



Universidad  
Carlos III de Madrid

Escuela Politécnica Superior

Departamento de Sistemas y Automática

GRADO EN INGENIERÍA EN TECNOLOGÍAS  
INDUSTRIALES

TRABAJO DE FIN DE GRADO

# HAPKIT: INTERFASE HÁPTICA CON CONTROL DE FUERZA

---

Autor: Enrique López Serrano

Tutor: Luis Enrique Moreno Lorente

Director: Antonio Flores Caballero

Leganés, Septiembre de 2015



# Agradecimientos

---

A mis padres y hermanos.

# Resumen

---

En este proyecto se pretende desarrollar un dispositivo háptico partiendo de la nada empleando para ello materiales que se pueden encontrar en casa, combinados con la electrónica necesaria. El sistema desarrollado simulará una interfaz háptica real.

Para ello se resolverá el problema realizando un estudio previo de las características de un sistema háptico. A continuación se procederá con el diseño y construcción del sistema físico para después controlarlo. Tras esto se realizarán diferentes pruebas que nos permitirán ajustar todos los parámetros.

# Abstract

---

This project aims to develop a haptic device. For this target we will use materials that can be found in house, combined with the necessary electronics. This system will simulate a real haptic application.

We will solve the problem doing a research about haptic systems characteristics. Then, we will design the physical system and build it. After that we are going to start to control this physical system. Finally we will make different proofs. This will help us to adjust all parameter.

# Índice General

---

1	Introducción y objetivos.....	1
1.1	Introducción .....	1
1.2	Objetivos .....	2
1.3	Fases de Desarrollo .....	2
2	Estado del Arte .....	5
2.1	Introducción a la robótica .....	5
2.2	Háptica .....	6
2.2.1	Realidad Virtual .....	6
2.2.1.1	¿Qué es la realidad virtual?.....	6
2.2.1.2	Clasificación de la Realidad Virtual.....	8
2.2.1.3	Componentes de la Realidad Virtual.....	10
2.2.2	Tecnología Háptica .....	10
2.2.2.1	Aplicaciones.....	10
2.2.2.2	Ventajas y Desventajas.....	11
2.2.2.3	Clasificación de los dispositivos.....	11
2.2.3	Conclusiones.....	14
3	Sistema Mecánico .....	16
3.1	Estructura de la maqueta .....	16
3.2	Diseños en 3D.....	18
3.2.1	Diseños CAD .....	19
3.2.1.1	Manivela .....	19
3.2.1.2	Mango .....	20
3.2.1.3	Porta Imanes .....	21
3.2.1.4	Bridas PCB .....	22
3.2.1.5	Brida de Ajuste .....	24
3.2.2	Impresión 3D .....	25
3.2.2.1	Material .....	25
3.2.2.2	Parámetros de Impresión.....	27
3.2.2.3	Colocación y resultados.....	28
3.3	Montaje .....	29
3.3.1	Materiales .....	29

3.3.2	Herramientas.....	30
3.3.3	Pasos.....	31
3.3.3.1	Preparación previa al montaje .....	31
3.3.3.2	Instrucciones de montaje.....	32
3.4	Ajustes .....	36
4	Sistema Electrónico .....	40
4.1	L298N .....	40
4.2	STM32F4 Discovery .....	43
4.2.1	Serie STM32.....	43
4.2.2	Descripción .....	43
4.2.3	Arquitectura ARM .....	44
4.2.4	Características de la plataforma STM32 F4 <i>Discovery</i> .....	45
4.2.5	Descripción de componentes y pines.....	46
4.2.6	Características de alimentación .....	47
4.2.7	Modelos STM32F4.....	47
4.2.8	Comparación con las plataformas Arduino Uno y Raspberry Pi .....	48
4.3	Motores .....	49
4.4	AS5045B .....	51
4.4.1	Introducción .....	51
4.4.2	Asignación de Pines.....	52
4.4.3	IMÁN .....	53
4.5	Conexiones .....	54
5	Sistema de Control .....	59
5.1	MATLAB/SIMULINK .....	59
5.2	Controlador PID.....	59
5.2.1	Introducción .....	59
5.2.2	Funcionamiento .....	60
5.3	Programa .....	64
5.3.1	Estructura del programa .....	65
5.3.2	Captura y Tratamiento .....	66
5.3.3	Maestro-Esclavo.....	67
5.3.4	Esclavo-Maestro .....	68
5.4	Calibración y puesta a punto.....	68
6	Conclusión y trabajos futuros.....	72

7	Costes del Proyecto.....	74
8	Bibliografía .....	77



# Índice de Ilustraciones

---

Ilustración 1: Sistema de Ventana (WOW).....	8
Ilustración 2: Sistema de Realidad Aumentada .....	9
Ilustración 3: Cybergrasp.....	12
Ilustración 4: CyberTouch .....	13
Ilustración 5: Hapkit Stanford .....	16
Ilustración 6: Primer Modelo .....	17
Ilustración 7: Segundo Modelo .....	17
Ilustración 8: Modelo Final.....	18
Ilustración 9: Manivela.....	20
Ilustración 10: Mango .....	21
Ilustración 11: Brida Imán.....	21
Ilustración 12: imán de Neodimio .....	22
Ilustración 13: Sistema con prototipo brida PCB.....	22
Ilustración 9: Prototipo brida PCB.....	22
Ilustración 14: Sensores en PCB.....	23
Ilustración 15: Sistema de ajuste A.....	24
Ilustración 16: Sistema de ajuste B .....	24
Ilustración 17: Brida de ajuste.....	25
Ilustración 18: Estructura ABS [8].....	26
Ilustración 19: Estructura PLA [9].....	26
Ilustración 20: Interface cura [10].....	27
Ilustración 21: Primera impresión.....	28
Ilustración 22: segunda impresión .....	28
Ilustración 23: tercera y cuarta impresión .....	28
Ilustración 24 : Tablero.....	31
Ilustración 25: Detalle corte.....	32
Ilustración 26: Detalle eje.....	32
Ilustración 27: Detalle embutición .....	32
Ilustración 28: Detalle ensamble.....	33
Ilustración 29: Detalle ejes 2 .....	33
Ilustración 30: Mando montado .....	34
Ilustración 31: Mando montado 2.....	34
Ilustración 32: Montajebrida PBC .....	35
Ilustración 33: Colocación Brida PCB.....	35
Ilustración 34: Detalle Caucho Motor .....	36
Ilustración 35: Colocación Motor .....	36
Ilustración 36: Croquis Tablero .....	37
Ilustración 37: Motor con reductora.....	38
Ilustración 38: Maqueta Final 1.....	38
Ilustración 39: Maqueta Final 2.....	39

