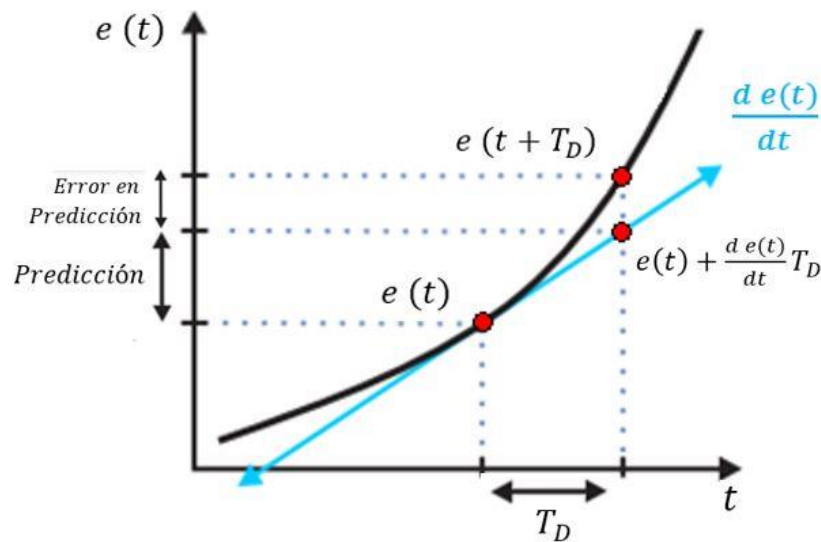


Figura 4.9 - Explicación Control Derivativo en un Instante de tiempo

En base a esta idea y uniéndola con el control proporcional, se deduce que la fórmula del control derivativo corresponde a la Ecuación 3, siendo "b(t)" la salida del sumador:

$$b(t) = K_p \left[e(t) + \frac{de(t)}{dt} T_D \right]$$

$$b(t) = K_p e(t) + K_p T_D \frac{de(t)}{dt}$$

$$b(t) = K_p e(t) + K_D \frac{de(t)}{dt}$$

Ecuación 3 - Ecuación de Sistema de Control Proporcional Derivativo

De esta forma se deduce la Ecuación 4.

$$K_D = K_p T_D$$

Ecuación 4 - Ecuación de la constante Derivativa

La Figura 4.10 nos ofrece un ejemplo de las salidas que generará un control proporcional derivativo. Si nos fijamos en la función de color negro se observa como si K_D es grande, el tiempo de acción de la derivada será grande y existirá una gran inestabilidad en el sistema, pues mayor será el error en la predicción. Si por el contrario nos fijamos en la función roja se puede ver como oscila en torno al SetPoint.

Un valor óptimo en el sistema de ejemplo será el que toma la función verde pues retorna la variable al SetPoint con unas mínimas oscilaciones.

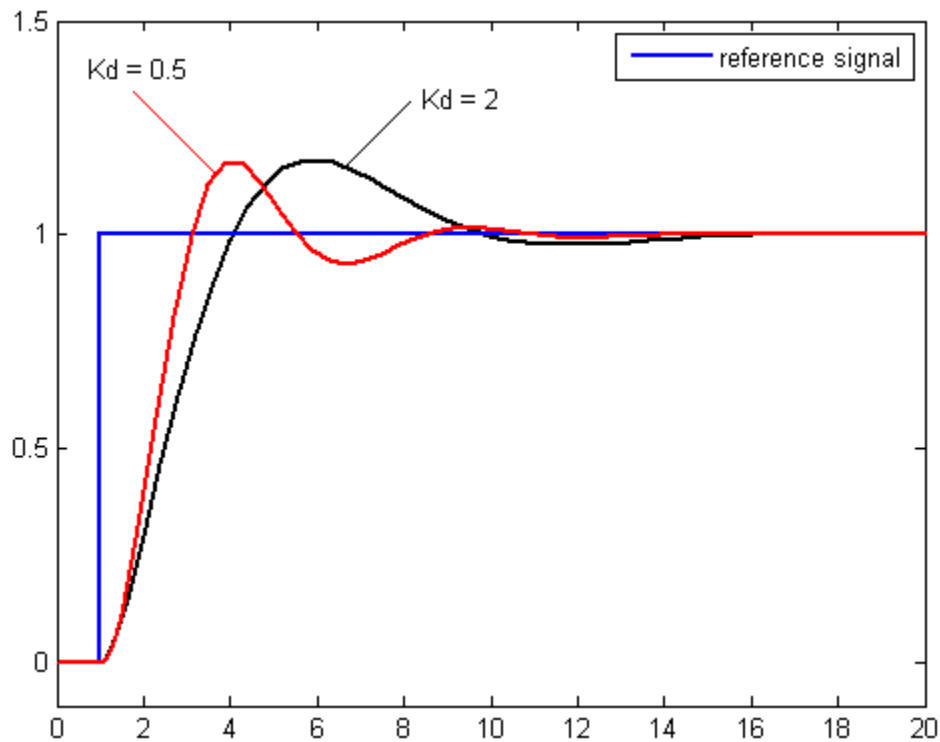


Figura 4.10 - Salidas de un Sistema de Control Proporcional Derivativo

Esta opción de control será adecuada cuando exista un retraso entre el dispositivo controlado y el sensor utilizado en la realimentación. [20] [25]

4.2.5 Control Proporcional Integrativo Derivativo (PID)

Se denominan controles PID a aquellos sistemas cuya corrección de la función del error conste de un control proporcional, un control integrativo y un control derivativo. De forma que utilizándolo de una manera correcta y hallando el mejor valor para sus constantes se pueden llegar a un control de posición óptimo.

El diagrama de bloques de dicho control será el descrito en la *Figura 4.11*:

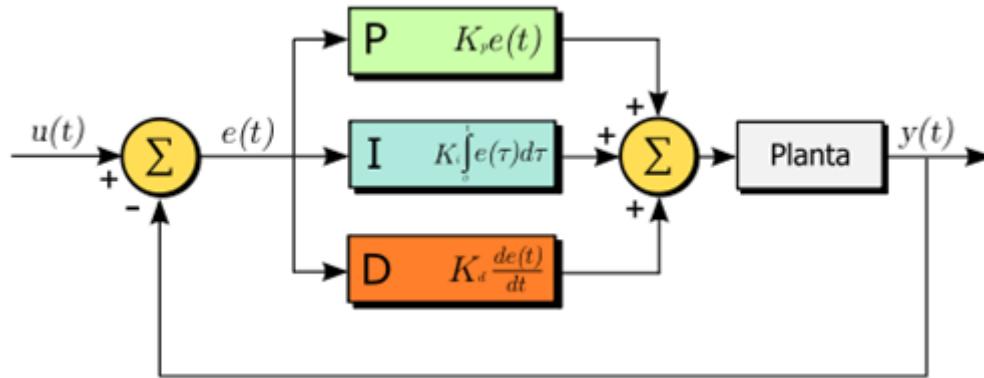


Figura 4.11 - Diagrama de Bloques de un Controlador PID [20]

De esta forma la ecuación del controlador PID será la *Ecuación 5*:

$$y(t) = K_P e(t) + K_I \int_0^t e(\tau) d\tau + K_D \frac{de}{dt}$$

Ecuación 5 - Ecuación de Sistema Controlador Proporcional Integrativo Derivativo

Tal y como se puede observar será la suma de la ecuación de cada tipo de control.

4.2.5.1 Ajuste de parámetros del PID

El ajuste de los parámetros de un controlador PID es vital para su buen funcionamiento. Esto se debe a que si estos parámetros se escogen de forma incorrecta el proceso a controlar se podrá volver inestable y donde se buscaba encontrar soluciones solamente se generarán problemas mayores. Un mal funcionamiento de un sistema de control puede llegar a ocasionar daños en los mecanismos que este controle por lo tanto es de vital importancia encontrar unos valores adecuados para las constantes.

El comportamiento óptimo ante un cambio de SetPoint varía en función de la aplicación que se le este dando a nuestro controlador. Para algunos procesos será conveniente eliminar la parte derivativa del control y para otros un control proporcional será suficiente.

A la hora de ajustar un lazo de control existen muchos métodos. Generalmente el más eficaz requiere el desarrollo de un modelo de proceso para luego poder elegir las constantes en función a la salida que dicho modelo presente. También existe el ajuste manual, no obstante, puede llegar a ser muy ineficiente si se hace sin cuidado. La elección de un método u otro dependerá principalmente del tiempo de respuesta del sistema y de si el lazo puede ser desconectado para ajustarlo.

4.2.5.2 Transformada Z

La Transformada Z en el ámbito del procesamiento de señales nos convertirá una señal definida en el dominio del tiempo discreto en una representación en el dominio de la frecuencia. Es por ello que trabajaremos con esta herramienta matemática, pues nuestro control PID se encuentra implementado mediante la utilización de un microcontrolador, es decir, funciona de forma discreta dada la naturaleza de todos los microcontroladores.

La Transformada Z puede ser definida como una transformada unilateral o bilateral. En nuestro caso al trabajar siempre con señales causales dada la naturaleza real de nuestros sistemas definiremos la Transformada Z Unilateral como la *Ecuación 6*.

$$X^+(z) = Z^+\{x[n]\} = \sum_{n=0}^{\infty} x[n]z^{-n}$$

Ecuación 6 - Ecuación de la Transformada Z Unilateral

Siendo la única diferencia entre la transformada unilateral y la bilateral que esta última tendrá en cuenta instantes de tiempo negativos.

La Región de Convergencia (ROC) se define la región donde la transformada-z existe. La ROC será una región del plano complejo en la cual el sumatorio de todos los términos de la serie convergerá. Las propiedades de la ROC son las siguientes que nos interesan para el análisis de nuestra señal serán:

- La ROC nunca contiene ningún polo
- Si $x[n]$ es causal la ROC se extenderá hacia afuera desde el último polo

4.3 Teoría Aplicada al Prototipo

En el caso que nos ocupa, el control de posición de motores en DC, la salida $y(t)$ nos la proporcionaría el encoder, mientras que el error sería la diferencia entre la posición a la que queremos llegar y la posición en la cual nos encontramos. Las técnicas de control que utilizaremos serán el control ON-OFF y PID.

Las condiciones iniciales siempre serán nulas al inicio debido a que siempre comenzaremos en una posición 0 pero tras el primer movimiento, si decidimos avanzar a la posición 8000, las condiciones iniciales de ese sistema en el segundo movimiento serán igual a la posición en la cual se encuentre el sistema, es decir, 8000.

En el siguiente capítulo se pretende implementar un sistema de control PID. Para ello se comenzará paulatinamente hasta alcanzar el objetivo deseado. En primer lugar experimentaremos el control en lazo abierto y tras esto pondremos en práctica cada uno de los distintos tipos de control en lazo cerrado que han sido explicados en este capítulo.

5 Resultados Experimentales

Tras todas las investigaciones realizadas, tanto históricas como tecnológicas, en este capítulo se relatarán los experimentos llevados a cabo a lo largo de este proyecto, relatando que problemas se han encontrado y como se han conseguido resolver para así llegar a obtener un resultado y a una serie de conclusiones.

En primer lugar hablaremos del montaje de la electrónica de la prótesis y como se comprobó su buen funcionamiento. Tras esto comenzaremos los experimentos analizando primero nuestro sistema en lazo abierto para luego pasar al lazo cerrado, donde se explicará todo el software y hardware implementado para llevar a cabo las pruebas experimentales de los distintos tipos de control en lazo cerrado. Por último se explicará cómo se montó el sistema protésico completo.

5.1 Montaje de la electrónica de la Prótesis

A la hora de realizar el montaje de la electrónica que incorpora la prótesis hubo varios problemas, todos ellos debidos al reducido tamaño del encoder que se sitúa en la parte trasera del motor. Debido al número de salidas del encoder y su pequeño área los diseñadores del chip decidieron utilizar un paso, distancia entre pines, de 2mm en vez de la comúnmente más utilizada de 2,54mm.

Esto originó que a la hora del montaje no se dispusieran de pines de ese tamaño al igual que tampoco se disponía de buses del tamaño adecuado. Se terminó comprando los pines pero por un fallo humano no se pensó en los buses, lo cual llevo a improvisar un bus con los materiales de los cuales se disponía. A su vez los pines del paso adecuado que se compraron al ser macho-macho obligaban a buscar cables hembra macho, los cuales en la parte de la hembra tienen un mayor grosor. Por todo esto se comprobó el grosor del conector hembra de los cables. Para lo cual como se puede observar en la *Figura 5.1* se obtuvo una substancial minimización del grosor del mismo.

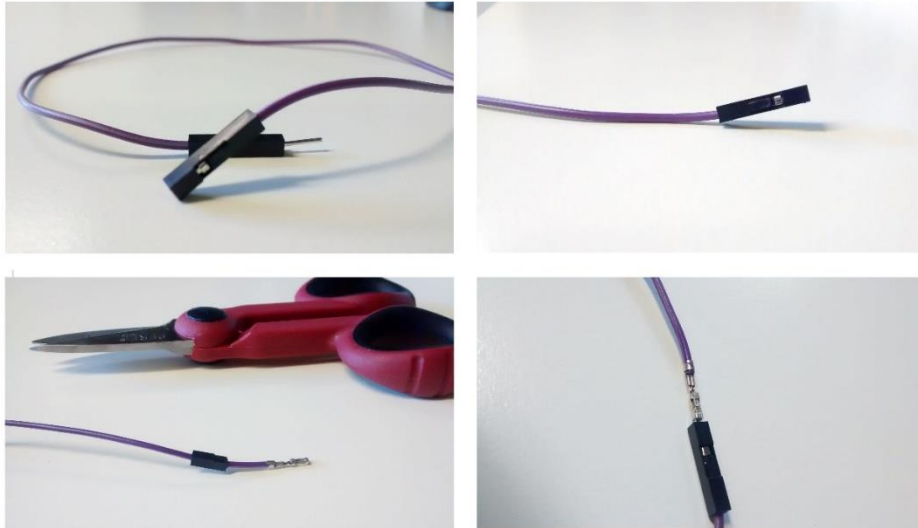


Figura 5.1 - Solución del Bus

Gracias a esta solución se pudo implementar un bus de una forma correcta, no obstante, como habrán podido imaginar, surgía otro problema. El conector se quedaba desnudo, y por lo tanto al introducirlos todos en el codificador se producirían contactos indeseados e interferencias en las señales que transportan los mismos y como consecuencia de ello un mal funcionamiento del dispositivo. Por consiguiente se llegó a la conclusión de que necesitaban ser aislados de forma efectiva.

En un primer intento se realizó con cinta aislante, no obstante debido al pequeño tamaño de los conectores y la escasa superficie de contacto la cinta aislante necesitaba de más de una vuelta para su correcto agarre, lo cual aumentaba el volumen del cable complicando la situación.

El segundo intento sería el definitivo, debido a la utilización de recubrimientos de cable de plástico termoretractil del diámetro adecuado. Gracias a ellos y con la ayuda de un estañador para aplicarle el calor necesario se logró llegar a un resultado aceptable. Tal y como se puede observar en la siguiente *Figura 5.2*.

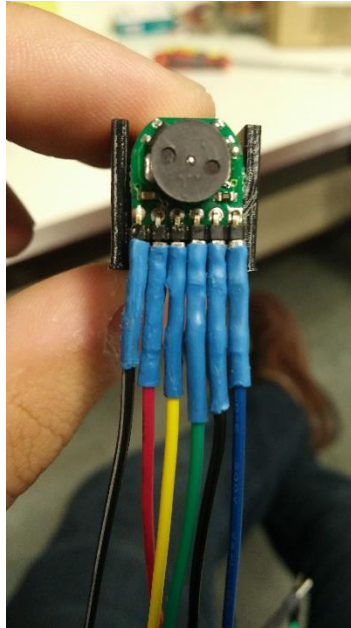


Figura 5.2 - Resultado final del Bus

Una vez solucionado este problema, se decidió a hacer una primera prueba de funcionamiento. Dicha prueba consistía en hacer girar el motor mediante los pines de alimentación y comprobar la correcta salida de las señales. El resultado obtenido fue completamente insatisfactorio. En la *Figura 3.3* se puede observar como el resultado de la pantalla del osciloscopio era completamente erróneo.

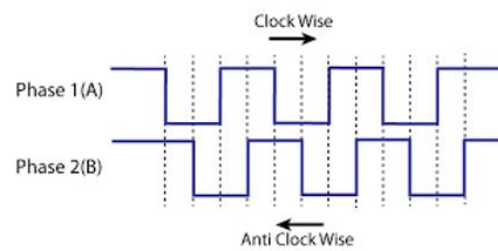
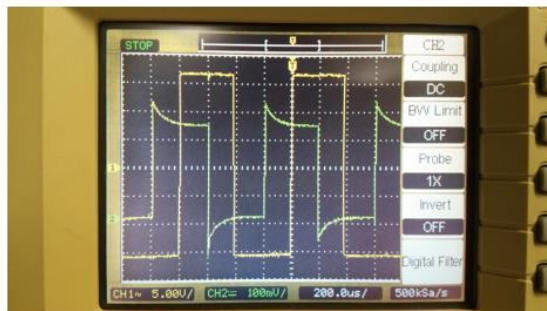


Figura 5.3 - Izq: Funcionamiento Incorrecto del Sistema. Drch: Funcionamiento Deseado del Sistema

Según se puede observar en la *Figura 3.3* tanto en el canal 1 como el canal 2 se obtienen unos resultados absurdos, debido claramente a un mal funcionamiento de algún componente del circuito.

Tras una investigación en la cual se analizaron todos los componentes uno a uno para así comprobar su correcto funcionamiento se descubrió que el causante del error era el bus de fabricación manual. Observando la *Figura 5.4* se observa que en el inicio del cable

queda una pequeña parte sin recubrimiento de plástico termofusible. Debido a este desliz se generaba un cortocircuito que inutilizaba el proyecto.

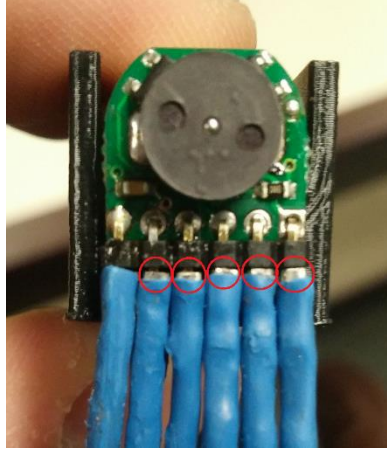


Figura 5.4 - Error en Bus

De forma que una vez corregido el bus, el resultado arrojado por la pantalla del oscilador se volvía coherente y correcto, obteniendo la señal que se muestra en la *Figura 5.5*. En la cual se observa claramente las similitudes con la función buscada.

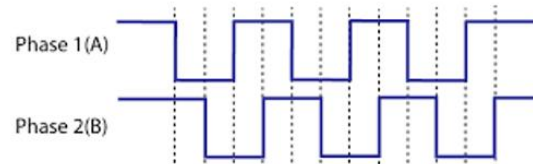
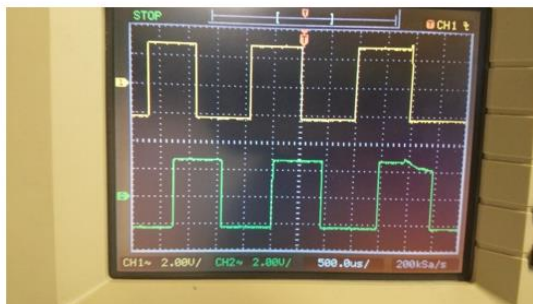


Figura 5.5 - Funcionamiento Correcto del Sistema

Una vez comprobado el correcto funcionamiento del encoder se dieron por finalizadas las comprobaciones y se pasó a realizar los primeros experimentos.

5.2 Lazo abierto

Aunque el objetivo final del proyecto sea el control de posición de una mano mediante lazo cerrado en una primera instancia se planteó el comenzar con las pruebas prácticas de la forma más sencilla posible, es decir, implementando un control en lazo abierto. A su vez este experimento se consideró una forma sencilla de tomar un primer contacto con el driver DRV8838 de forma individual, es decir, sin incorporarlo junto al encoder, para así obtener una base para que los experimentos posteriores resulten más fluidos.

Este sistema funcionará en base a un contador que aumenta de forma proporcional cada vez que el bucle principal del programa es ejecutado. De forma que el programa se encargará de que cuando ese contador sea mayor que una constante, cuyo valor decidimos nosotros, se cambie el sentido del giro del motor para así lograr una flexión y extensión completa de un dedo gracias al mecanismo de la prótesis.

Para realizar este experimento se utilizó el Driver DRV8838 y se implementó el circuito que puede observarse en la *Figura 5.6*.

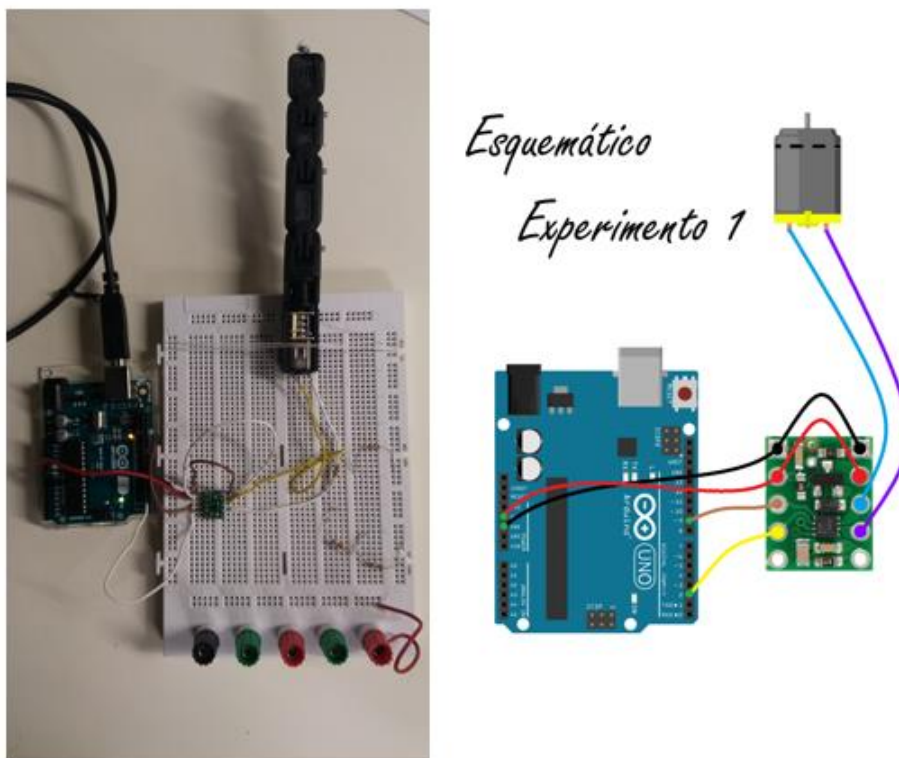


Figura 5.6 - Esquemático Experimento 1

Como se puede observar en la imagen, con el driver utilizado se logra que la implementación necesaria para poder llevar a cabo el control en lazo abierto sea realmente simple. En la siguiente *Figura 5.7* se puede observar la distribución de pines del driver.

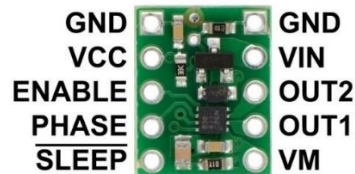


Figura 5.7 - Distribución de Pines Driver

Como se puede observar tanto en el esquemático como en la foto de la implementación real, para simplificar el circuito se comenzó alimentándolo todo desde el microcontrolador, pese a las contraindicaciones del fabricante debido a que esto puede ocasionar interferencias entre pistas. El pin ENABLE es donde deberá ir nuestra señal PWM generada por el microcontrolador mientras que el pin PHASE es donde se decidirá el sentido de giro.

En la parte superior de la *Figura 5.8* se puede observar que la señal en el tiempo se trata de un diente de sierra, siendo tan solo un contador de tiempo tal y como hemos explicado anteriormente, mientras que en la parte inferior de la misma figura se observa como cambiará el sentido del giro para así realizar la flexión-extensión del dedo.

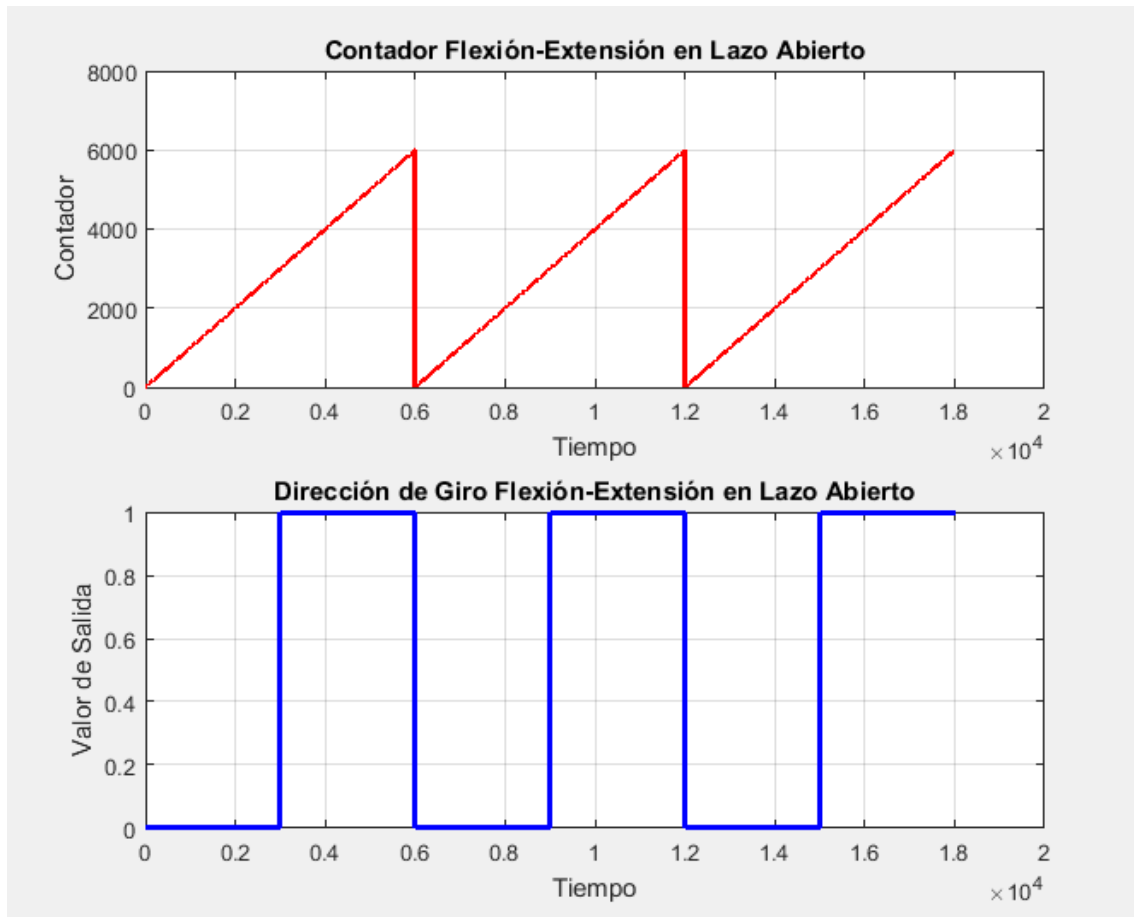


Figura 5.8 - Gráfica Contador y Dirección del Lazo Abierto

Una vez realizado el código y el circuito, se probó su funcionamiento mediante la incorporación del dedo. El resultado obtenido fue el esperado, podíamos controlar la posición del dedo pero de una forma inexacta. En la primera flexión-extensión el dedo hacía un trabajo de carácter aceptable, no obstante a partir de la segunda flexión-extensión y debido a los desajustes ocasionados a lo largo de la primera flexión-extensión el dedo sufría un desajuste que aumentaba de forma substancial en la tercera flexión-extensión.

Debido a la inexactitud que los controles de lazo abierto provocan en este proyecto, quedo descartada esta forma de control de posición y se continuó investigando en la línea del lazo cerrado, es decir, incorporando el encoder descrito en la sección anterior.

5.3 Lazo Abierto con Encoder

Antes de seguir adelante con el control en lazo cerrado se observó las posibilidades que tenía el experimento realizado en lazo abierto si se le sumara un encoder al motor. Gracias a este elemento podríamos sensar la posición del dedo contando el número de vueltas que da el motor y de esta forma, viéndolo de forma esquemática tal y como se muestra en la *Figura 5.9* tendremos la entrada a un sistema con su respectiva salida y por lo tanto podremos calcular la respuesta en frecuencia del mismo y así saber cuáles son sus polos y sus ceros.

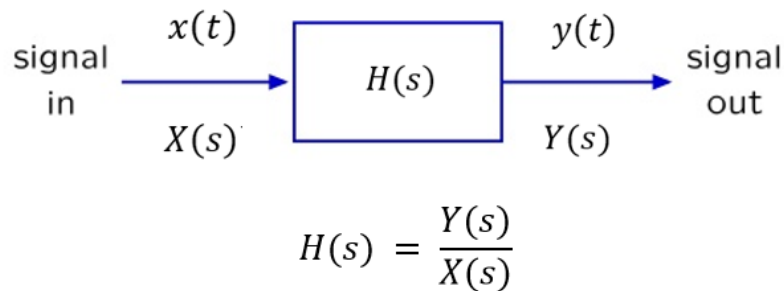


Figura 5.9 – Teoría de Sistemas

El sistema se compondrá del Arduino, el driver, el motor y el mecanismo del dedo. Este mismo sistema estará diseñado para que al introducirse en el mismo una señal en forma de diente de sierra, tal y como hemos visto en la *Figura 5.8*, la respuesta sea un movimiento de flexión y extensión del dedo, que se observará gracias a la salida del encoder.

5.3.1 Elección de salida correcta

Llegado a este punto y con el fin de caracterizar nuestro sistema de la mejor manera posible se plantea un experimento que consistirá en la toma de los datos de un ciclo del sistema, es decir, una flexión y una extensión del dedo, de tres formas distintas para

observar si existen variaciones a la salida. Estas tres formas serán: la salida del sistema sin la carga mecánica que ofrece el mecanismo del dedo, la salida del sistema con el mecanismo del dedo utilizando tensores (los encargados de ofrecer una fuerza inversa a la del motor para que así la extensión del dedo se realice) degradados con un amplio uso, y por último la salida del sistema con los tensores recién reemplazados.

Analicemos estas tres salidas de forma individual, siendo la entrada para todas ellas la mostrada en la *Figura 5.10*.

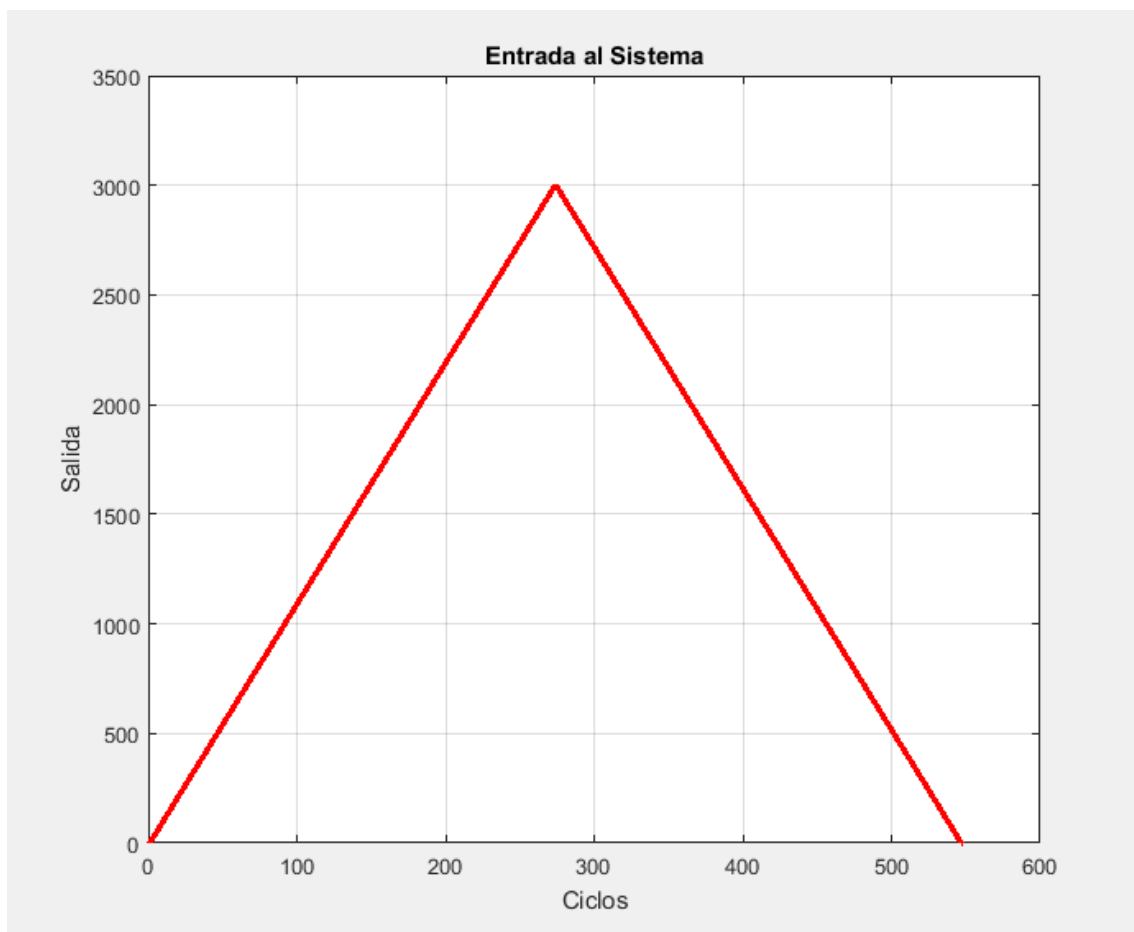


Figura 5.10 - Entrada al Sistema

Salida del Sistema Sin Carga Mecánica

En primer lugar observamos en la *Figura 5.11* la salida del sistema sin la carga mecánica. Se puede observar como el sistema aporta dos efectos no lineales en los puntos de puesta en marcha y cambio de sentido del motor, como sugiere la curva de la *Figura*

5.11 entre los ciclos 270 y 290. Esto se deberá al tiempo en el cual la inercia de los motores tarda en ponerse en marcha y en cambiar de sentido. Sin embargo estos efectos se pueden modelar como retardos por lo que podríamos obtener un modelo lineal de nuestro sistema como se hará más adelante.

En la curva de la *Figura 5.11* es donde se encuentra el pico de la máxima posición a la que se llega, siendo esta posición muy cercana a 15.000, en concreto 14.800.

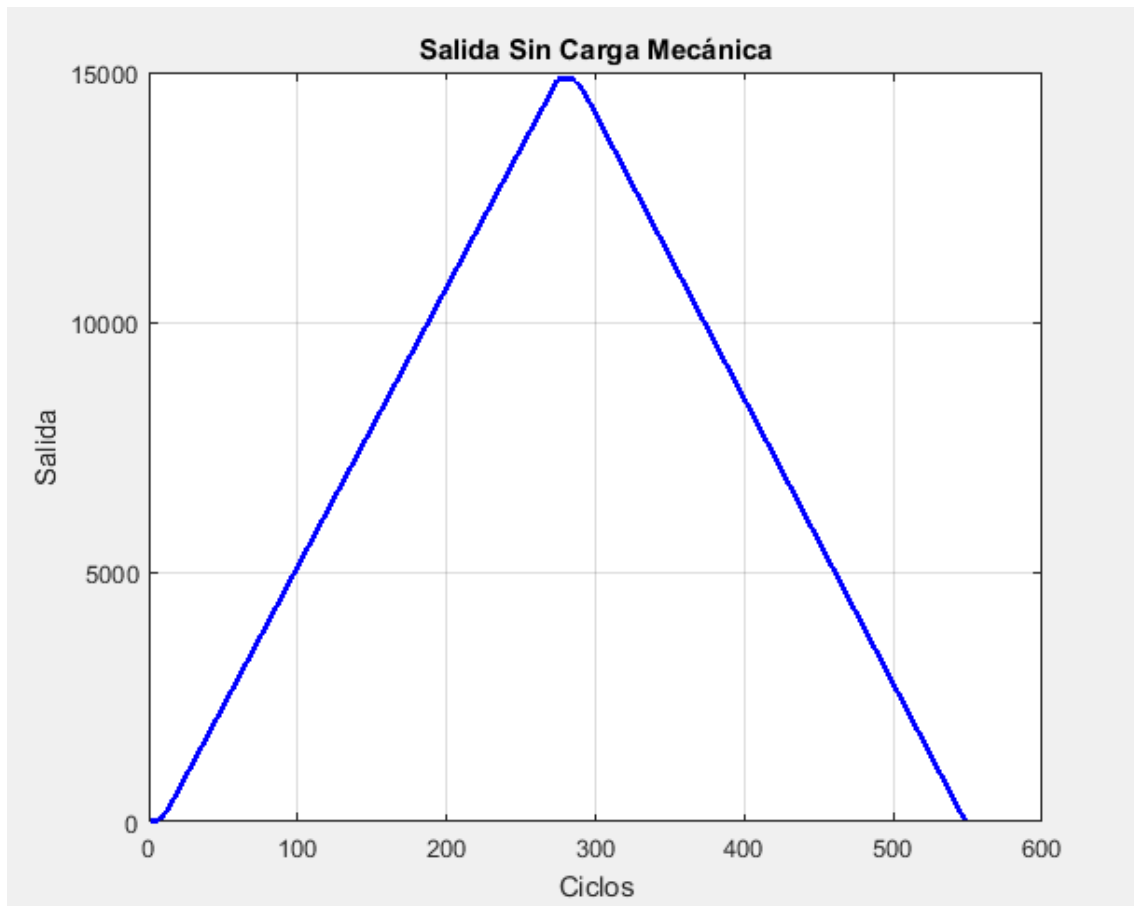


Figura 5.11 - Salida del sistema Sin Mecanismo

A su vez también se ve como comienza en cero y termina en cero también, algo que señala el buen funcionamiento de este control en lazo abierto.