Ilustración 40: Puente H.....	40
Ilustración 41: L298N .....	41
Ilustración 42: Patillaje del L298N [13] .....	42
Ilustración 43: L298N .....	42
Ilustración 44: Procesador Cortex- M4 [14] .....	44
Ilustración 45: Placa STM32F4 Discovery.....	45
Ilustración 46: Descripción de componentes y pines.....	46
Ilustración 47: Cuadro comparativo modelos STM32 F4 .....	47
Ilustración 48: Motores.....	49
Ilustración 49: Regla mano izquierda.....	50
Ilustración 50: Ley de Lorentz [17] .....	50
Ilustración 51: Esquema Pines .....	52
Ilustración 52: Imán de Neodimio .....	53
Ilustración 53: Características magnéticas.....	53
Ilustración 54: Actuación magnética .....	54
Ilustración 55: Esquema Conexiones STM32F4 Discovery .....	57
Ilustración 56: Interface PCB.....	58
Ilustración 57: Esquema PID.....	60
Ilustración 58: Proporcional .....	63
Ilustración 59: Integral .....	63
Ilustración 60: Derivativo .....	63
Ilustración 61: Regulador PID.....	64
Ilustración 62: Entorno de Trabajo Simulink.....	65
Ilustración 63: Conversión a grados 1 .....	66
Ilustración 64: Controlador .....	67
Ilustración 65: Condición de paso .....	68
Ilustración 66: Selección Puerto COM.....	69
Ilustración 67: Calibración Sensores .....	70
Ilustración 68: Programa .....	71

# Índice Tablas

---

Tabla 1: Características PLA y ABS.....	26
Tabla 2: Parámetros de impresión .....	27
Tabla 3: Materiales.....	30
Tabla 4: Herramientas.....	31
Tabla 5: Casos Puente H.....	40
Tabla 6: Casos L298N.....	41
Tabla 7: Microcontroladores STM32.....	43
Tabla 8: Comparación entre STM32F4 Discovery, Arduino Uno y Raspberry Pi.....	49
Tabla 9: Conexiones STM32F4 <i>Discovery</i> .....	55
Tabla 10: Leyenda Conexiones STM32F4 <i>Discovery</i> .....	56
Tabla 11: Tiempo empleado en el proyecto .....	74
Tabla 12: Coste Materiales.....	75
Tabla 13: Costes de equipo y software .....	75

# Estructura de la memoria

---

El proyecto se encuentra dividido en un total de 7 capítulos, los cuales se describen a continuación.

- Capítulo 1: Descripción de todas los objetivos y fases del proyecto.
- Capítulo 2: Se habla de la historia relacionada con el mundo de la robótica y se introduce el concepto de realidad virtual y sistemas hápticos.
- Capítulo 3: Descripción de la parte física del sistema. Se explican por qué y cómo se han diseñado los elementos del sistema. Se explica la estructura de la maqueta y como se ha montado, indicando dónde se sitúa cada elemento.
- Capítulo 4: Descripción de la electrónica del sistema necesaria para el funcionamiento y el control de la maqueta. Explicación de cada uno de los elementos empleados, así como de las conexiones entre los diferentes elementos.
- Capítulo 5: Se describe el funcionamiento de un control PWM y se explica el programa desarrollado para el control de la maqueta. Se explica el funcionamiento de un controlador PID.
- Capítulo 6: Se comentan las conclusiones obtenidas y los posibles trabajos futuros a desarrollar.
- Capítulo 7: Desglose de forma aproximada de los costes del proyecto.

# **1 Introducción y objetivos**

## **1.1 Introducción**

A lo largo de la historia, la humanidad ha intentado facilitar su trabajo desarrollando diferentes máquinas, que de un modo u otro han ido facilitando dichos trabajos. De manera que estos artefactos han ido facilitando la vida de los hombres. Con el paso de los años estas máquinas han ido adquiriendo complejidad recibiendo todo tipo de nombres. Fue en el año 1921, de la mano del escritor checo Karel Capek, cuando para designar a estas máquinas se acuñó el término robot.

Este continuo desarrollo ha provocado que la humanidad siga investigando y ampliando la utilidad de dichas máquinas o desarrollando otros dispositivos con otras aplicaciones. Esto ha derivado en un compendio de nuevas tecnologías que poco a poco se han ido desarrollando en las diferentes escuelas de ingeniería del mundo.

Un ejemplo de este fenómeno es el de las aplicaciones que están directa o indirectamente relacionadas con la Háptica. Este es un término que designa la ciencia del tacto y todo lo referente a este sentido. Esta es una palabra que viene del Griego “*háptō*”, que hace referencia a todo lo referente al tacto. Este término ha extendido su significado haciendo alusión por exclusión a todo lo el conjunto de sensaciones no visuales y no auditivas.

Por lo tanto las aplicaciones hápticas son aquellas que tienen algo que ver con el tacto o con las sensaciones que no son ni ópticas ni auditivas. Se trata de una tecnología emergente desde los 90 que avanza prometedoramente en algunos campos de aplicación.