Salida del Sistema con Mecanismo y gomas degradadas

Se pretende demostrar que el desgaste físico de las partes elásticas del mecanismo también causa efectos que afectan al sistema global.

Tal y como se esperaba y se observa en la *Figura 5.12* el sistema el sistema muestra efectos no lineales al comienzo y al cambio de dirección del motor, al igual que en el caso anterior, no obstante se observan claras diferencias.

La más vistosa es que tras el ciclo el sistema ya no acaba en 0, sino que acaba en un número negativo. La otra diferencia es que el pico de la salida se encuentra en la posición 14.326.

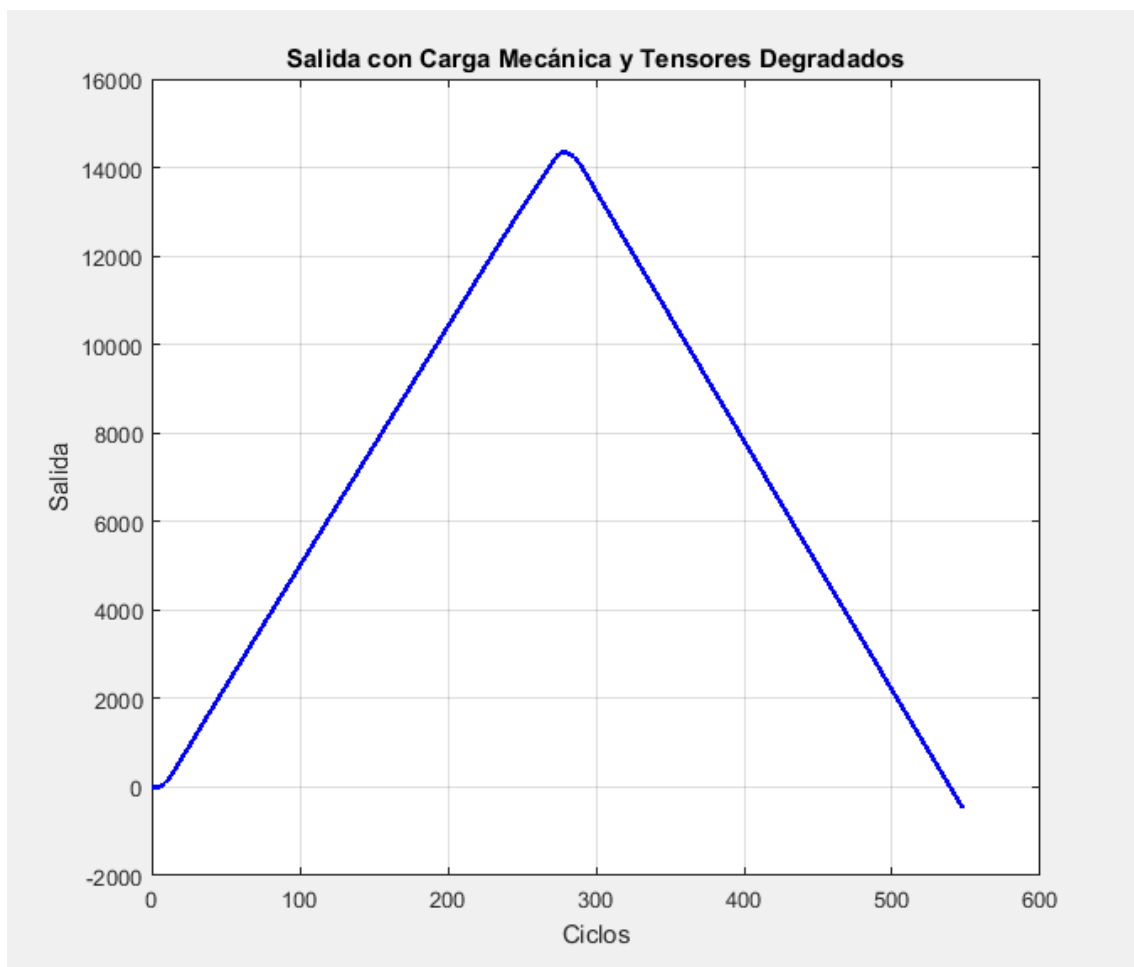


Figura 5.12 - Salida del Sistema con Mecanismo y Gomas Degradadas

El causante de estas diferencias respecto a la *Figura 5.11* no podía ser otro que el mecanismo del dedo.

Como ya se sabe el dedo se contrae gracias a la tensión que ejerce un hilo, que a modo de tendón, atraviesa todas las piezas que lo componen. Gracias a esta tensión el dedo se flexiona. Sucede que debido a un desajuste, el movimiento del dedo al flexionarse ha llegado a un punto en que el motor no puede tirar más del hilo porque el dedo ya está contraído del todo, quedándose así en una posición máxima menor que si el motor estuviera libre, esto es lo que causa que al liberar el dedo, en el movimiento de extensión, se avance todo lo que el motor está preparado para avanzar dado que al final del movimiento de extensión no existe una fuerza que le impida al motor seguir girando.

No escogeremos esta salida para calcular la $H(s)$ de nuestro sistema, debido a que el sistema está pensado para que los elásticos se reemplacen con frecuencia.

Salida del Sistema con Mecanismo y gomas no degradadas

Observando la *Figura 5.13* se puede ver que el cambio con respecto a la *Figura 5.12* es substancial debido a que el ciclo termina muy cercano al cero y el pico de posición está más próximo al del motor en caso de carecer del mecanismo del dedo, ya que en este caso el pico se encuentra en la posición 14.556.

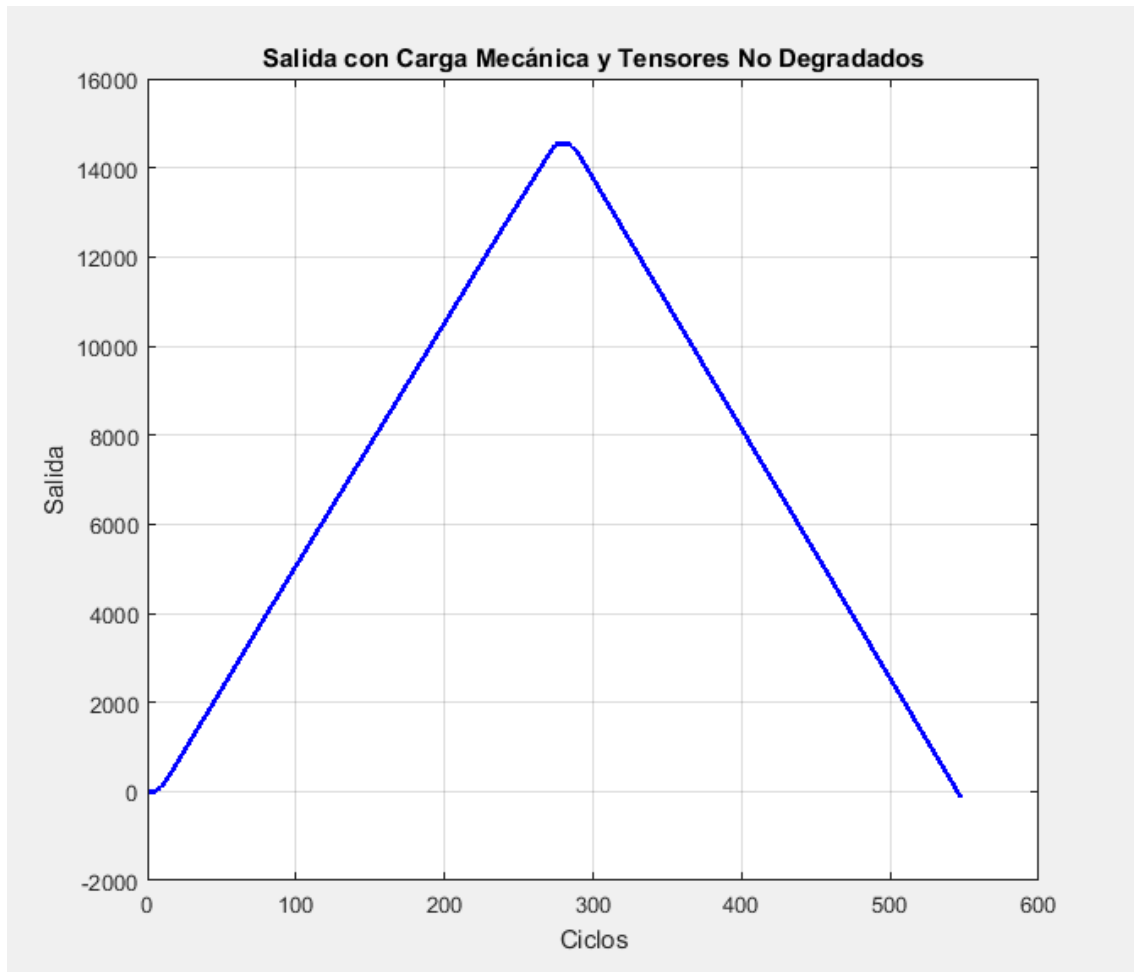


Figura 5.13 - Salida del Sistema con Mecanismo y Gomas no Degradadas

El motivo de este cambio con respecto al hecho de tener las gomas degradadas podría radicar en que unas gomas nuevas ejercen más fuerza a la hora de tensar el hilo y por lo tanto al motor le cuesta más recogerlo y no se llegaría a parar tanto la rotación del motor, por lo cual no existiría apenas desajuste y al bajar se aproximaría más a cero.

Esto es tan solo una hipótesis que quedaría por demostrar, no obstante queda claro que hay una mejora substancial y por consiguiente, esta será la salida que utilizaremos para calcular el $H(s)$ de nuestro sistema. A su vez el prototipo, como se ha comentado antes, está pensado para el reemplazo diario de los componentes elásticos, por lo tanto esta elección es coherente con el uso de la prótesis.

Conclusiones

En conclusión de los tres apartados:

- El lazo abierto con el motor libre es exacto, no obstante al introducir el mecanismo del dedo se vuelve más inexacto, debido a los efectos de carga que realiza el mecanismo, el hilo y los tensores.
- No se puede concluir que el desgaste de las gomas afecte negativamente al funcionamiento del prototipo, aunque sí que existe un empeoramiento en la exactitud.
- Para el cálculo de la $H(z)$ del sistema utilizaremos *Figura 5.13*, es decir, la gráfica con el mecanismo del dedo y las gomas no degradadas.

5.3.2 Cálculo de $H(z)$ del Sistema en Lazo Abierto

Una vez escogida la salida se realizará el cálculo de la $H(z)$ mediante Matlab. Gracias a su APP "System Identification".

El procedimiento para calcular el sistema será el siguiente, este programa solamente funcionará para sistemas que sean LTI, Gracias a esta aplicación de Matlab se puede lograr que al introducirle una señal de entrada y la de salida del sistema te calcule de forma automática la respuesta en frecuencia del sistema. Se puede elegir si hacerlo de forma continua o de forma discreta. Dada la naturaleza de nuestro circuito de control tomaremos siempre en tiempo discreto. Cuando una señal se encuentra en tiempo discreto, la forma de calcular su transformada es recurriendo a la transformada Z, ya explicada de forma breve en el apartado 4.3.5.2.

De esta forma, nosotros introducimos la entrada y la salida que se observa en la *Figura 5.14*.

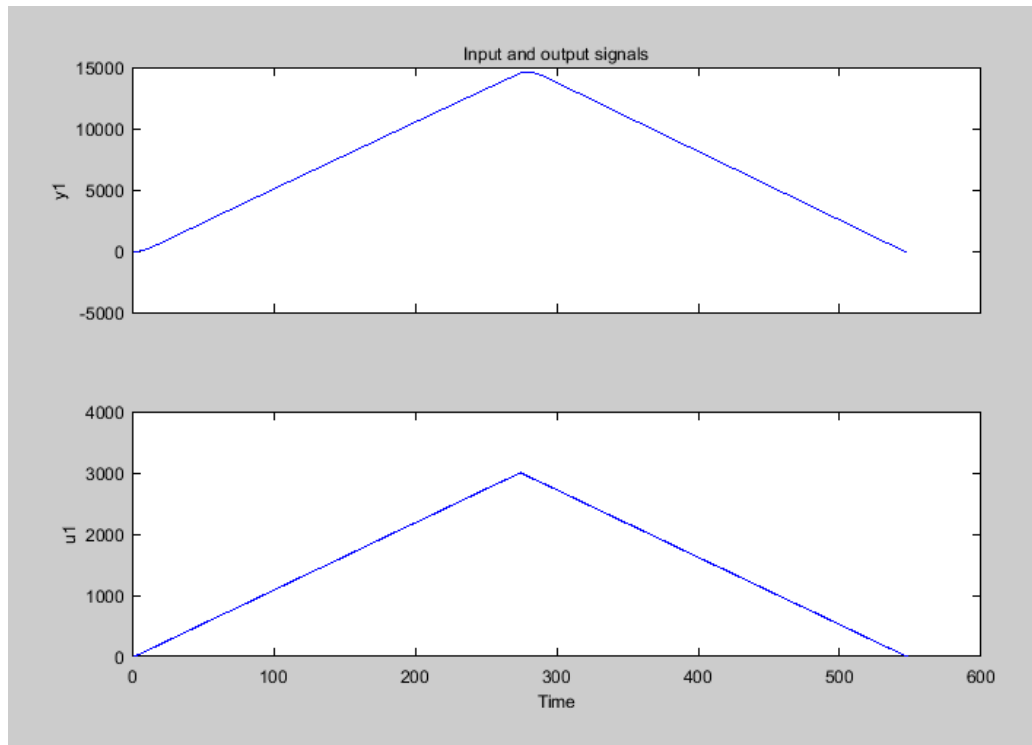


Figura 5.14 – Input (abajo) y Output (arriba) Introducidos en el Programa

Viendo la salida que genera a la entrada se decide probar a calcular la $H(z)$ como si fuera un sistema de segundo orden, es por ello que se comienza indicándole que el sistema tiene:

- 1 polo y 1 cero : 97.1% de semejanza
- 2 polos y 1 cero : 97.13% de semejanza
- 1 polo y 1 cero : 97.09% de semejanza

Dadas las semejanzas escogeremos la opción número dos, 2 polos y 1 cero. Esto se debe a que es la que más se aproxima a la idea existente. La función de transferencia que es devuelta es la siguiente:

$$H(Z) = \frac{0.2333 z^{-1}}{1 - 1.756 z^{-1} + 0.8031 z^{-2}}$$

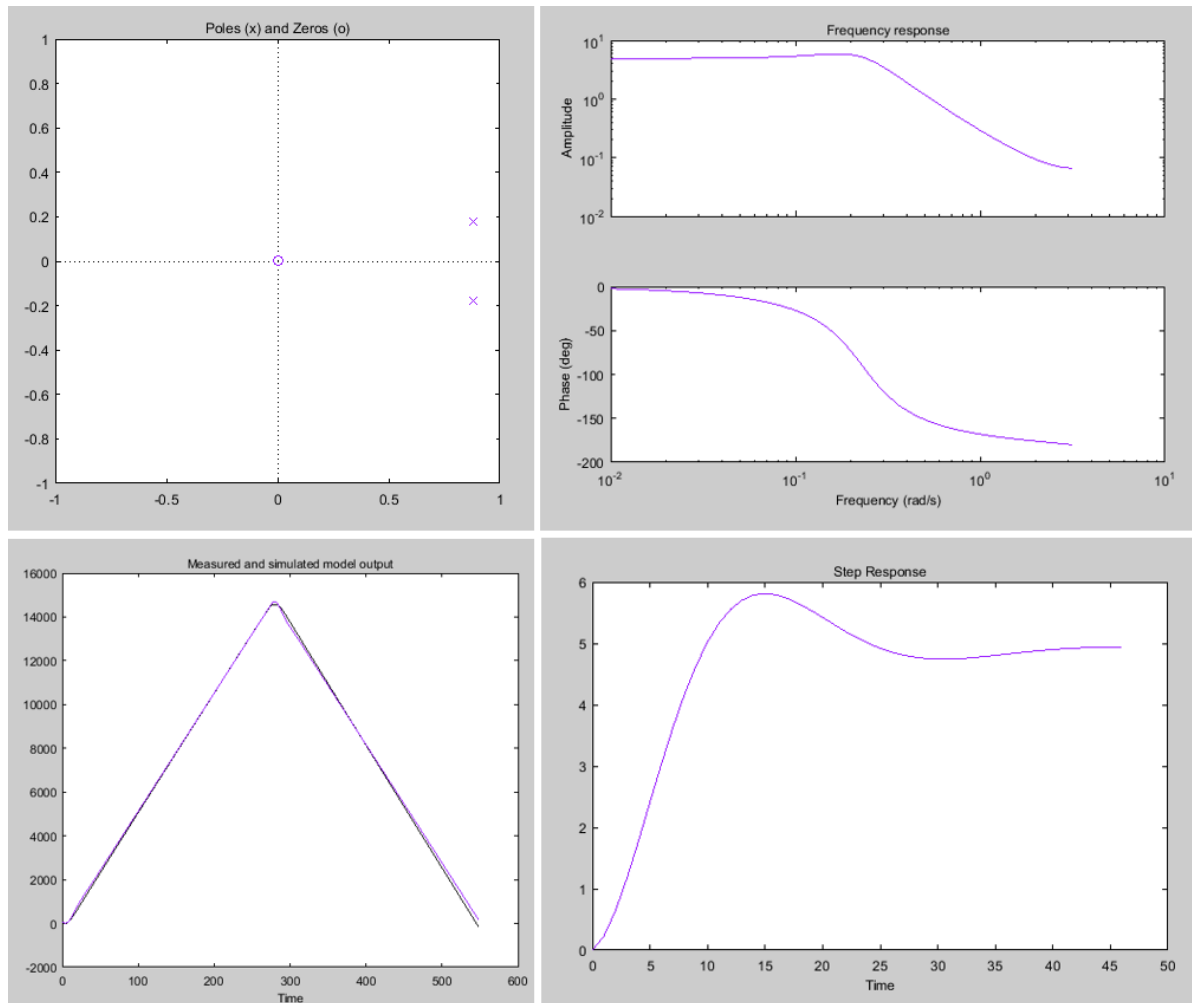


Figura 5.15 - Información del Sistema Escogido

También nos aporta los datos que se pueden observar en la *Figura 5.15*. Comenzando por el diagrama de polos y ceros arriba a la izquierda y siguiendo a su derecha con la respuesta en frecuencia. En la parte de abajo se puede observar la diferencia entre las salidas, estas diferencias pueden ser causadas debido a los efectos no lineales. Por último se puede observar abajo a la derecha la respuesta al escalón.

La respuesta al escalón es muy característica y visual ya que se observa cómo se asemeja esa función a lo obtenido con el control proporcional que se podrá ver en el apartado 5.4.1.