Uno de los primeros usos de interfaz háptica se empleó en los sistemas servo de los aviones. Estos sistemas actuaban en una única dirección, de tal modo que el piloto manipulaba el mando y, la fuerza que el piloto ejerciera se multiplicaba y se transmitía a los elementos terminales; sin obtener como respuesta un factor de fuerza resistente. En los inicios, los aviones no tenían estos servos, de manera que la fuerza que el piloto ejerciera era la que se transmitía a los actuadores. De esta forma el piloto tenía en todo momento información relativa a los elementos finales. Con los nuevos servos el piloto no notaba esta resistencia. Esto hacía que los pilotos perdieran el control en algunas situaciones. Para solventar este problema se instaló un sistema de control que proporcionaba una resistencia a la palanca del piloto. Este es un ejemplo de interfaz háptica.

Algunos de los campos de aplicación háptica son los teleoperadores y simuladores, los juegos y videojuegos donde hay que destacar los joysticks y controladores de juego con retorno háptico. O las tabletas vibratorias. Poco a poco estas aplicaciones van ganando cada vez más terreno en realidad virtual, añadiendo la sensación del tacto a los entornos visuales. Esta es el caso del sistema denominado “3D-Mobile Immersive Workstation”.

En medicina existen diversas interfaces hápticas que ayudan a minimizar los daños ocasionados por el uso de procedimientos invasivos, permitiendo operar de forma remota.

Además se han desarrollado diferente prótesis, por ejemplo manos, que ayudan en caso de lesión o incluso amputación.

Todo esto presenta la Háptica como un campo de estudio muy amplio, aun si explotar y con un gran futuro, que puede ayudar a desarrollar nuevas formas de interacción máquina-persona.

## 1.2 Objetivos

El objetivo que persigue el Proyecto Final de Grado, descrito en este documento, es el desarrollo de un sistema háptico con fuerzas de entrada y salida partiendo desde “cero”. Por lo tanto se deberá diseñar, construir y controlar un sistema físico real.

El sistema estará formado por dos palancas que nos permitirán simular un sistema háptico real. Así, el sistema reaccionará ante dos fuerzas de entrada, con una fuerza resultante de salida en una de las palancas. Esta fuerza será el resultado de la oposición al movimiento que presente la otra palanca.

Para esto, se diseña la maqueta partiendo de la nada, empleando materiales “caseros”. Tras el diseño del sistema físico, se precederá a la elección de los componentes electrónicos que nos ayudarán controlar la maqueta. La maqueta contará con un controlador STM32F4Discovery, del fabricante ST Microelectrónica, que será programado mediante Matlab y que gobernará el correcto funcionamiento. Por lo tanto, para poder desarrollar este proyecto se deberá conocer el entorno de programación y el funcionamiento de la placa de control.

Con las entradas producidas por el usuario, una de movimiento y otra de oposición, registradas por los sensores; y usando la técnica de control PID, el sistema podrá simular sensaciones hápticas similares a la que se experimentan en cualquier entorno real. Esto nos ayudará a comprender el funcionamiento de este tipo de tecnología y la complejidad de su programación.

Tras la construcción y programación de la maqueta se realizará un estudio del funcionamiento del sistema desarrollado que nos ayudará a valorar el trabajo realizado.

## 1.3 Fases de Desarrollo

Conociendo los objetivos, la primera fase que se desarrolló fue la de hacer un trabajo de estudio e investigación para poder abordar convenientemente el problema presentado. Este estudio se basó en la búsqueda de otros proyectos ya existentes relacionados con la háptica. Además se buscó en la web diferentes artículos relacionados con el tema y se tomó como referencia la página <http://hapkit.stanford.edu>.

Tras esto se procedió a diseñar la maqueta teniendo en cuenta las necesidades que con nuestro sistema se querían cubrir. Cuando se decidió que modelo emplear se comenzó con el diseño de las piezas. Para esto se realizaron diferentes prototipos en Solid Edge, programa que ya se había empleado durante el Grado. Una vez que finalizó con este desarrollo se procedió a la impresión de las mismas en una impresora Prusa I3.

El siguiente paso fue el de seleccionar la electrónica que se quería emplear. Para los sensores de posición se optó por usar un AS5045B de la empresa AMS, pues ya había conocimientos sobre su funcionamiento. Para el control de la fase de potencia de los motores se optó por un L298N, porque permitía controlar dos motores con una sola placa. Esto aparte de ser más barato, resulta muy eficiente. Para controlar el sistema se optó por una placa de control STM32F4. Esta es una placa similar a un Arduino Uno o a una Raspberry Pi, pero nos aporta diferentes ventajas que las otras placas no nos aportaban. Una de ellas por ejemplo es que la placa STM32F4 entiende directamente el lenguaje de Simulink, algo que nos facilitó la programación. [1]

Cuando ya se tenían todos los elementos electrónicos y todas las piezas impresas se procedió a montar la maqueta comenzando por la bancada. Esta fue fabricada en madera de pino contrachapada. Tras esto se procedió a la colocación de los mangos, anteriormente ensamblados, y por último se ensambló la electrónica procediendo con el cableado de acuerdo al esquema electrónico se había diseñado con anterioridad.

Una vez construida se decidió realizar una serie de pruebas con la maqueta para conocer su funcionamiento antes de controlarla. Las pruebas nos hicieron darnos cuenta de que iba a ser muy complicado controlar la maqueta si no se empleaban motores con reductora. Por lo tanto se procedió a realizar algún que otro cambio en la estructura de la maqueta y se decidió emplear otro tipo de motores, con mayor tensión y con reductora.

El siguiente paso fue el de controlar la maqueta. Para ello se siguieron diferentes pasos.