Por último el programa ofrece una comprobación para que puedas observar cómo sería la salida del mencionado sistema y si se parece o no se parece. (Véase *Figura 5.15*)

En el caso que nos incumbe podríamos decir que esta función aproxima de forma medianamente exacta a la señal de salida del sistema digital. Las diferencias las podemos atribuir a los efectos no lineales del motor debido a su inercia.

5.4 Lazo Cerrado

Una vez visto el funcionamiento del dedo y su control de posición en lazo abierto, se mostrará los resultados experimentales obtenidos gracias al uso de un control de lazo cerrado.

Para iniciar se tratará de cómo hemos conseguido implementar el sistema de control PID dentro de nuestro prototipo y daremos una descripción del sistema y de las partes más destacables del código que lo hace funcionar.

- Versión 1: Control PID con Arduino UNO y L293D
- Versión 2: Control PID con Arduino UNO y DRV8838
- Versión 3: Control PID con Arduino DUE y DRV8838

Tras esto se analizarán las respuestas generadas por nuestro sistema final mediante todos los tipos de control explicados en el apartado 4.3 de este documento, es decir:

- Control ON-OFF (Versión 1)
- Control Proporcional (Versión 3)
- Control Proporcional Integrativo (Versión 3)
- Control Proporcional Derivativo (Versión 3)
- Control Proporcional Integrativo Derivativo (Versión 3)

Con el objetivo de escoger el control que mejor se adapte a nuestro prototipo y así poder aplicarlo a los dedos para una eficiencia máxima.

5.4.1 Explicación del Software y Hardware del Control

El proceso de alcanzar el nivel suficiente de desarrollo, tanto de software como de hardware, para poder llegar al sistema que se buscaba desde un principio no ha sido inmediato. Se han necesitado tres versiones distintas para llegar a la implementación buscada.

Los siguientes apartados serán una descripción de cada versión y su evolución.

Versión 1: Control PID con Arduino UNO y L293D

Para la primera versión fue utilizado el controlador Arduino UNO. Este posee una velocidad suficiente como para controlar un solo dedo no obstante se hace pequeño cuando hablamos de controlar los 5 dedos y el servomotor. Es por ello que se tiende a hacerlo con Arduino DUE.

En primer lugar hablaremos del circuito que implemento esta versión. Se comenzó con un control de motor mediante el controlador L293D, también llamado puente en H. La decisión de utilizar este controlador y no utilizar el driver DRV8838 fue tomada en base a que, al ser una primera versión, se buscaba una comprensión completa del circuito. A su vez se pretendía poder observar las señales PWM que hacían funcionar el circuito para el experimento del siguiente apartado, apartado 5.4.2, cosa que sería más complicada utilizando el DRV8838.

Hardware:

El esquema del circuito implementado fue el que se puede observar en la *Figura 5.17*. Siendo el diagrama de bloques del L293D el mostrado en la *Figura 5.16*.

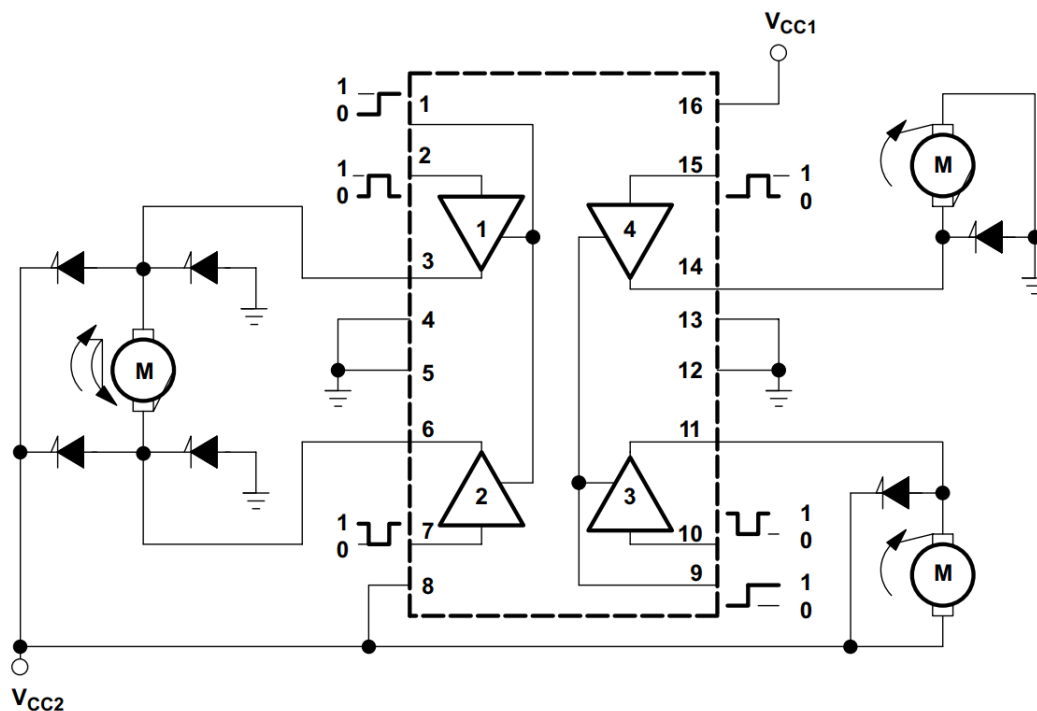


Figura 5.16 - Diagrama de Bloques de L293D [26]

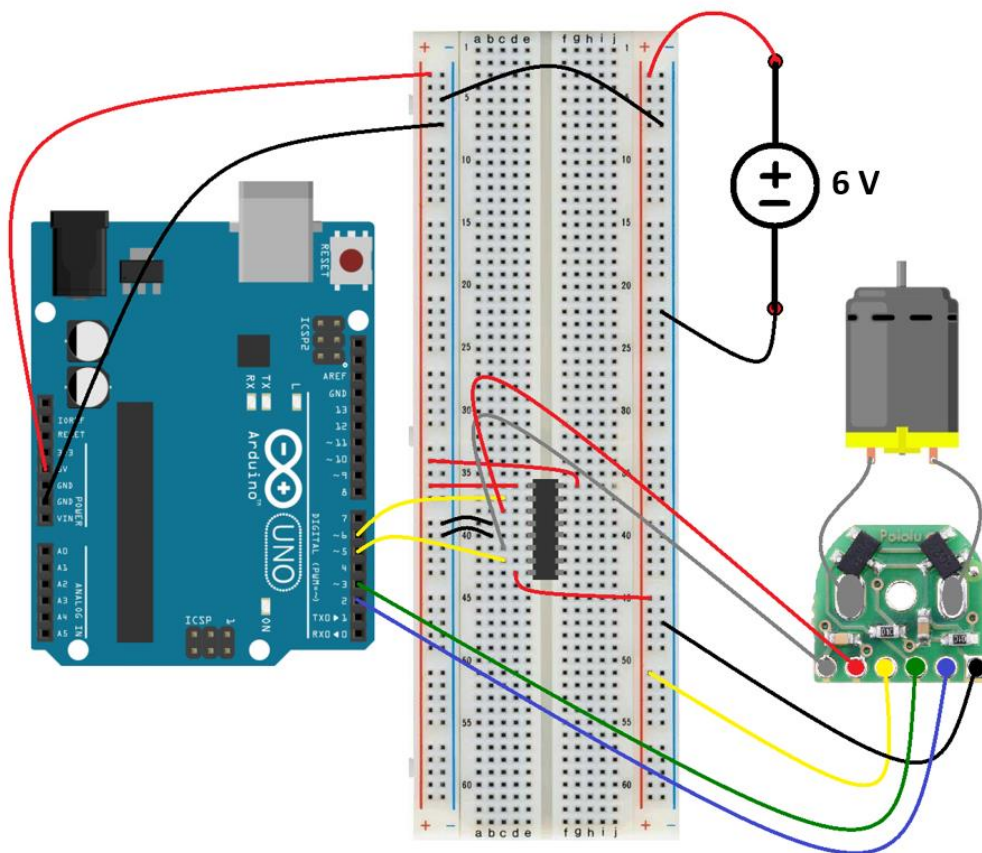
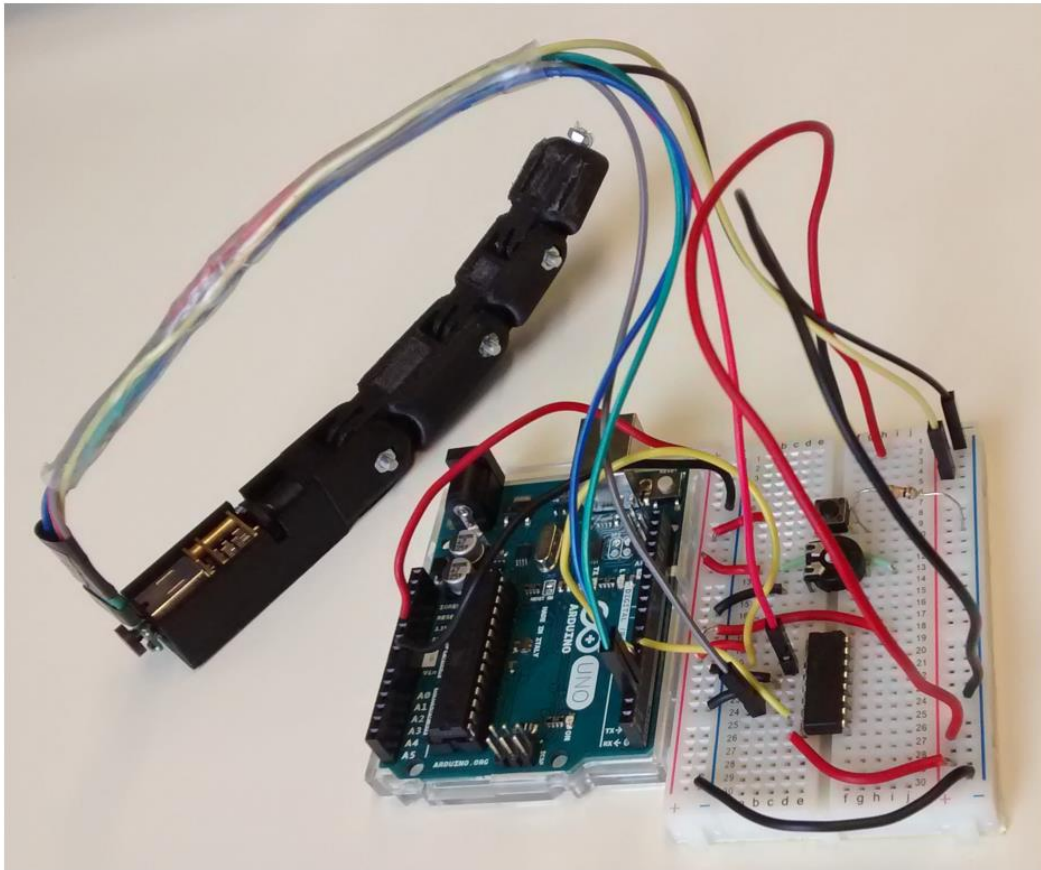


Figura 5.17 - Esquema del Circuito con L293D

El funcionamiento del Puente en H tal y como se puede observar en la *Figura 5.16* será el siguiente, se describirán solamente los aspectos de la imagen que guarden relación con nuestra implementación:

Se puede observar como en los pines de entrada del L293D número 1, 2 y 7 se tratan de las entradas de las señales PWM. No obstante el pin 1 será el encargado de encender o apagar los amplificadores. El pin 8 será el de la alimentación externa, gracias a una fuente de voltaje. Y los pines 3 y 6 serán las salidas para nuestro motor. Estas salidas serán la señal de entrada amplificada al valor del voltaje de alimentación externo para así poder manejar el motor. El pin 16 corresponderá con la alimentación a través del controlador, en nuestro caso 5V.

En cuanto al codificador (véase *Figura 5.17*), se ve en la *Figura 5.18* como las salidas 3 y 6 del L293D van a las entradas de los motores M1 y M2. Y como la alimentación externa y la tierra se conectan en la protoboard. Por último las salidas que proporciona el codificador se pondrán como entradas en el Arduino UNO.

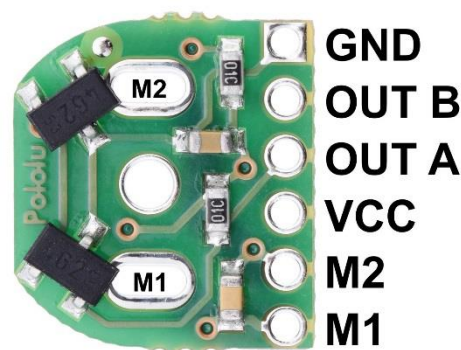


Figura 5.18 - Nombre de Pines del Codificador

Este encoder funciona gracias a los sensores de Hall que contiene, los cuales actúan al detectar variaciones magnéticas generadas por el disco que, enganchado al rotor trasero del motor, rota creando así estos fenómenos. En este caso este encoder emitirá 12 pulsos por vuelta que saldrán por las salidas A y B para su control mediante el algoritmo PID.

Software:

En este apartado se explicará el funcionamiento del software y las partes del código más importantes. Para los siguientes prototipos el funcionamiento global será el mismo aunque se le hayan realizado pequeñas variaciones al código, que serán explicadas. Todo el código de los experimentos realizados se introducirá a modo de Anexo en una carpeta que acompañará al documento.

El funcionamiento general del programa está dividido en tres bloques:

- En la primera parte se encuentran las declaraciones de variables y del PID. Este PID se ha podido implementar gracias a la librería para PID que posee Arduino creada por Brett Beauregard. Esta librería se encarga de hacer todo el cómputo del PID con las variables que le introduzcamos. Solamente es necesario introducirle el input para que te calcule el Output del sistema y de esta forma poder trabajar con el mismo.
- La segunda parte del código corresponde al Setup() del sistema Arduino. Se trata de un código que solamente se ejecuta al iniciar el programa. En este realizaremos el cambio de frecuencia para los pulsos PWM, declarar las interrupciones y la inicialización del PID.
- En el tercer bloque nos encontramos con el Loop() que será el código que ejecute continuamente Arduino. Este código básicamente no hará nada hasta que se cumpla el tiempo de muestreo configurado en el PID y una vez en acción escogerá si mantener o no el motor apagado y en el caso de no haber llegado a la posición, el sentido en el que el motor debe girar (Ver *Figura 5.19*). A su vez estará continuamente comprobando si hay datos del terminal para actuar en consecuencia.

```

Input = (double)contador;           // Lectura del encoder óptico. El valor del contador
                                     // se incrementa/decrementa a través de las
                                     // interrupciones extremas (pines 2 y 3).

while(!myPID.Compute());           // Mientras no se cumpla el tiempo de muestreo, se
                                     // queda en este bucle.

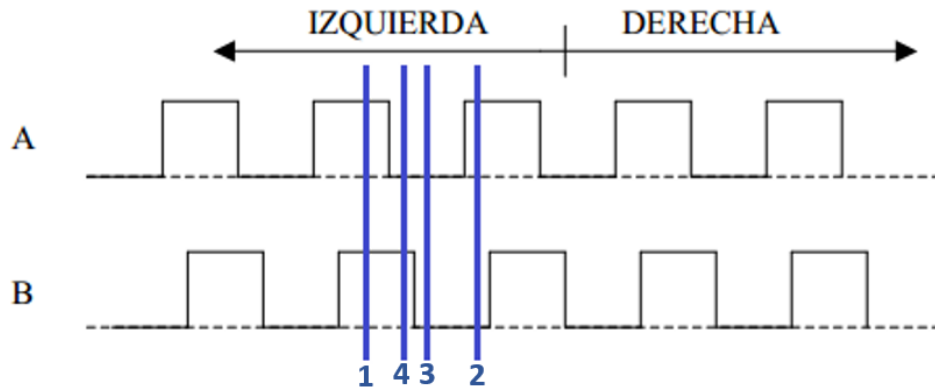
// ***** Control del Motor *****
if (((long)Setpoint - contador) == 0) // Cuando está en el punto designado, parar el motor.
{
  digitalWrite(PWMA, LOW);         // Pone a 0 los dos pines del puente en H.
  digitalWrite(PWMB, LOW);
  digitalWrite(ledok, HIGH);       // Se enciende el led (pin 13) para avisar visualmente
                                     // que está en la posición designada.
}
else
  // En caso contrario hemos de ver si el motor ha de ir
  // hacia delante o hacia atrás. Esto lo determina el
  // signo de la variable "Output".
  {
    if (Output > 0.0)               // Mueve el motor hacia delante con el PWM
                                     // correspondiente a su posición.
      {
        digitalWrite(PWMB, LOW);    // Pone a 0 el segundo pin del puente en H.
        analogWrite(PWMA, abs(Output)); // Por el primer pin sale la señal PWM.
      }
    else
      // Mueve el motor hacia atrás con el PWM
      // correspondiente a su posición.
      {
        digitalWrite(PWMA, LOW);    // Pone a 0 el primer pin del puente en H.
        analogWrite(PWMB, abs(Output)); // Por el segundo pin sale la señal PWM.
      }
  }

if (etime > 200000) //Si ha pasado 17 segundo (estampos en microsegundos)
{
  //Imprime cada segundo los datos, solamente hace eso
  Serial.println(contador);
  timep = time;
}
}

```

Figura 5.19 - Código Decisor de Acción de Motor [27]

- La función Encoder() será la que se lleve a cabo cada vez que exista una interrupción. Estas se leerán en los pines digitales del arduino 2 y 3 y cada vez que se detecte un cambio en los pulsos PWM que emite el encoder sumará o restará 1 al contador. Esto lo realiza mediante la palabra PIND de Arduino UNO, la cual te devuelve los 8 valores de los pines digitales y al realizar un AND con el valor 12 (00001100) nos quedaremos con los valores de los pines 2 y 3. Gracias a este enmascaramiento los valores de la palabra podrán ser los que dicta la *Figura 5.20*.



Pin 2 (A)	Pin 3 (B)	Estado	Valor PIND(Bin)	Valor PIND(Dec)
0	0	3	0000 0000	0
0	1	4	0000 0100	4
1	0	2	0000 1000	8
1	1	1	0000 1100	12

Figura 5.20 – Explicación Funcionamiento del Encoder

Por lo tanto en función del estado de estas entradas se tendrá un estado u otro y si lo comparamos con el estado pasado podemos deducir el sentido de giro del encoder y así saber si debemos de sumar o restar posiciones. De tal forma que código quedaría tal cual aparece en la *Figura 5.21*.