En primer lugar se procedió a generar la salida de los motores en Simulink. Tras esto se realizaron las pertinentes pruebas en la que se ajustaron los sentidos de giro y rango de tensión de trabajo. A continuación se procedió a obtener el valor de los sensores. Esto se realizó también en Simulink, creando una máscara adrede. En este punto fue necesario ajustar la posición de los sensores para que proporcionaran un rango adecuado de pulsos. Aquí nos topamos con el problema de que uno de los sensores no funcionaba y tras estudiar el por qué se vio que era necesario puentear las entradas de tensión del sensor. Es decir, había que puentear las entradas de 5V y 3V.

A continuación, se procedió a programar la parte de control que gobernaría la comunicación entre el mando Maestro-Esclavo. Para ello se procedió comparando el valor de los sensores y generando una señal de salida que es tratada por un controlador PID. Así la señal que este genera es enviada a la máscara que controla el motor del esclavo, que generará un movimiento espejo que imitará el movimiento del esclavo. Después de esto se programó de forma similar la comunicación entre el Esclavo y el Maestro.

Finalmente se realizaron diferentes pruebas con la maqueta y se realizaron los pertinentes retoques tanto en el programa como en la maqueta.

Para alimentar la maqueta fue necesario diseñar un montaje intermedio en entre la fuente de alimentación y la fase de potencia. La placa de control es alimentada mediante un cable USB conectado al PC.



## **2 Estado del Arte**

### **2.1 Introducción a la robótica**

A lo largo de la historia, la humanidad ha intentado facilitar su trabajo desarrollando diferentes máquinas, que de un modo u otro han ido facilitando dichos trabajos. De manera que estos artefactos han ido facilitando la vida de los hombres

Hay un sinnúmero de trabajos que a las personas no les gusta hacer, o que requieren una gran capacitación. Otros que requieren una gran cantidad de repeticiones, muchas de ellas con un sinnúmero de movimientos precisos. Acciones que los robots pueden hacer sin necesidad de descansar, dormir, comer...

Pero estas máquinas no pueden funcionar de modo autónomo y necesitan la supervisión de un humano para su correcto funcionamiento. De esta necesidad de controlar y regular el funcionamiento de las máquinas desarrolladas surgen en paralelo diferentes sistemas que garantizan su buen hacer.

Así es como surge la ingeniería de control. Esta rama de la ingeniería aparece ya en la antigua Grecia, donde destacan los mecanismos de Ktesibois, Filon y Herón. Famoso es el reloj despertador de Platón, basado en un sistema de Ktesibois, la Clepsydra, cuyo objetivo era que el nivel de un depósito de agua aumentara a una velocidad constante.

Uno de los primeros sistemas realimentados de la historia son los dispensadores de vino, cuyo funcionamiento se basa en el principio de los vasos comunicantes, logrando que el volumen de vino suministrado fuera constante en todo momento.

En la Edad Media prosigue el desarrollo de sistemas de control realimentado. Entre los diferentes sistemas destaca el sistema de control de un molino de harina realizado por H.U Lansperg hacia 1200. Este sistema regulaba que la cantidad de grano suministrada al molino dependía de la del viento y de la dureza del propio grano. En el siglo XVII se presentan los primeros sistemas de control de temperatura, donde destaca el inventor holandés Cornelis Jacobszoon Drebbel, quien fue capaz de controlar la temperatura de un horno empleando un contenedor con alcohol y un sistema de palancas.

Con la llegada de la revolución industrial se dispara el desarrollo de dispositivos de control, pudiendo ser el más importante de todos es el regulador de Watt. Este regulador controlaba el flujo de vapor que la caldera suministraba a la cabina en función de la velocidad. Este sistema incluía en un mismo ingenio un sistema de control y de acción.

En el siglo XX se desarrolla la teoría del control clásico. Esta estaba basada en métodos matemáticos como las Transformadas de Laplace y Fourier y la descripción externa de los motores. En este periodo destacan los trabajos de Niquist (1932) y Bode (1934), quienes ayudan al desarrollo de técnicas frecuenciales.

En la década de los 50, se desarrollan métodos temporales con el objetivo de solucionar los problemas de las aplicaciones aeroespaciales. Aparece entonces un nuevo método de diseño de control, denominado como Teoría de Control Moderna. Con el continuo desarrollo de las tecnologías y en concreto de la electrónica aparecen nuevos métodos de control como la supervisión, el control secuencial, analógico digital, digital directo...

Para el desarrollo de este sistema se combinaron diferentes elementos físicos, eléctricos y electrónicos; siendo todos ellos controlados por una Ingeniería de Control desarrollada para tal uso. Una de las derivadas de este razonamiento es la importancia que la Ingeniería de Control tiene hoy en día.

Así, mientras que, por ejemplo, la mecanización provee al hombre de maquinaria que reducen sus esfuerzos de trabajo; la automatización otorga el control reduce considerablemente los esfuerzos humanos mentales y sensoriales. La ingeniería de control se encargó desde un principio del control de sistemas complejos sin necesidad de la intervención humana (directa).

Los diferentes campos en los que la Ingeniería de Control tiene aplicación, como el control de procesos, control de sistemas electromecánicos, supervisión y ajuste de controladores...; hacen de la Ingeniería de Control una de las materias más interesantes de la actualidad.

## **2.2 Háptica**

### **2.2.1 Realidad Virtual**

#### **2.2.1.1 ¿Qué es la realidad virtual?**

Definir la realidad virtual es algo bastante complicado, ya que existe un sinnúmero de definiciones de diferentes investigadores, que intentan englobar en una única definición todos los conceptos que contiene el término "Realidad Virtual". Esto hace que este término sea muy relativo y en lugar de dar una única definición, se van a enumerar diferentes definiciones con el objetivo de entender que se debería entender por realidad virtual.