Se puede observar como si el estado anterior por ejemplo es 12 y el actual es 4 está girando a la derecha y por lo tanto deberemos sumarle 1 al contador. De esta forma se realiza el conteo de las vueltas.

```
// Encoder x4. Cuando se produzca cualquier cambio en el encoder esta parte
// hará que incremente o decremente el contador.
void encoder()
{
    ant=act;                // Guardamos el valor 'act' en 'ant'
                           // para convertirlo en pasado.

    act=PIND & 12;         // Guardamos en 'act' el valor
                           // que hay en ese instante en el
                           // encoder y hacemos un
                           // enmascaramiento para aislar los
                           // dos únicos bits que utilizamos
                           // para esta finalidad.

    if(ant==12 && act==4)  contador++; // Incrementa el contador si el
                           // encoder se mueve hacia delante.

    if(ant==4 && act==0)   contador++;
    if(ant==0 && act==8)   contador++;
    if(ant==8 && act==12)  contador++;

    if(ant==4 && act==12)  contador--; // Decrementa el contador si el
                           // encoder se mueve hacia atrás.

    if(ant==0 && act==4)   contador--;
    if(ant==8 && act==0)   contador--;
    if(ant==12 && act==8)  contador--;
}
}
```

Figura 5.21 - Código de la Función cuando se provoca una Interrupción [27]

Versión 2: Control PID con Arduino UNO y DRV8838

Una vez que ya teníamos la primera versión y habiendo entendido el circuito al completo y entendido el código, se decide pasar a simplificar el circuito gracias al driver DRV8838. (Véase su funcionamiento en el apartado 5.2) El cual aplica una sencillez al circuito observable en la *Figura 5.22*.

Los cambios en el software se limitaron a adaptar las entradas a las que el driver requiere, ya que en esta ocasión a la hora de adaptar el sentido de giro no dependerá de dos salidas como en el puente en H sino que solamente de una salida, así como la señal PWM solo se obtendrá del Pin 6 del Arduino.

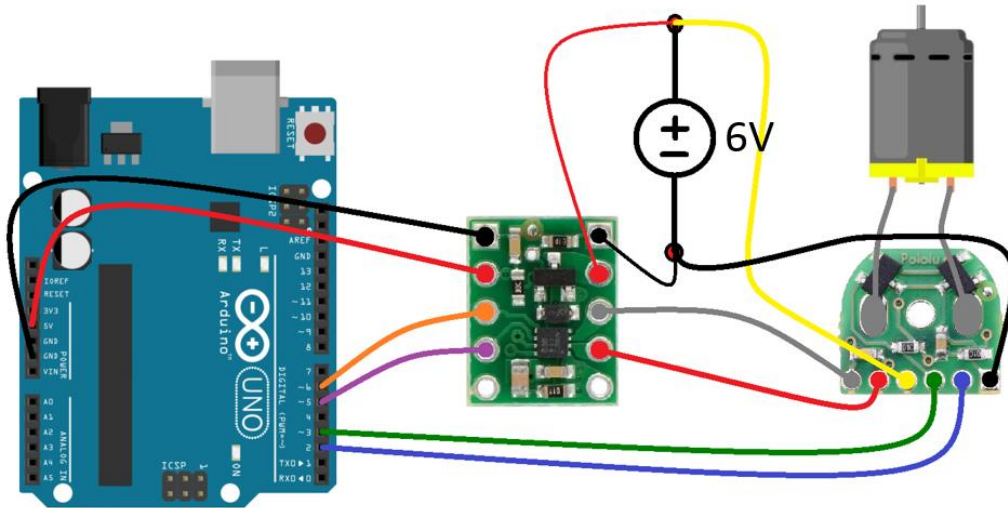


Figura 5.22 - Esquema Versión 2

Versión 3: Control PID con Arduino DUE y DRV8838

En esta última versión es la que utilizaremos para llevar a cabo los experimentos de control de posición.

En cuanto al hardware, esta versión será exactamente igual que el esquema de la versión dos solamente sustituyendo el Arduino UNO por el Arduino DUE. Se utilizarán las mismas entradas y salidas. (Véase *Figura 5.23*)

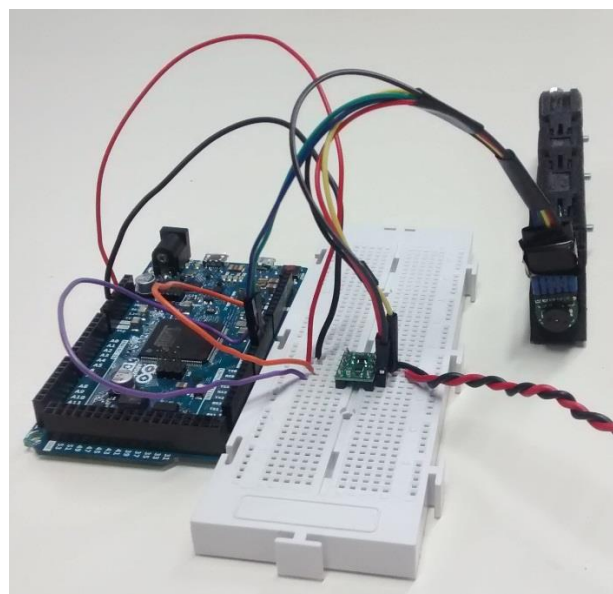


Figura 5.23 - Implementación Real Versión 3

No obstante el software ha cambiado en una pequeña medida. Pues el control de la frecuencia de las señales PWM en el Arduino DUE no es algo sencillo, más bien requiere una comprensión del microcontrolador más avanzada. Es por ello que para modificar la frecuencia por defecto de 1kHz de las salidas PWM se tuvo que modificar uno de los archivos del núcleo de la extensión del IDE de Arduino. En mi caso el archivo se encontraba en el siguiente archivo:

```
C:\Users\daniel.cerezo\AppData\Local\Arduino15\packages\arduino\hardware\sam\1.6.11\variants\arduino_due_x\variant.h
```

Los cambios realizados se pueden observar en la *Figura 5.24*:

```
/*
 * PWM
 */
#define PWM_INTERFACE      PWM
#define PWM_INTERFACE_ID  ID_PWM
#define PWM_FREQUENCY      20000 // changed from 1000
#define PWM_MAX_DUTY_CYCLE 255
#define PWM_MIN_DUTY_CYCLE 0
#define PWM_RESOLUTION     8

/*
 * TC
 */
#define TC_INTERFACE      TC0
#define TC_INTERFACE_ID  ID_TC0
#define TC_FREQUENCY      20000 // changed from 1000
#define TC_MAX_DUTY_CYCLE 255
#define TC_MIN_DUTY_CYCLE 0
#define TC_RESOLUTION     8
```

Figura 5.24 - Cambio Realizado en Fichero

Tras solventar este problema, nos encontramos que la función de Encoder() no podía realizarse debido a que la palabra PIND formaba parte solamente del microcontrolador de Arduino UNO y no del microcontrolador del Arduino DUE. Por lo tanto se sustituyó esta función por una con la misma funcionalidad pero de forma que el Arduino DUE la aceptase.

5.4.2 Control ON-OFF

El experimento que se relatará a continuación se trata de una comprobación del funcionamiento ON-OFF de una parte del control que se ha implementado en el apartado 4.2.1 gracias a la implementación en forma de puente en H de la Versión 1 del circuito de control. Este tipo de control es típico en sistemas digitales basados en microcontrolador.

Todos los datos se obtuvieron de forma práctica en el laboratorio tal y como se puede observar en la *Figura 5.25*. Para ello se aprendió el funcionamiento del osciloscopio de 4 canales emplazado en el laboratorio realizando el montaje.

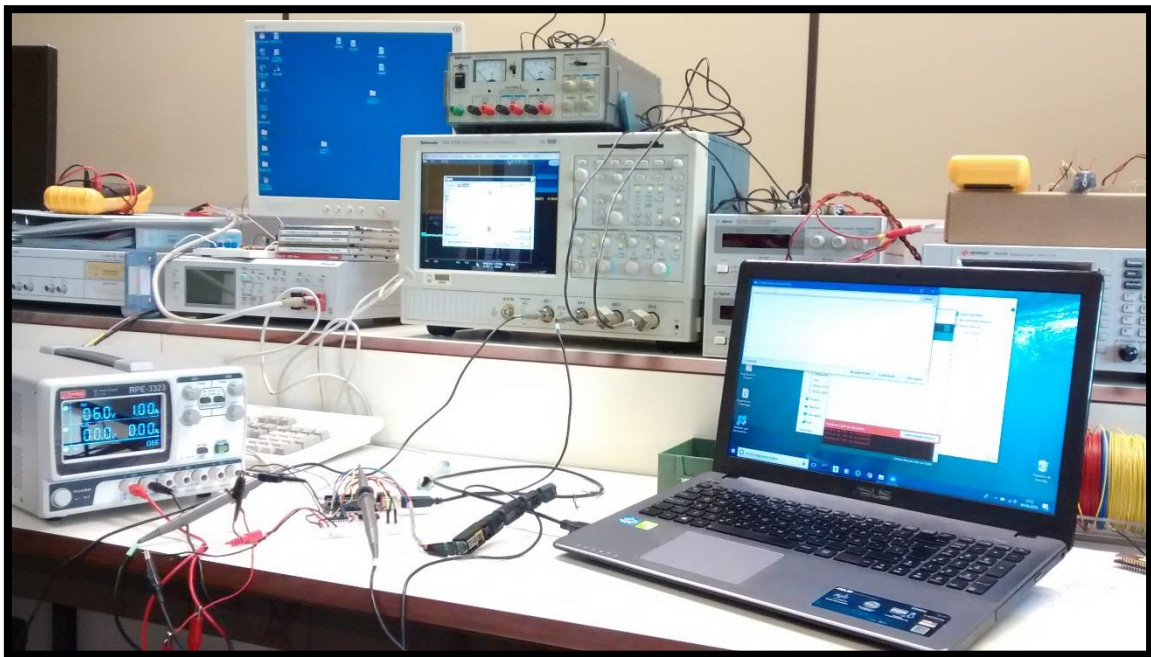


Figura 5.25 - Montaje Experimento ON-OFF

El control ON-OFF, también llamado SI/NO, es el control más sencillo dentro de los controles con realimentación. Como su propio nombre indica solamente tendrá dos estados, encendido o apagado. En nuestro caso los estados serán: motor en rotación y motor parado. En este caso no distinguiremos entre rotación en un sentido o en el contrario debido a que el control se ejerce en cada pin de forma separada, es decir, los

pinos de entrada al L293D son los pines 5 y 6, a los cuales se les aplica este control de forma independiente para llegar a un fin determinado.

Este control se observa en las entradas al puente en H, lugar en el cual hemos colocado las sondas del osciloscopio. Para así poder observar el funcionamiento ON-OFF de cada pin.

El movimiento realizado se puede observar en la siguiente gráfica, *Figura 5.26 arriba*, en la cual se ve como sube a la posición 9000 y tras un tiempo baja a la posición de partida. Con este movimiento se observa como el osciloscopio nos muestra la imagen que se puede ver en la *Figura 5.26 abajo*.

Siendo el amarillo el pin 5 y el azul el pin 6 se puede observar como en el momento en que nosotros enviamos la orden no existe ninguna clase de crecimiento exponencial hasta llegar al valor buscado, directamente alcanza los 5V y pasa a estar encendido. Al igual que al momento de apagarse, en el cual tampoco se observa un decrecimiento exponencial del valor.

No obstante si observamos mejor los datos, estos nos aportan más información sobre el funcionamiento de nuestro circuito.

En la *Figura 5.26*, se puede observar las similitudes de la señal digital extraída de la placa controladora Arduino con la señal analógica tomada de forma simultánea con el osciloscopio. El movimiento completo que se describe en la imagen lo dividiremos en tres partes llamadas A, B y C. En la primera de ellas, apartado A, la cual correspondería con el recuadro 1º) y 2º) de la *Figura 5.26*, el dedo se flexionará hasta la posición 9000, tras esto, apartado B, el dedo se mantendrá en esa posición un tiempo determinado, recuadro 3º), y por último, apartado C, descenderá a la posición 0, recuadro 4º) y 5º).

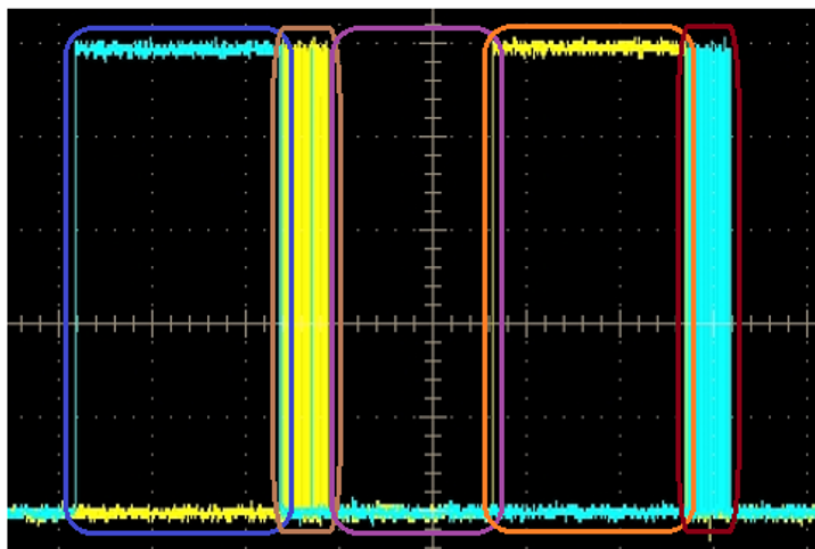
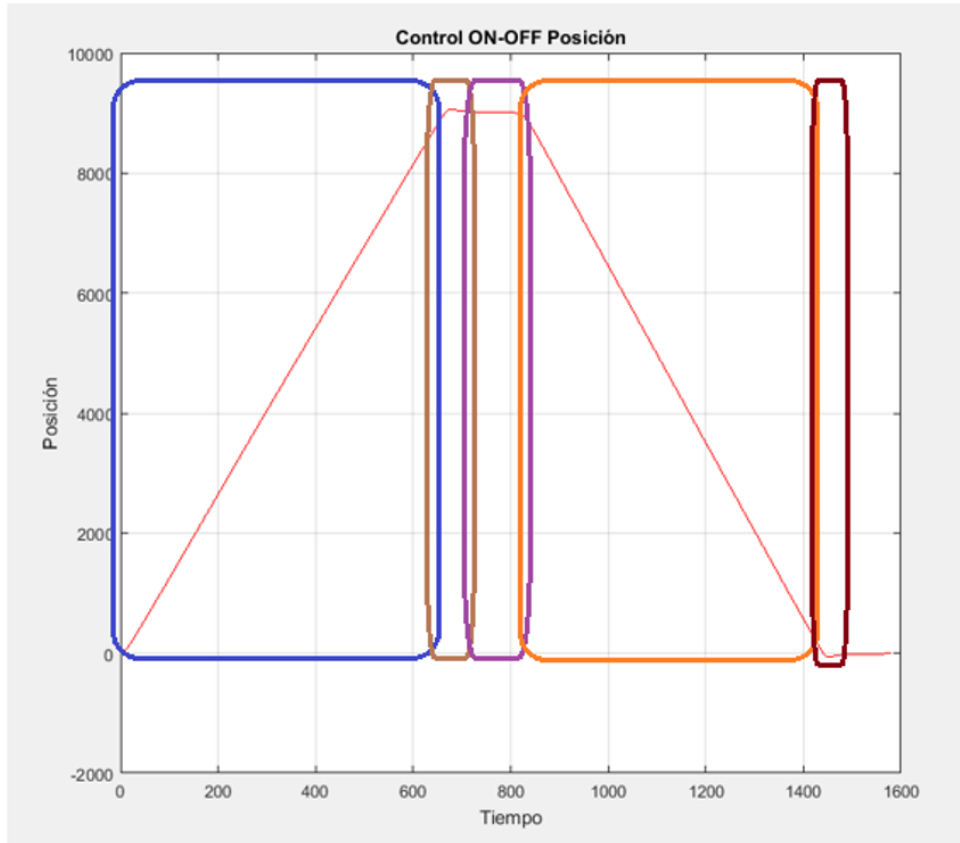
Como habréis podido observar los tramos A y C constan de dos partes diferenciadas. Para una mayor claridad se explicará cada tramo por separado.

Tramo A:

En este tramo se ve como la señal que marca el giro del motor es la coloreada en azul en la *Figura 5.26*. Se observa a su vez como la otra señal, correspondiente con el otro sentido de giro, se mantiene a 0 durante la flexión, tal y como debe de ser para un funcionamiento correcto del puente en H.

No obstante llega un momento en el cual la señal que hasta ahora marcaba el sentido de giro se hace cero y se observan picos continuos de la señal que marcaba el sentido contrario. Este fenómeno se debe al control PID que se ha implementado en la placa controladora.

Como se puede observar en la *Figura 5.27*, la cual es un zoom en la zona de parada de movimiento de la *Figura 5.26*, existe una clara sobre oscilación, recordar que se busca llegar a la posición 9000. Estas correcciones causadas por la acción del PID se traducirán físicamente a vueltas en sentido contrario para así poder alcanzar la posición exacta buscada, en la *Figura 5.26* se correspondería con el 2º) recuadro.



- 1º) Flexión
- 2º) Ajuste a Posición
- 3º) Reposo
- 4º) Extensión
- 5º) Ajuste a Posición

Figura 5.26 - Comparación Señal Digital con Analógica

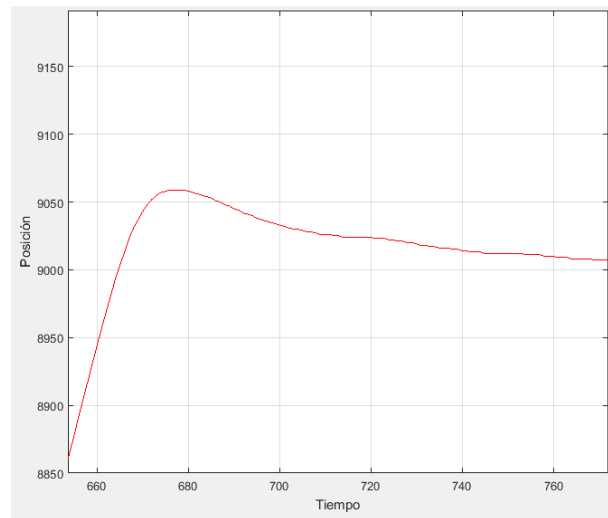


Figura 5.27- Sobreoscilación debido a ajuste PID en tramo A

Tramo C:

En el tramo C ocurrirá lo mismo pero de forma contraria, tal y como se ve reflejado por la contraposición de las señales descritas en el Tramo A. En la *Figura 5.28*, se puede observar la sobreoscilación que causará la parada del movimiento del tramo C, correspondiente a la extensión del dedo.

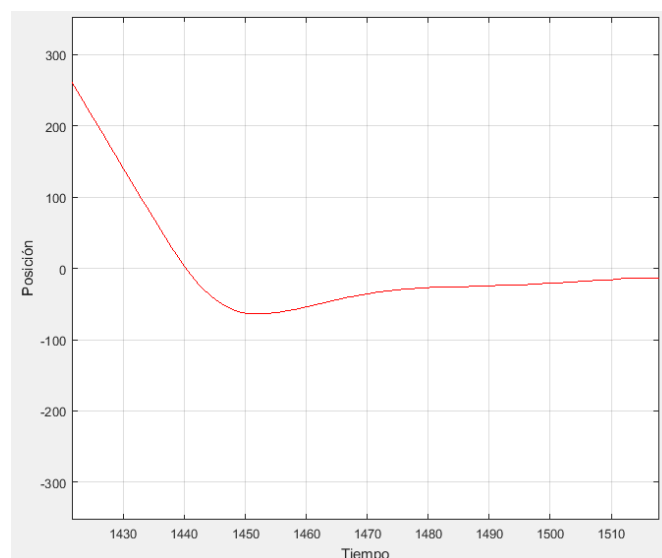


Figura 5.28 - Sobreoscilación debido a ajuste PID en tramo C

5.4.3 Ajustes PID (Control Proporcional)

Una vez visto el control ON-OFF ahora pasaremos a los ajustes de los parámetros del PID en lazo cerrado y como se afectan estos a la salida. En este experimento se recogerán los datos directamente del Arduino DUE y se graficarán, con el objetivo de ver cuál será la constante proporcional más indicada para nuestra planta.