- Realidad Virtual es la experiencia de tele presencia, donde tele presencia es la sensación de presencia utilizando un medio de comunicación.
- Realidad Virtual es un paso más allá de lo que sería la simulación por computadores, tratándose más bien de una simulación interactiva, dinámica y en tiempo real de un sistema.
- Realidad Virtual es un entorno de tres dimensiones sintetizado por computadora en el que participantes acoplados de forma adecuada pueden atraer y manipular elementos físicos simulados en el entorno y, de alguna

manera, relacionarse con las representaciones de otras personas pasadas, presentes o ficticias, o con criaturas inventadas.

- Realidad Virtual es una simulación por computadora, dinámica y tridimensional, con alto contenido gráfico, acústico y táctil, orientada a la visualización de situaciones y variables complejas, durante la cual el usuario ingresa, a través del uso de sofisticados dispositivos de entrada, a mundos que aparentan ser reales, resultando inmerso en ambientes altamente participativos, de origen artificial. Una nueva y sorprendente forma de navegar información.
- Realidad Virtual es una manera mediante la cual los humanos visualizan, manipulan e interactúan con computadoras y datos extremadamente complejos.

En todo sistema de Realidad Virtual debe haber una serie de elementos que son la base de la misma. Estos elementos son:

- a) *La Simulación*: Es la capacidad de replicar aspectos suficientes de un objeto o ambiente de forma que pueda convencer al usuario. Los mundos simulados no necesariamente deben regirse por las mismas leyes físicas que en la realidad. Es por esta característica que la Realidad Virtual es aplicable prácticamente en cualquier campo de la actividad humana.
- b) *La Interacción*: Interacción para tener control de la exploración del sistema creado; de no existir esta interacción el sistema no dejaría de ser una película o recorrido fijado a priori. Para lograr la interacción existen diversas interfaces hombre-máquina, que van desde teclado y mouse hasta guantes o trajes sensoriales. La interactividad con el mundo virtual supone que el usuario pueda mover objetos (además de a sí mismo) y modificarlos, y que tales acciones, produzcan cambios en ese mundo artificial.
- c) *La Percepción*: La percepción es considerado el factor más importante, algunos sistemas de realidad virtual se dirigirán principalmente a los sentidos (visual, auditivo, táctil) por medio de elementos externos (Cascos de Visualización, Guantes de Datos, Cabinas, etc.); otros trataran de llegar directamente al cerebro, evitando así las interfaces sensoriales externas; y otros, los más simples recurrirán a toda la fuerza de la imaginación del hombre para experimentar una realidad virtual relativa. De este modo el usuario puede creer que realmente está viviendo las situaciones artificiales que el sistema de realidad virtual genera. La característica de percepción está asociada con el grado de inmersión que el usuario siente.

### 2.2.1.2 Clasificación de la Realidad Virtual

A la hora de clasificar la realidad virtual podremos establecer dos criterios. En primer lugar podemos clasificar la realidad virtual en función del *grado de inversión* o en función de la *interfaz con el usuario*. De esta forma podremos encontrar: [2]

#### 1. Clasificación según la Interfaz con el Usuario:

- Sistema de Ventana: Se trata de un sistema no inmersivo. Se utiliza una serie de pantallas para mostrarnos el mundo virtual. Son conocidos como WOW (Window On a World). En este tipo de sistemas el usuario puede estar completamente rodeado por pantallas o solamente tener varias pantallas delante de él. El grado de inmersión será mayor cuantas más pantallas nos rodeen.

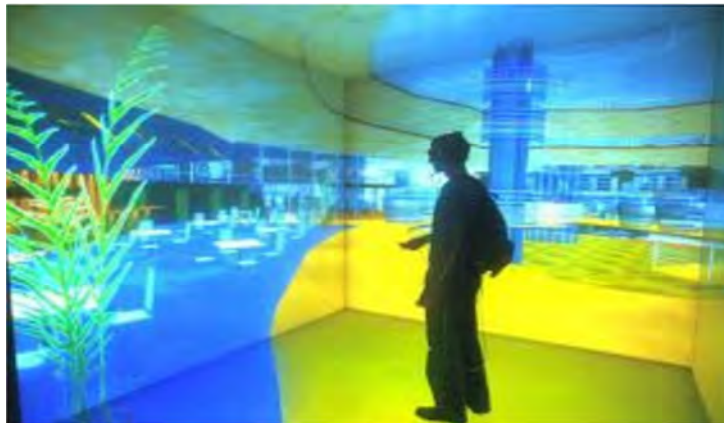


Ilustración 1: Sistema de Ventana (WOW)

- Sistema de Mapeo por Video: Se basa en la filmación mediante cámaras y el tracking mediante sistemas como los sensores inerciales, de una o más personas con la incorporación de dichas imágenes a la pantalla del computador, donde podrán interactuar con otros usuarios o con imágenes gráficas generadas por el computador. De esta forma, las acciones que el usuario realiza en el exterior de la pantalla se reproducen en la pantalla del computador permitiéndole interactuar con lo de dentro. El usuario puede, a través de este enfoque, simular su participación en diferentes formas de interacción física. Otra interesante posibilidad del mapeo mediante vídeo consiste en el encuentro interactivo de dos o más usuarios a distancia. Este tipo de sistemas puede ser considerado como una forma particular de sistema inmersivo.
- Sistemas de Telepresencia o Teleoperación: Esta tecnología conecta sensores remotos en el mundo real con los sentidos de un operador humano. Los sensores pueden estar, por ejemplo, en robots o en los extremos de herramientas. Así el usuario puede operar el equipo como si fuera parte de él.