El experimento que se pretende realizar será analizar la respuesta de nuestro sistemas para los valores de la constante proporcional de 0.1, 0.2, 0.5, 1, 10, 100 y 1000. Para hallar estos el microcontrolador nos devolverá por el puerto serial un valor del contador de posiciones en cada ciclo. Los resultados se pueden observar en la *Figura 5.29*.

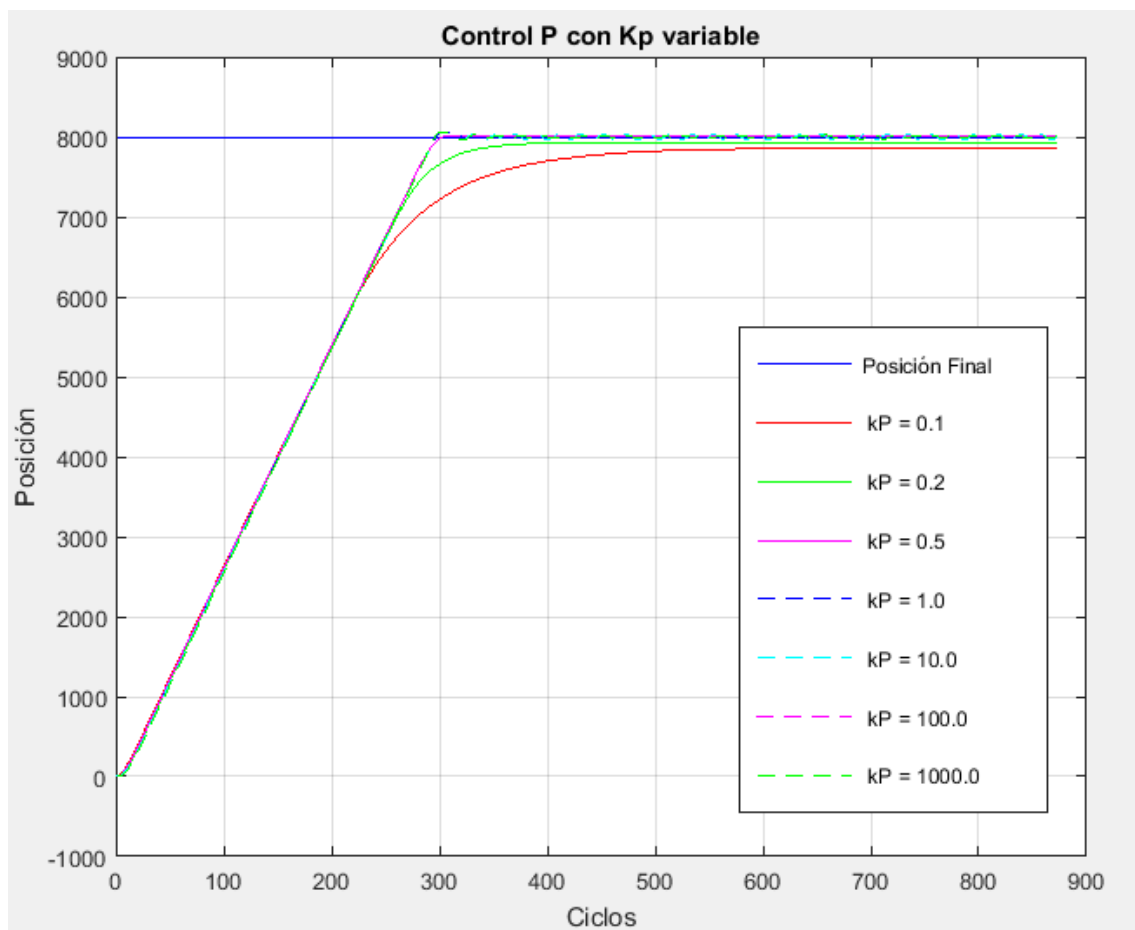


Figura 5.29 - Resultado Experimental Control de Posición

Antes de indagar más para obtener el valor más indicado, al ver la *Figura 5.29* se observa por comparación directa que esta no es como la teoría nos dice. La causa de que nuestro sistema no responda tal y como se expone en la teoría se debe a que es una implementación de un sistema real, es decir, un sistema con limitaciones. En caso que nos atañe las limitaciones son las del voltaje y corriente máximas que admiten los componentes y más en concreto, el motor en DC. Dicho motor posee un voltaje máximo de 6V y una corriente máxima de 1600mA, por lo tanto una vez que el motor está funcionando a máximo rendimiento, el crecimiento será lineal hasta la posición deseada. Justo eso es el fenómeno que estamos observando y que se le llama Slew-rate (Véase *Figura 5.30*).

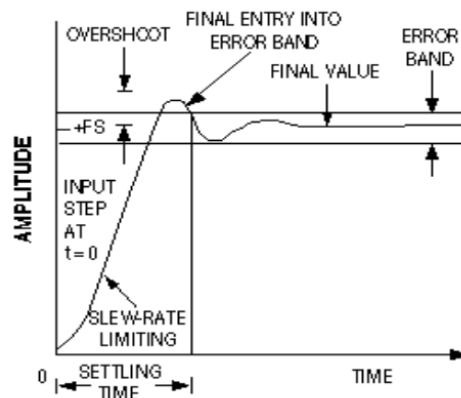


Figura 5.30 - Slew-Rate en Control PID [28]

El Slew-Rate es un concepto que se aplica en otros ámbitos de la electrónica. Se utiliza sobre todo en los amplificadores. Para estos el Slew-Rate se define como el máximo rango de cambio de la tensión de salida para todas las señales de entrada posibles. Esto nos limitará el funcionamiento en los amplificadores.

En nuestra implementación el que exista un Slew-Rate es completamente lógico por las limitaciones anteriormente comentadas y este mismo tampoco nos afecta en absoluto a nuestro sistema. Como curiosidad haciendo un sencillo cálculo podremos obtener el Slew-Rate de nuestro sistema, tal y como se puede observar en la *Figura 5.31*.

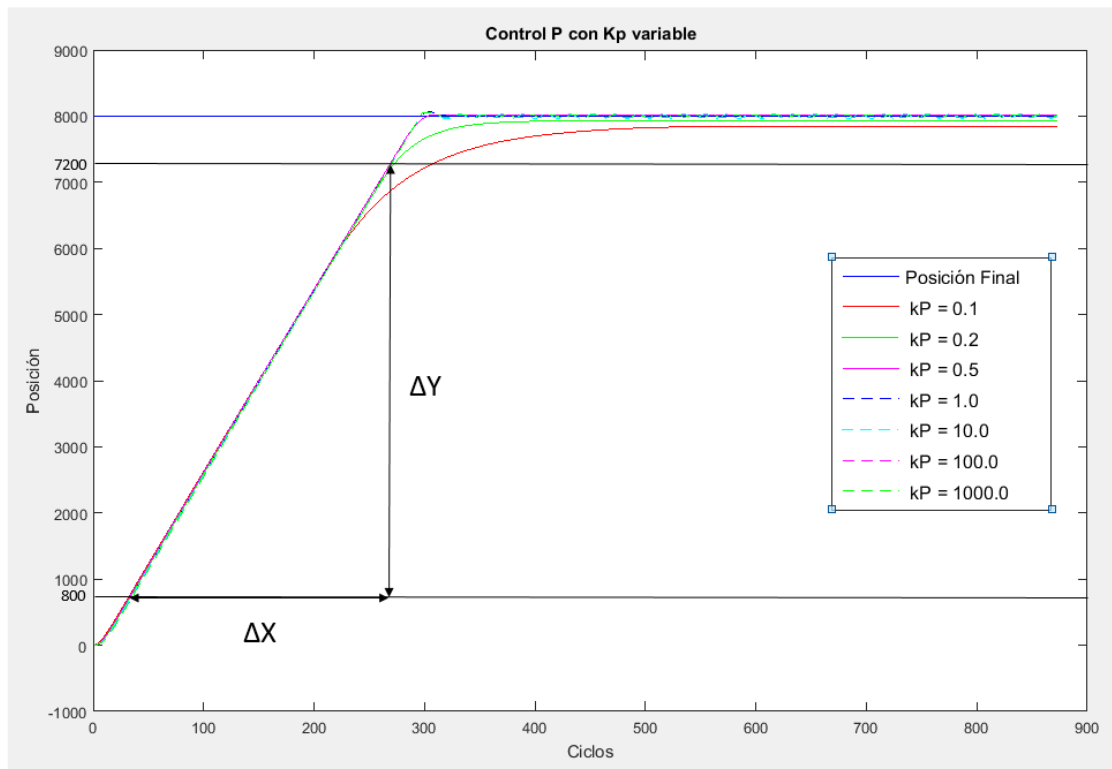
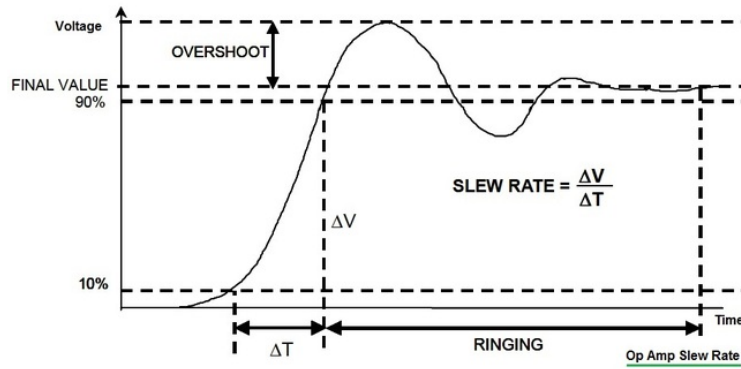


Figura 5.31 - Cálculo del Slew-Rate [29]

Obteniendo los valores x e y de las posiciones al 10% de la posición final, siendo esta 8000, y el 90% de la posición final se puede calcular el Slew-Rate de nuestro sistema, tal y como se ve en la ecuación 7.

$$SR = \frac{\Delta Y}{\Delta X} = \frac{7200 - 789}{266 - 35} = \frac{6411}{231} = 27.75$$

Ecuación 7 - Calculo Slew-Rate del Control Proporcional

Una vez explicado el efecto del Slew-Rate en nuestro sistema se llegan a las siguientes conclusiones:

- El Slew-Rate nos limitará la velocidad máxima de operación ya que su comportamiento se impondrá independientemente de los efectos del PID y el lazo cerrado.
- El efecto del Slew-Rate se observará de forma inevitable en todos los tipos de control que vamos a analizar.

Una vez explicado el fenómeno del Slew-Rate comentaremos los resultados experimentales obtenidos en la *Figura 5.29*. Para esto vamos a realizar un zoom a la figura para así obtener la *Figura 5.32*.

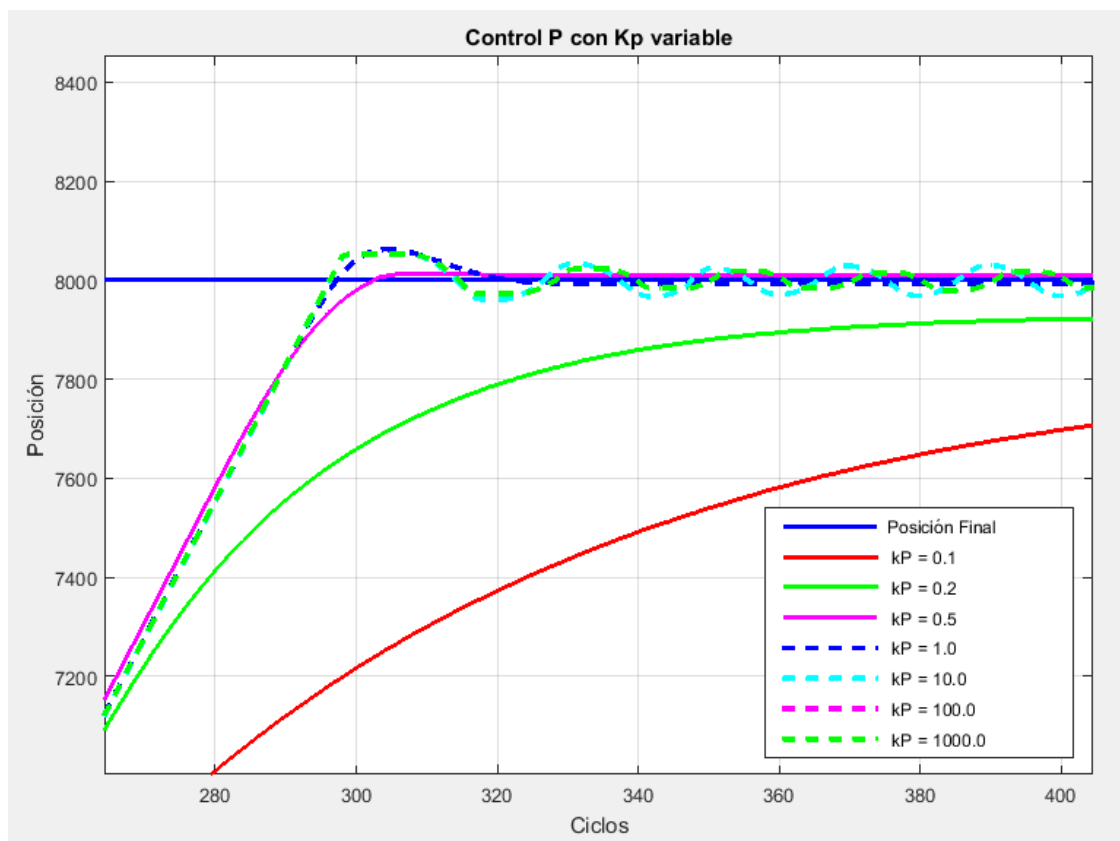


Figura 5.32 - Resultados Experimentales Control Proporcional Zoom

Tal y como se puede observar en la *Figura 5.32*, para un valor de K_p igual a 0.1 y 0.2, el ajuste es ineficiente. Para un valor de 0.5 observamos como se pasa pero no llega a oscilar al igual que con el valor de 1. Luego ya vemos como para valores mayores que 10 comienza a oscilar.

Siguiendo el método de ajuste manual, observaremos el momento en el cual el sistema empieza a oscilar y dividiremos entre 2 ese número para hallar nuestra constante proporcional. En nuestro caso se considera que empieza a oscilar en el valor de 1, por lo tanto el valor de nuestra variable proporcional de ahora en adelante para los siguientes experimentos será 0.5.

Conclusion:

- Constante Proporcional = 0.5

5.4.4 Control Proporcional Integrativo

Una vez con el valor de la constante proporcional fijado vamos a tratar de obtener el valor de la constante integrativa adecuado a nuestro sistema. Para ello observaremos la salida de nuestro sistema con el valor de la constante proporcional fijado y haciendo variaciones en el valor de la constante integrativa.

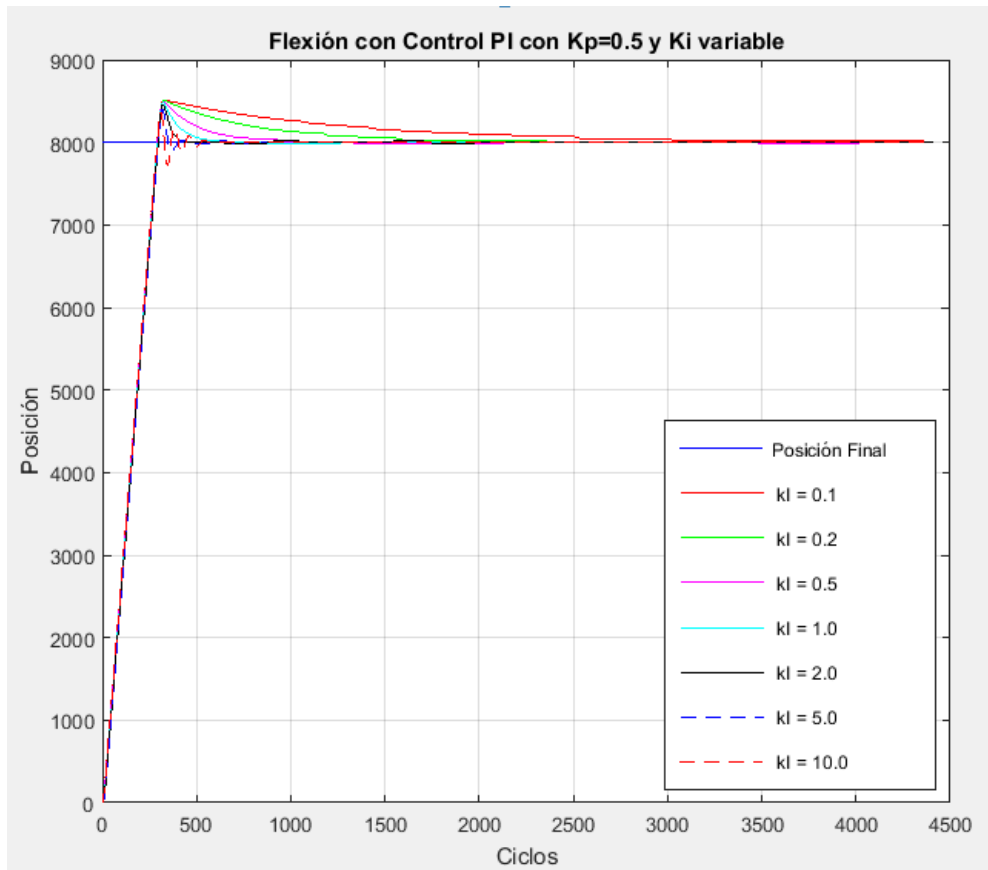


Figura 5.33 - Flexión con Control PI con $K_p=0.5$ y K_i variable

Observando la *Figura 5.33* podría considerarse que el Slew-Rate ha bajado, no obstante hay que fijarse en la escala de tiempos pues el Slew-Rate esta vez valdrá lo mostrado en la *Ecuación 8*.

$$SR = \frac{\Delta Y}{\Delta X} = \frac{7199 - 800}{267 - 38} = \frac{6399}{229} = 27.94$$

Ecuación 8 - Cálculo Slew-Rate del Control Proporcional Integrativo

Es decir, un valor muy cercano al anterior. Esto nos servirá como comprobación del correcto funcionamiento de nuestro circuito.

Si hacemos un zoom a la *Figura 5.33* en la zona de paso, se obtendrá la *Figura 5.34* la cual utilizaremos para decidir el mejor valor de la constante de integración.

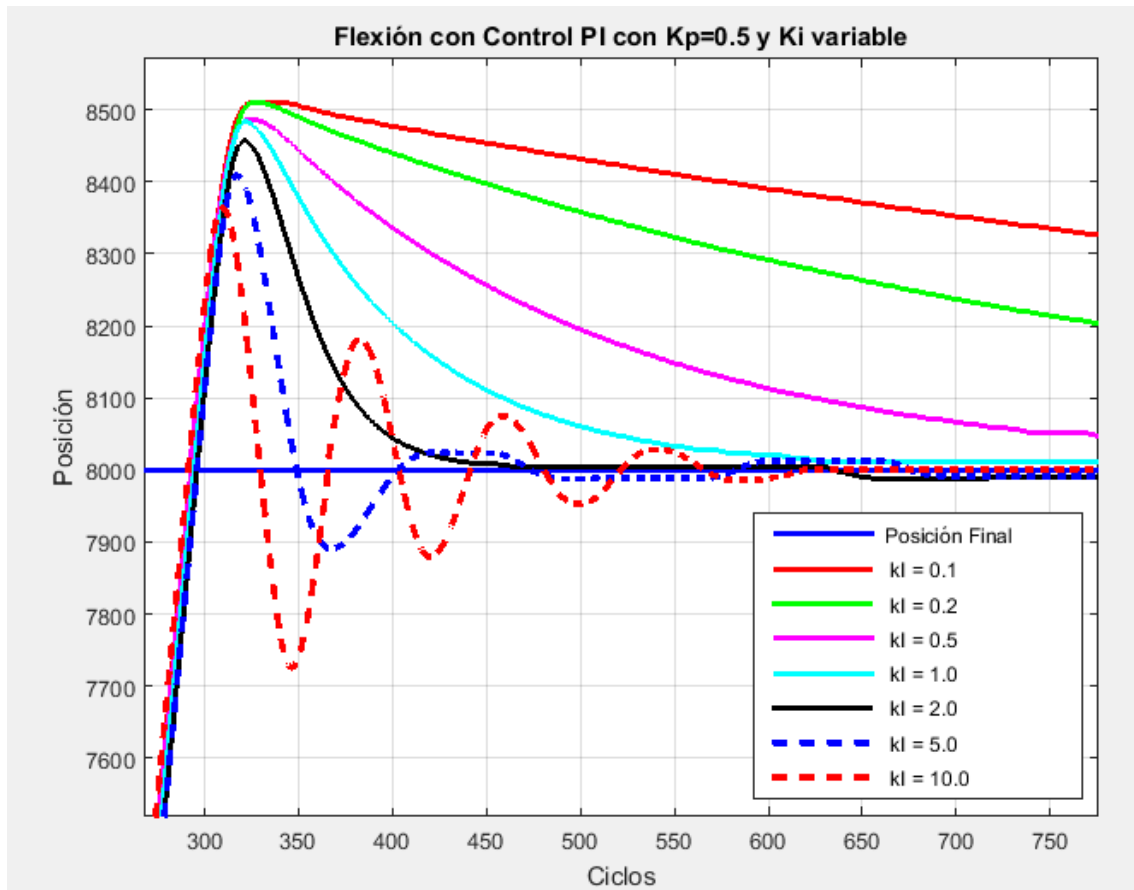


Figura 5.34 – Zoom de Flexión con Control PI con $K_p=0.5$ y K_i variable

En la *Figura 5.34* se puede observar como el hecho de que la constante de integración sea muy baja influirá en un lento descenso hasta alcanzar la posición deseada. A su vez se vé claramente como si la constante integrativa es muy alta provocará oscilaciones no deseadas en nuestro sistema. Así que observando las salidas obtenidas con estos valores se decide escoger el valor de 2 para nuestra constante de integración, pues pese a la sobreoscilación se estabiliza antes que cualquiera.

Conclusion:

- Constante Proporcional = 0.5
- Constante Integrativa = 2

5.4.5 Control Proporcional Derivativo

Se procederá de la misma forma que se ha procedido en el apartado anterior. Se obtendrán las salidas del sistema para varios valores de K_d , manteniendo K_p en 0.5, y tras esto se graficarán las distintas opciones para así luego poder escoger la mejor de las opciones.

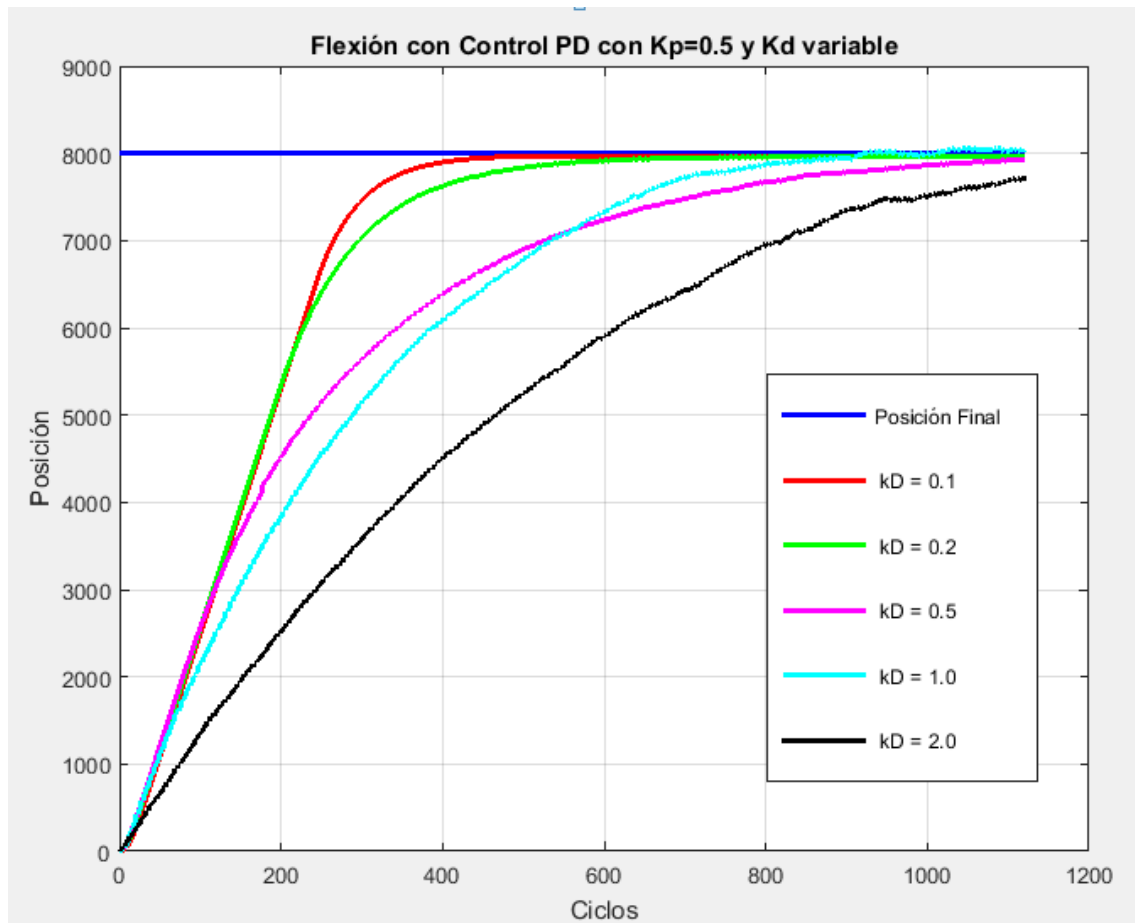


Figura 5.35 - Flexión con Control PD con $K_p=0.5$ y K_d variable

Si calculamos el Slew-Rate, *Ecuación 9*, se puede observar mantiene un valor prácticamente igual a los anteriores lo cual nos da certeza de su veracidad.

$$SR = \frac{\Delta Y}{\Delta X} = \frac{7204 - 807}{267 - 37} = \frac{6397}{230} = 27.81$$

Ecuación 9 - Cálculo Slew-Rate del Control Proporcional Derivativo

En este caso la *Figura 5.35* no requiere de hacer zoom debido a que está claro cual es el valor con el que nos quedaremos. Si observamos los valores 0.5, 1 y 2 se puede ver

como provocan más problemas que soluciones, sin contar con las oscilaciones continuas que provocan en el motor el cual puede llegar a dañarse (esto se sabe por experiencia propia). Entre el valor 0.1 y 0.2 descartaremos el 0.2 por que es menos inmediato que el valor de 0.1 Es por ello que nos decantaremos por el valor de 0.1 para nuestra constante derivativa.

Conclusion:

- Constante Proporcional = 0.5
- Constante Derivativa = 0.1

5.4.6 Control Proporcional Integrativo Derivativo (PID)

Por último una vez comprobados los distintos funcionamientos de los métodos de control de lazo cerrado terminaremos este experimento probando la validez de las tres constantes en un control PID.

Como se puede observar en la *Figura 5.36* se llega al punto deseado de forma adecuada y eficaz, se producirá una pequeña sobre oscilación que será compensada rápidamente.

A su vez si calculamos la Slow-Rate obtendremos un valor similar al anterior, tal y como se ve en la *Ecuación 10*:

$$SR = \frac{\Delta Y}{\Delta X} = \frac{7190 - 793}{265 - 37} = \frac{6397}{228} = 28.05$$

Ecuación 10 - Cálculo Slew-Rate del Control Proporcional Integrativo Derivativo

De esto podemos concluir que:

- Los valores $K_p = 0.5$, $K_i = 2$ y $K_d = 0.1$ son adecuados para nuestra implementación

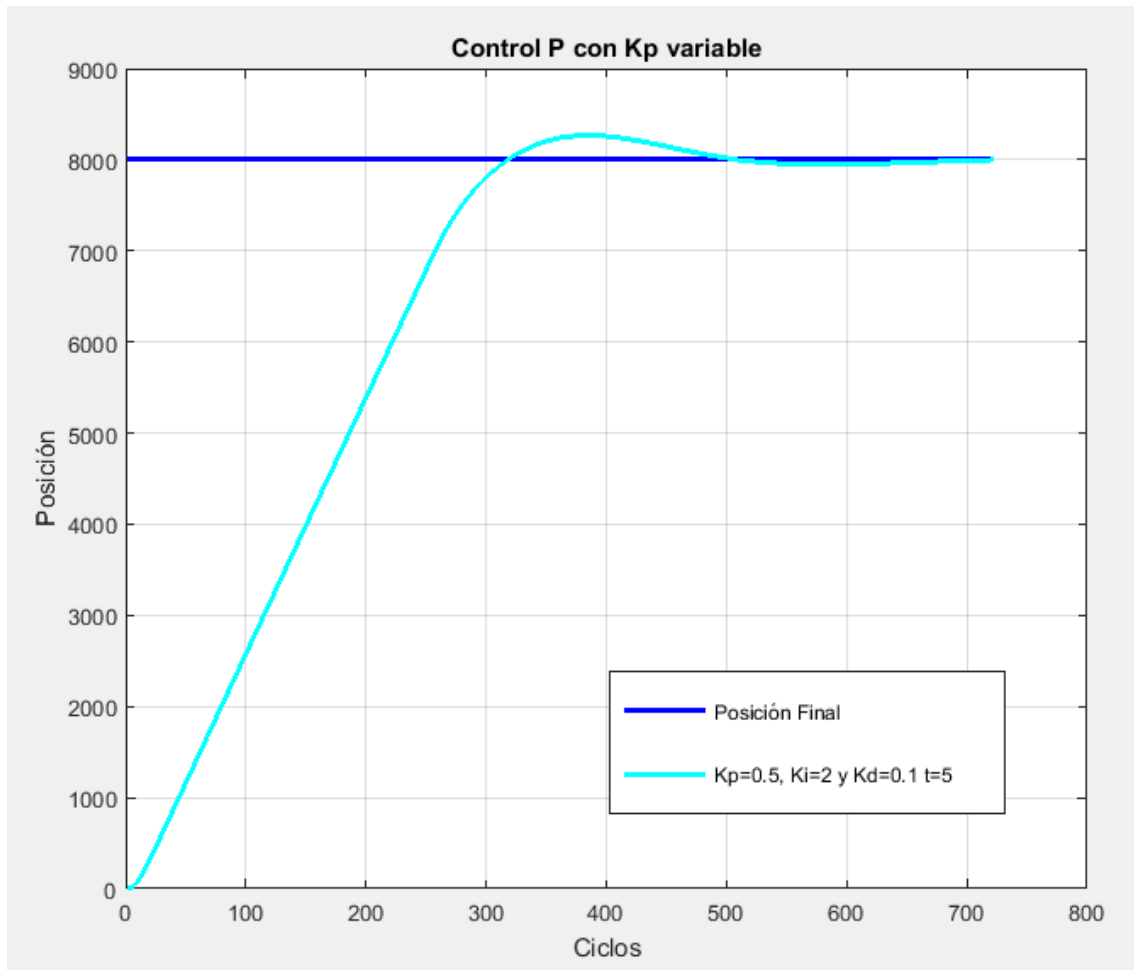


Figura 5.36 - Funcionamiento Final del PID

5.4.7 Estudio de Polos y Ceros según las variables PID

Como se ha ido viendo en anteriores apartados, las salidas que aportará nuestro sistema serán distintas en función a las constantes del PID que le introduzcamos. La siguiente sección pretende ser un estudio sobre los datos ya obtenidos en el transcurso de los experimentos en torno a los distintos controles del lazo cerrado.

Se dividirá en los tres partes diferenciadas estudiados anteriormente, control proporcional, control proporcional integrativo y proporcional derivativo. Se obtendrán los polos y los ceros de todos los valores distintos de las constantes y se representarán para su posterior comentario.

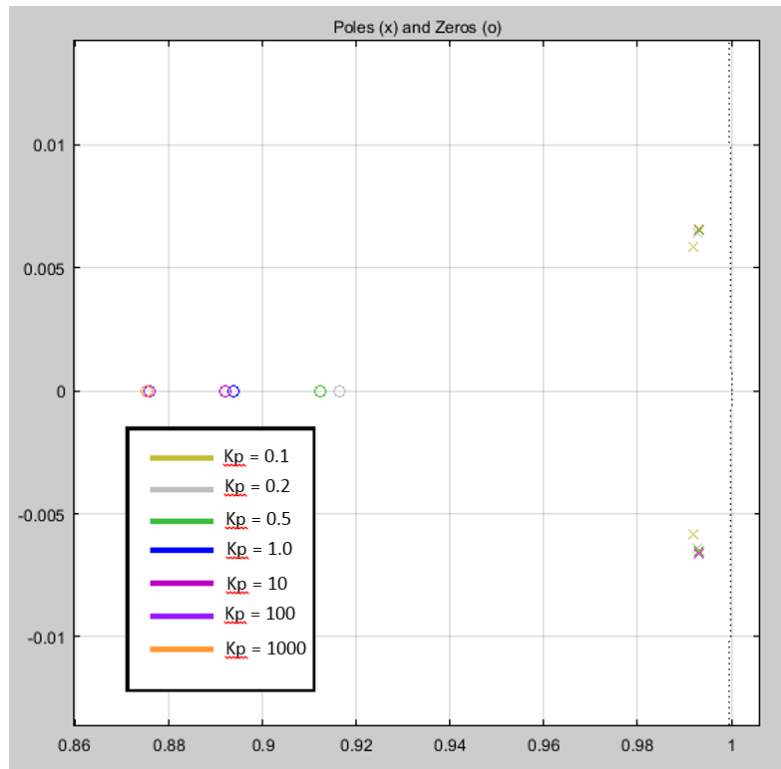


Figura 5.37 - Estudio Polos y Ceros Control Proporcional

Se puede observar en la *Figura 5.37* como según aumenta el valor de la constante, los ceros se alejan de los polos hacia el origen. Lo cual producirá una inestabilidad considerable.

En la *Figura 5.38* que representa el estudio de los polos y los ceros dejando fijada la variable proporcional, se puede observar que a medida que crece la constante integrativa, más se acerca también al origen. También se observa como los polos permanecen intactos, moviéndose solamente los ceros.

En el caso del control proporcional derivativo serán, tal y como se puede observar en *Figura 5.39*. Como se puede observar en dicha figura los polos se irán acercando según la variable aumente.

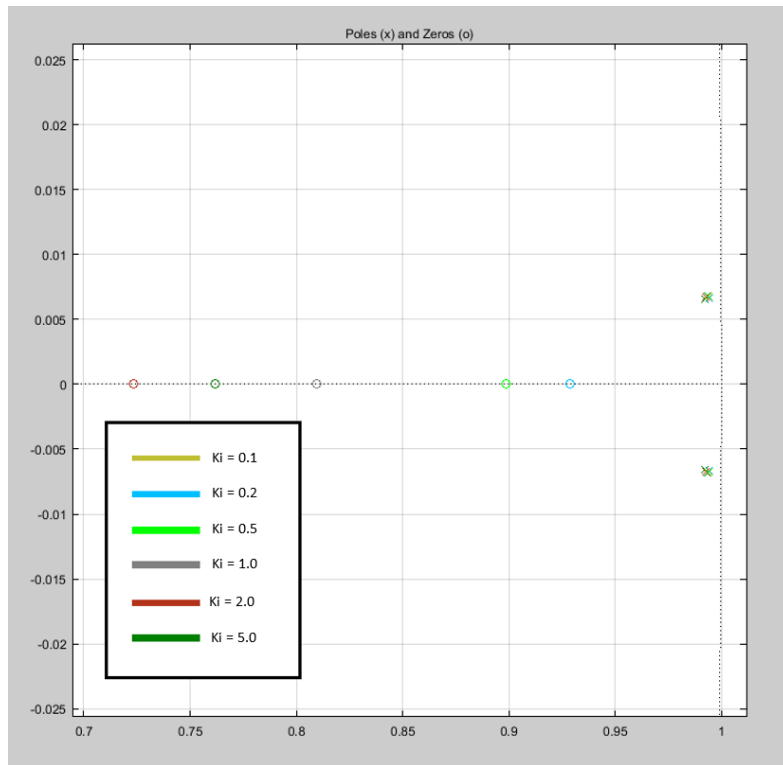


Figura 5.38 - Estudio Polos y Ceros Control Proporcional Integrativo

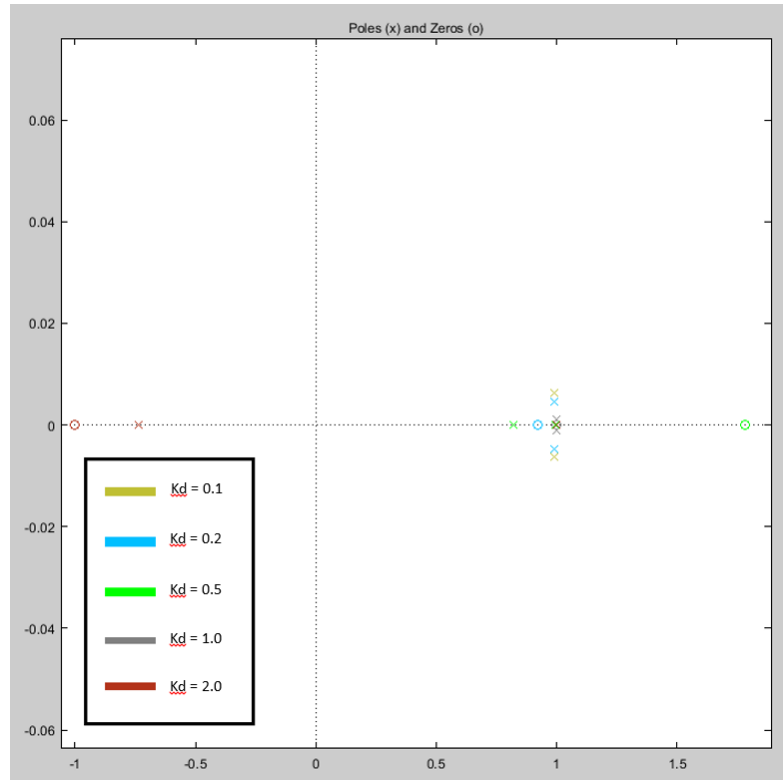


Figura 5.39 - Estudio Polos y Ceros Control Proporcional Derivativo

5.5 Control de Prótesis Completa

Tras la realización de todo el trabajo previo, se dio paso al montaje de la prótesis completa. Para ello se replicó para cada dedo el esquema de nuestra tercera versión explicada anteriormente. Se implementó un control PID con las mismas variables, de forma que todos los dedos se moviesen uniformemente. Para ello se tuvo que replicar el código una vez por cada dedo. A su vez se implementó el control del servomotor que se encargará del movimiento del dedo gordo de la mano.