- Sistemas de Realidad Mixta o Aumentada: Este tipo de realidad se obtiene al fusionar sistemas de Telepresencia y sistemas de Realidad virtual. Aquí las entradas generadas por el computador se mezclan con entradas de telepresencia y/o la visión de los usuarios del mundo real. Mediante el uso de un visor transparente HMD se modifica la percepción real del usuario mediante la introducción de objetos en un entorno real que solo podremos ver a través del HMD.

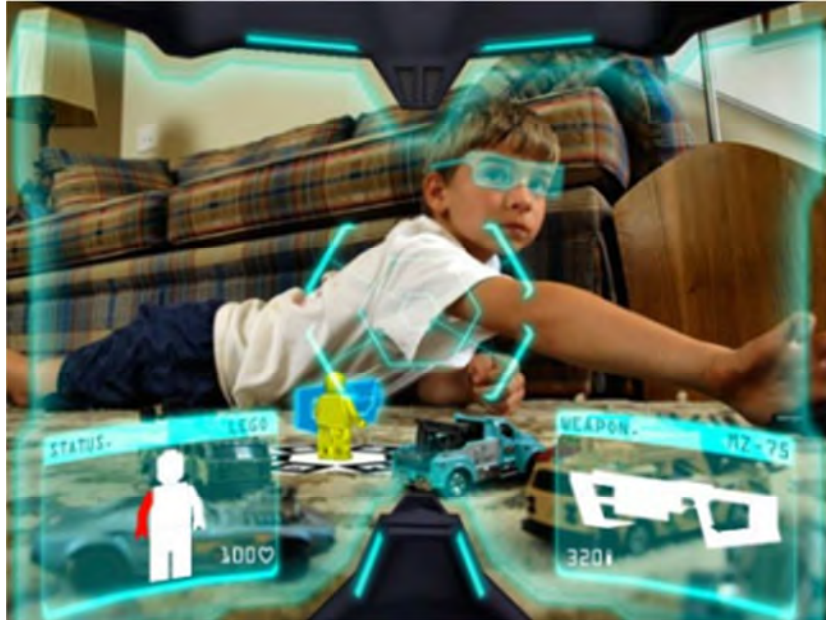


Ilustración 2: Sistema de Realidad Aumentada

- Sistema de Realidad Virtual en Segunda Persona: Este sistema no inmersivo. En este tipo de sistemas una imagen en movimiento del usuario es proyectada junto con otras imágenes en una pantalla donde el usuario puede verse a sí mismo. Para lograr esto, el participante es ubicado frente a una pantalla de video, en la cual se proyecta su imagen pero sumada con otra imagen utilizada como fondo o ambiente. A diferencia de los sistemas de inmersión, los sistemas en segunda persona involucran percepciones y respuestas en tiempo real a las acciones de los humanos involucrados, sin estar éstos sometidos al uso de interfaces inmersivos. Un ejemplo de este tipo de sistemas es un juego de golf virtual en el que el usuario se ve a sí mismo en un campo de golf golpeando una pelota virtual.

## 2. Clasificación según el Grado de Inmersión:

- Sistemas Inmersivos: Presentan al usuario un mundo virtual basado en su posición y orientación, proporcionándole estímulos artificiales a uno o varios sentidos (normalmente vista, oído y tacto) que son registrados mediante el uso de diferentes dispositivos (cascos, visores, guantes, etc.)
- Sistemas no Inmersivos: También utilizan el ordenador en el cual podemos interactuar en tiempo real con diferentes personas, en espacios y ambientes que en realidad no existen sin la necesidad de dispositivos adicionales. Nos acercamos en este caso a la navegación, a través de la cual, ofrecemos al

sujeto la posibilidad de experimentar determinados espacios, mundos, lugares, como si se encontrase en ellos. La realidad virtual no inmersiva ofrece un nuevo mundo a través de una ventana de escritorio. Este enfoque no inmersivo tiene varias ventajas sobre el enfoque inmersivo como son su bajo coste y la fácil aceptación de los usuarios.

### **2.2.1.3 Componentes de la Realidad Virtual**

Del mismo modo que no hay una única definición para el término de Realidad Virtual. No hay tampoco un único componente que represente la Realidad Virtual. Los componentes que permiten lograr la del usuario en un entorno de Realidad Virtual son:

1. **Animación 3D:** La animación 3D transmite al usuario una gran sensación de realismo. Esta sensación de ambiente 3D puede lograrse gracias a una visión 3D real utilizando cámaras adaptadas a tal efecto. La animación 3D se puede llevar a cabo en: cámaras, personajes y objetos.
2. **Visualización 3D:** En la actualidad existen un varios sistemas para lograr visión 3D, tales como “Displays auto estereoscópicos o de paralaje”, “Displays volumétricos”, “Electroholográficos”o “Multiplexado por direccionamiento”. Todos ellos permiten crear imágenes en 3D con los que el usuario puede interactuar.
3. **Sistemas de Captura de Movimiento:** La Captura de movimiento (Motion Capture) nos permite obtener un modelo biomecánico 3D computerizado del movimiento humano y analizar los ángulos y aceleraciones de los diferentes segmentos corporales. En consecuencia es posible manipular dicho modelo mediante programas. Todo esto crea múltiples campos de aplicación, como en el ámbito de la animación virtual, como en la medicina o ingeniería.
4. **Dispositivos Hápticos:** Los dispositivos hápticos son aquellos dispositivos que son tanto de entrada y salida (input/output) que sirven como interfaz entre el hombre y la máquina. Estos dispositivos permiten al usuario interactuar un objeto real o virtual incorporando el sentido del tacto. Para ello, los hápticos deben establecer entre el usuario y el entorno virtual una transferencia bidireccional de información en tiempo real. Además, este tipo de dispositivos nos da la oportunidad de simular fuerzas de contacto. A continuación se podrá ver un estudio detallado de los distintos tipos de dispositivos hápticos que existen y las características más importantes de cada uno de ellos.

## **2.2.2 Tecnología Háptica**

La tecnología háptica se refiere al conjunto de interfaces tecnológicos que interactúan con el ser humano mediante el sentido del tacto. Como se ha comentado anteriormente los dispositivos hápticos son dispositivos de entrada/salida que permiten al usuario mejorar la interacción con la Realidad Virtual.

### **2.2.2.1 Aplicaciones**

Los principales campos de aplicación son:

- **Teleoperadores y simuladores:** Los teleoperadores son controladores que controlan herramientas de forma remota, y para ello es necesario que las fuerzas de contacto existentes retornen hacia el teleoperador. A esto se le denomina teleoperancia háptica. El primer operador háptico se desarrolló en los años 50 en Argonne National Laboratory EE. UU por el Doctor Raymond Goertz. Este dispositivo permitía operar remotamente sobre sustancias radiactivas. Desde esta experiencia, el empleo de una fuerza de retorno se ha ido empleando ampliamente en muchos otros tipos de teleoperación, tal como la exploración de las profundidades marinas. Los simuladores hápticos se emplean en la actualidad en entrenamiento de operaciones médicas y en simuladores de vuelo para el entrenamiento de pilotos.
- **Entretenimiento:** Algunos dispositivos desarrollados para juegos tienen las mismas características que definen a un dispositivo háptico, como por ejemplo algunos joysticks y controladores de juego. La forma más sencilla de lograr esto en los juegos es lo que se conoce como “tabletas vibratoras” (“Rumble Packs”) gracias a las que el usuario recibe información extra.
- **Medicina:** Existen diversos interfaces hápticos en el caso de simulación médica que pueden resultar útiles a la hora de intentar minimizar los daños ocasionados por el uso de procedimientos invasivos y permitir operar de forma remota. Actualmente existen médicos expertos que pueden hacer operaciones remotas con robots y utilizando interfaces hápticas como por ejemplo el Robot DaVinci. Además, en rehabilitación se emplean diferentes dispositivos hápticos.

### 2.2.2.2 *Ventajas y Desventajas*

El uso de esta tecnología tiene diferentes ventajas y desventajas tales como:

- **Ventajas:** Gracias a esta tecnología el mundo digital puede comportarse como real. Cuando los objetos puedan ser capturados, manipulados, modificados y re-escalados digitalmente; el diseñador podrá probar las cosas y reproducir los resultados como si estuviera manejando un objeto físico.
- **Desventajas:** El principal problema es el de la depuración de los programas. Esto es muy complejo, ya que implica un análisis de datos en tiempo real. Además la precisión al tacto requiere unos grandes avances en este campo.

Por lo tanto, a pesar de las ventajas, que podrían generar grandes avances tecnológicos; son las desventajas, en temas de seguridad, las que han frenado la aplicación de esta tecnología en aplicaciones cotidianas.

### 2.2.2.3 *Clasificación de los dispositivos*

A la hora de seleccionar un interfaz háptico para una aplicación, debemos discernir, en una primera etapa, que tipo de realimentación deseamos recibir:

- **Interfaces que proporcionan un feedback de dureza:** Los interfaces que proporcionan feedback de fuerza aportan datos relacionados con la dureza, peso e inercia del objeto virtual. Los dispositivos de feedback de fuerza son:
  - a) Dispositivos Desktop: permiten la interacción virtual con el objeto virtual a través de un terminal materializado.
  - b) Exoesqueletos: permiten la manipulación directa, en múltiples puntos de contacto, de objetos virtuales con retorno de fuerza.



Ilustración 3: Cybergrasp

- **Interfaces con retorno táctil:** Los interfaces que proporcionan un feedback táctil nos permiten adquirir datos tales como la geometría del objeto virtual, su rugosidad y temperatura, entre otros. Los dispositivos que proporcionan feedback táctil son:
  - a) Dispositivos Desktop: Permiten al usuario sentir o percibir las dureza de un superficie, su rugosidad, seguir contornos lisos...
  - b) Guantes: Permiten simular con libertad de movimiento el contacto en múltiples puntos con el objeto virtual, identificando por ejemplo su textura, forma, pero no peso.





Ilustración 4: CyberTouch

En la segunda etapa, debemos analizar las características técnicas de cada uno de los interfaces comerciales disponibles en el mercado, para seleccionar el más apropiado para nuestro objetivo. Las principales especificaciones técnicas a considerar son:

- **Numero de grados de libertad:** Se debe definir el número de grados de libertad que requiere nuestro dispositivo, de tal forma que el diseño cuente con el número de grados de libertad necesarios.
- **Espacio de trabajo:** La extensión del volumen dentro del cual el manipulador puede posicionar el elemento terminal.
- **Rango de Control de fuerza:** Niveles máximo y sostenido de fuerza que ejerce el dispositivo.
- **Fricción a Parente:** Esta debe ser inferior a la mínima fuerza que podamos percibir mientras interactuamos con el entorno virtual. En caso contrario nuestra interfaz dejaría de ser transparente; y el sistema no sería capaz de discernir si las percibidas por el usuario provienen de la realimentación o de las pérdidas mecánicas del dispositivo en sí. Los valores de fricción aparente deben mantenerse en valores por debajo del 7% de las fuerzas y del 12,7 % de los pares de fuerza aplicados en la interacción con el entorno virtual. [3]
- **Rigidez:** La rigidez de un interfaz háptico se relaciona íntimamente con la habilidad del mismo para generar restricciones al movimiento del operador impidiendo que se penetre dentro de los distintos sólidos virtuales, y permitiendo, de este modo, su inspección y manipulación. La rigidez máxima dependerá de la fuerza máxima que pueda desarrollar y del mínimo desplazamiento que detecte. Un sistema háptico debe proporcionar una rigidez mínima de 20 N/m. [3]
- **Back-driveability:** con este término se alude a la transparencia del interfaz, en tanto que no debe ejercerse ninguna fuerza sobre la mano del usuario mientras no exista interacción física con el entorno virtual. Esta capacidad del sistema para seguir el movimiento de la mano del usuario rápidamente y sin oposición recibe el nombre de back-driveability.