El esquema del circuito realizado se puede observar en la *Figura 5.40*.

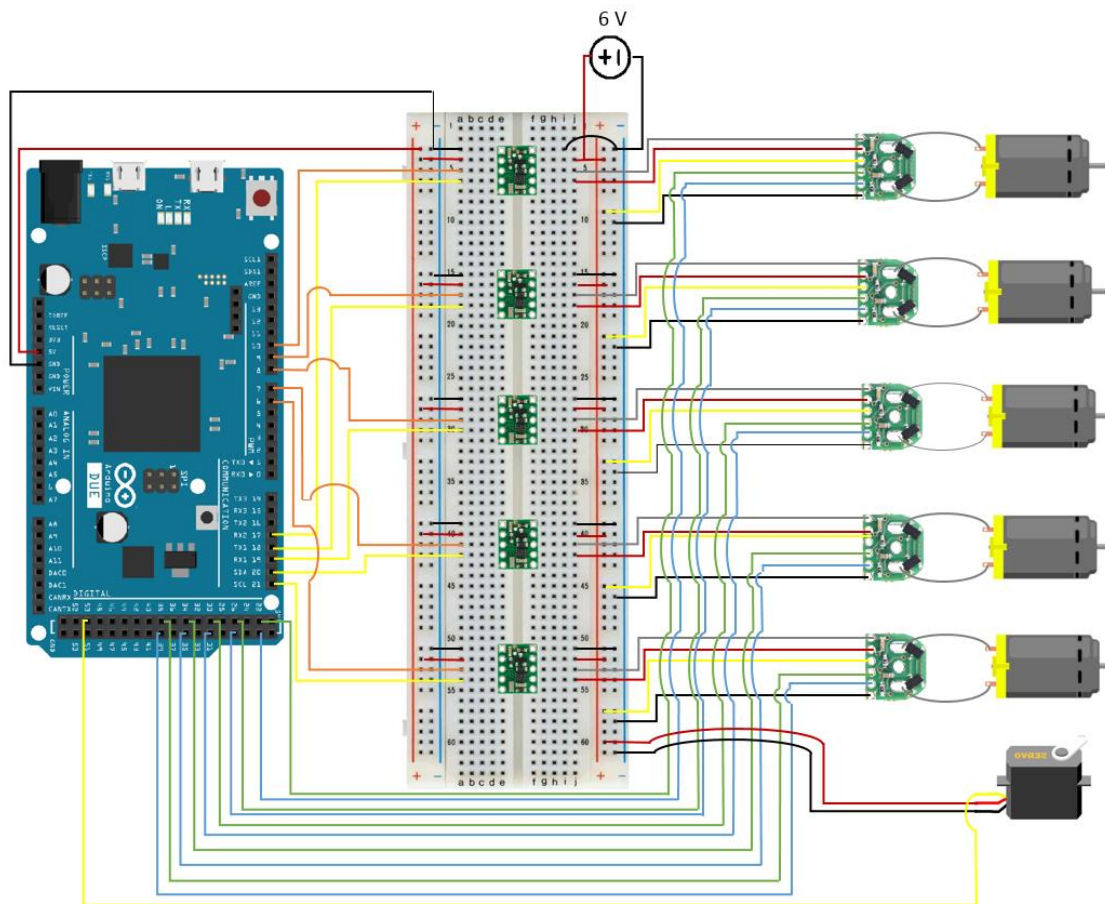


Figura 5.40 - Esquemático de Prótesis Completa

De esta forma se ha logrado controlar las posiciones de los dedos con una precisión absoluta. Gracias a las órdenes que le enviamos a través del puerto serial al Arduino DUE podremos enviarle las instrucciones necesarias para el movimiento. Se han

programado una serie de movimientos predeterminados pero también se le puede ordenar que un determinado dedo este en una determinada posición.

En la *Figura 5.41* se puede observar el resultado final de la prótesis y como es capaz de sostener herramientas gracias a su pulgar oponible.

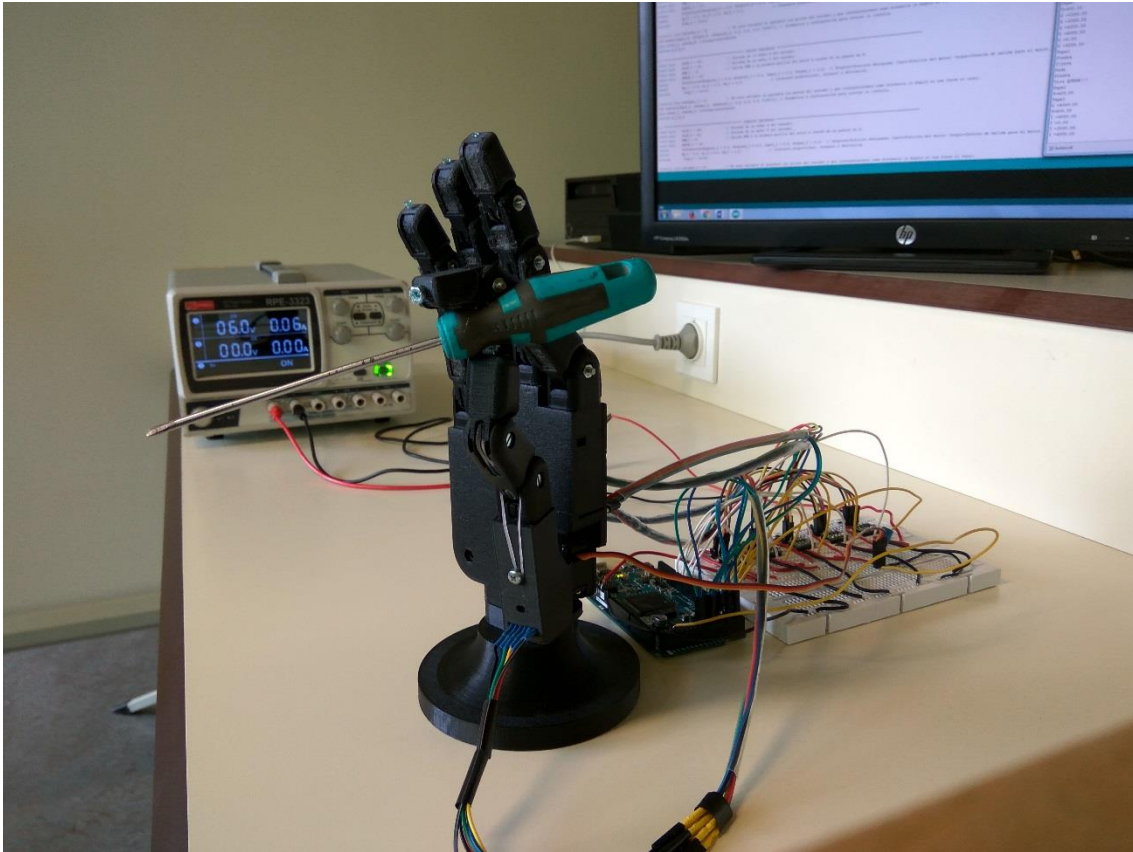


Figura 5.41 - Modelo Construido Sujetando un Objeto

6 Conclusiones

6.1 Metodología de Diseño

En el transcurso de este proyecto se han estudiado, comprendido he implementado los diferentes sistemas necesarios para controlar el movimiento de una mano robótica. Así como las diferentes formas de control de posición para los dedos.

Los sistemas de lazo abierto han resultado ser inexactos y carentes de precisión, no obstante al añadirle una realimentación y convertirlos en sistemas de lazo cerrado, estos han demostrado ser una forma precisa y eficaz de controlar la posición de los dedos.

El diseño del prototipo de este Trabajo de Fin de Grado se ha implementado distintas formas de control, así pues, se han visto las ventajas y desventajas que ofrecía cada uno de ellos al igual que los pros y contras de la solución escogida finalmente. A su vez, la intención de este trabajo ha sido la de implementar dicho control en una aplicación real, como bien es la mano robótica construida, para así poder observar las ventajas y desventajas que ofrecían cada uno de ellos.

El hecho de haber probado el circuito de control de la mano mediante dos etapas de potencia distintas ha ayudado a la comprensión del mismo así como a la simplificación de la etapa de potencia de la prótesis al mínimo exponente. A su vez también ha sido de ayuda el amplio estudio realizado sobre un solo dedo antes de montar la prótesis completa.

El uso de dos microcontroladores Arduino distintos en un principio produjo inconvenientes, al ser parte del código incompatible entre los microcontroladores. No obstante gracias a esto se ha llegado a un correcto funcionamiento de la prótesis y a una comprensión de ambos controladores.

6.2 Ventajas/Desventajas

Las ventajas que conlleva este modelo son múltiples, al haber utilizado la tecnología de impresión en 3D y el tener un diseño modular hace que el coste sea mínimo y la rapidez de construcción se maximice.

El sistema de transmisión se degrada con rapidez, siendo necesario cambiar prácticamente de forma diaria los tensores y de forma menos frecuente también el hilo que actúa a modo de tendón. No obstante a pesar de que el mantenimiento sea prácticamente diario, no supone ningún coste extra a tener en cuenta, dada su sencillez de diseño.

En cuanto al sistema de control se ha podido exprimir al máximo los datos del encoder debido a la rapidez que posee el controlador Arduino DUE, haciendo que el funcionamiento de la mano resulte firme y constante.

En cuanto a la calibración de la mano es necesario ir con cuidado, pues si se deja en una posición cerrada, al volverse a iniciar el sistema esa posición será nuestra posición 0, pudiendo llegar al deterioro de las piezas de la mano si esta se fuerza, incluso un motor, dado el torque del que disponen.

La ventaja de los controladores PID implementados es el grado de control que aportan de forma exacta, no obstante una mala calibración de las variables que rigen este control puede llevar al descontrol de nuestra mano robótica y por lo tanto a una incapacidad en el uso de la misma, es debido a esto por lo que se ha de tener especial cuidado en la implementación de estos sistemas para que siempre sean estables.

A su vez se ha observado que el sistema de control proporcional derivativo no es implementable en este modelo dado que crea inestabilidad al mover los polos del sistema hacia el lado positivo del plano complejo, es por ello que el grado de ayuda de la componente derivativa en nuestro movimiento se ha minimizado.

En general el prototipo presentado presenta claras ventajas, se trata de un sistema sencillo de montar, un lenguaje de programación abierto y con una amplia comunidad activa como es Arduino. Si bien no resulta tan robusta como una solución propietaria, sí

que se trata de una alternativa factible de bajo coste para el día a día, si se lleva el correcto mantenimiento por parte del usuario.

6.3 Puntos de Mejora

Un claro punto de mejora sería la programación que dispone el microcontrolador, pues actualmente no somos capaces de realizar movimientos secuenciales. A su vez también se podría implementar un control manual sin la necesidad de un ordenador para programar los movimientos.

Un claro punto a mejorar será el colocarle un sistema de control mioeléctrico a nuestra prótesis para que así la experiencia de usuario pueda ser completa. Así necesitaríamos incorporar sensores, adaptación de la señal y probablemente otro lazo de control.

Al ser un sistema protésico el circuito de control de la etapa de potencia debería no ocupar un gran volumen e ir alimentado con una batería, de esta forma se reduciría el tamaño considerablemente.

A su vez otra alternativa que debería estudiarse sería la de sustituir el hilo que a modo de tendón transmite el movimiento. Actualmente se usa una fibra de plástico utilizada en el mundo de la pesca, no obstante existen otros materiales similares más robustos que se podrían emplear para aportar así un mayor grado de solidez a la estructura interna de la misma.

Se podría utilizar otro tipo de sensores para captar la señal de salida y hacer el control. Por ejemplo algún material que cambie su impedancia en función del cambio de longitud, que se pudiera colocar a lo largo de los dedos, así dando una salida analógica que pudiera dar resolución y precisión.

Otra mejora sería utilizar algún tipo de control analógico con los motores para evitar el problema de slew-rate y hacer la respuesta más rápida y aumentar la velocidad de respuesta de la mano.

Bibliografía

- [1] R. Michalec, Modeling and control of multifingered dextrous manipulation for humanoid, Université Pierre et Marie Curie - Paris VI, 2011. English..
- [2] N. 2007 y H. S. a. H. S. 2. Monkman, Robot Grippers, 2007.
- [3] R. Larrubia, «mano articulada arduino,» 19 09 2014. [En línea]. Available: <https://www.youtube.com/watch?v=P9nfWO4lWd8>.
- [4] K. V., «3D Printed Controllable Prosthetic Hand via EMG,» 1 6 2015. [En línea]. Available: https://www.youtube.com/watch?v=KSP4o_WCqVs.
- [5] e. Inc., «World Design Guide,» 2015. [En línea]. Available: <https://ifworlddesignguide.com/entry/148430-handiii>.
- [6] Shadow Robot Company, «Shadow Dextrous Hand C6M Technical Specification,» [En línea]. Available: http://mindtrans.narod.ru/pdfs/shadow_dextrous_hand_technical_specification_C6M.pdf.
- [7] A. Villoslada, «Hackaday.io,» [En línea]. Available: <https://hackaday.io/project/9890-dextra>.
- [8] L. S. d. Lirio, «¿Cuántos huesos tiene la mano?,» LaReserva.com, 06 06 2017. [En línea]. Available: http://www.lareserva.com/home/cuantos_huesos_tiene_la_mano.
- [9] Pololu, «Pololu Robotics & Electronics,» Pololu, [En línea]. Available: <https://www.pololu.com/product/2990>.
- [10] Pololu, «Pololu Robotics & Electronics,» Pololu, [En línea]. Available: <https://www.pololu.com/product/3081/specs>.

- [11] Pololu, «Pololu Robotics & Electronics,» [En línea]. Available: <https://www.pololu.com/product/2990/resources>.
- [12] SpeedyTech, «Speedy-Tech Drive your best future,» [En línea]. Available: http://bdspeedytech.com/index.php?route=product/product&product_id=202.
- [13] Arduino, «Compare board specs,» [En línea]. Available: <https://www.arduino.cc/en/Products/Compare>.
- [14] [En línea]. Available: <https://arduino.stackexchange.com/questions/48208/arduino-due-not-executing-code-after-power-on>.
- [15] [En línea]. Available: <https://www.indiamart.com/proddetail/iso-course-on-design-of-embedded-system-using-arduino-uno-r3-9420066991.html>.
- [16] J. C. Villajulca, «Instrumentación y Control.net,» [En línea]. Available: <https://instrumentacionycontrol.net/sistema-de-control-de-lazo-abierto-open-loop/>.
- [17] valenpachi123, «InfoGram,» [En línea]. Available: <https://infogram.com/sistema-de-control-de-bucle-abierto-y-cerrado-1g957prkv8w8m01>.
- [18] Anónimo, «Unidad 2: Sistemas de Control,» [En línea]. Available: <http://docplayer.es/55192341-Unidad-2-sistemas-de-control.html>.
- [19] Salvador-UNSA, «Sistema control onoff,» 19 11 2011. [En línea]. Available: <https://es.slideshare.net/shuaranca/sistema-control-onoff>.
- [20] simulaciondeprocesosudenar, «Simulación de Procesos Udenar,» [En línea]. Available: <https://simulaciondeprocesosudenar.wordpress.com/control-proporcional-integral-derivativo-pid-y-estabilidad-de-sistemas-dinamicos/>.

-
- [21] EbRaInTe, «PV en funcion del tiempo (proporcional),» English Wikipedia, 2008. [En línea]. Available: <https://commons.wikimedia.org/wiki/File:Proporcional.PNG>.
- [22] J. C. Villajulca, «El control proporcional: definiciones prácticas y precisas,» 1 04 2010. [En línea]. Available: <http://instrumentacionycontrol.net/el-control-proporcional-definiciones-practicas-y-precisas/>.
- [23] EbRaInTe, «Pv en funcion del tiempo (integral),» English Wikipedia, 2008. [En línea].
- [24] Anónimo, «Control Proporcional Integral Derivativo (1ª PARTE),» 24 08 2012. [En línea]. Available: <http://todoproductividad.blogspot.com/2012/08/control-proporcional-integral-derivativo.html>.
- [25] controltheoryorg, «Acciones proporcional, integral y derivativa,» 16 04 2012. [En línea]. Available: <https://www.youtube.com/watch?v=KUDZ-Uv5IN8>.
- [26] Texas Instrument, «DatasheetL293D,» [En línea]. Available: <http://www.ti.com/lit/ds/symlink/l293.pdf>.
- [27] Anónimo, «Robotica,» [En línea]. Available: <https://sites.google.com/site/proyectosroboticos/control-de-motores/control-pid-con-libreria>.
- [28] Rocatek, «Rocatek,» [En línea]. Available: http://www.rocatek.com/forum_lazos_control.php.
- [29] RF Wireless World, «Home of RF and Wireless Vendors and Resources,» [En línea]. Available: <http://www.rfwireless-world.com/Terminology/op-amp-equations.html>.
- [30] D. M. Junge, «Motor de vapor Boulton y Watt, 1788,» 17 08 2008. [En línea]. Available:

https://es.wikipedia.org/wiki/Regulador_centralizado#/media/File:Boulton_and_Watt_centrifugal_governor-MJ.jpg.

[31] Siemens, «ProgramaciónSiemens.com,» [En línea]. Available: <https://programacionsiemens.com/pid-en-step7/>.

[32] Radio-Electronics, «Radio-Electronics,» [En línea]. Available: http://www.radio-electronics.com/info/circuits/opamp_basics/operational-amplifier-slew-rate.php.