- **Rango dinámico:** Ratio entre el máximo valor de salida del actuador frente a la fricción del mecanismo.
- **Ancho de banda:** El ancho de banda de un dispositivo puede considerarse como una medida de su calidad, en cuanto a que cuanto mayor sea éste, menores serán sus retardos en la transmisión de información, aumentando así la estabilidad general del sistema. Algunas de las variables usadas para caracterizar los interfaces “force feedback” deben utilizarse igualmente para seleccionar a los interfaces con “feedback táctil”, tales como: el espacio de trabajo, los grados de libertad, el peso, o el control del ancho de banda. De hecho, dispositivos con “force feedback” pueden también proporcionar un “feedback táctil”.

### 2.2.3 Conclusiones

Las interfaces hápticas conforman un área de investigación íntimamente relacionada con el campo de la robótica. Los grandes avances registrados en este campo en las últimas décadas han impulsado el desarrollo de dispositivos que mejoran la comunicación hombre-máquina, facilitando la interacción entre ambos mediante la aplicación de sensaciones táctiles.

La interacción háptica es un área de investigación relativamente nueva, dentro de la que varios grupos están desarrollando dispositivos que permiten al usuario la posibilidad de interacción física con un medio virtual o remoto. Los interfaces con force feedback se encuentran en un estado bastante avanzado de desarrollo en comparación con los interfaces con retorno táctil, ya que los primeros se vieron beneficiados por los avances realizados en el campo de la teleoperación; queda sin embargo, en ambos casos, un amplio campo por explorar.

Las limitaciones del hardware de los interfaces reducen la fidelidad con la que se puede simular la interacción con el mundo real; como por ejemplo: la precisión de los sensores, la capacidad de los actuadores, o la transparencia de la transmisión mecánica empleada. Otro punto a tener en cuenta es la variabilidad de las características del usuario junto con las características de la interacción del mismo con el interfaz, que pueden cambiar dinámicamente.

La seguridad es otro campo a tener en cuenta en la investigación, ya que debemos salvaguardar la seguridad del técnico ante un mal funcionamiento del computador o del dispositivo.

Otro campo en el que se centran las últimas investigaciones es el desarrollo de una referencia de software para force feedback, ya que al tiempo que se evoluciona en el hardware, se tiene que desarrollar en paralelo un software, unos modelos que sepan tratar adecuadamente la computación de las fuerzas, así como la generación de las mismas. En relación a los interfaces táctiles, uno de los principales problemas que abordan los grupos de investigación en este campo es que, en general, estos interfaces proporcionan sensaciones en un área concreta, normalmente en el extremo de los dedos. Debido a esto, es necesario

realizar estudios que identifiquen los tipos de realimentación de tipo táctil para aplicaciones específicas y la tecnología más apropiada para manifestar esta realimentación.

Otro problema de los interfaces hápticos táctiles, y una prueba de la inmadurez de la tecnología, es la ausencia de modelos generales de software que puedan ser usados para determinar las sensaciones que se generan con respecto a la interacción con el entorno, a excepción de las fuerzas de contacto, modelos para los cuales ya existen aplicaciones.

## **3 Sistema Mecánico**

Para la realización de la maqueta se empleó como modelo la maqueta desarrollada por la Universidad de Stanford. Este dispositivo fue diseñado para ser un elemento de muy bajo coste, lo cual permite un fácil acceso a él. Además se trata de un sistema de muy fácil montaje.



**Ilustración 5: Hapkit Stanford**

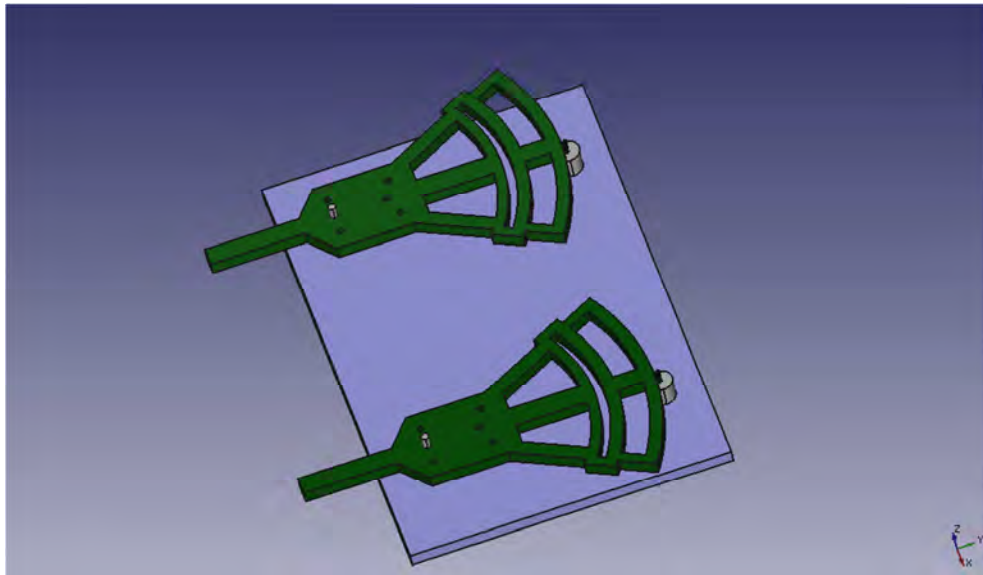
No obstante en este proyecto se ha procurado emplear elementos que se tengan en casa en la medida de lo posible. Los cuales se mencionan en el siguiente capítulo.

### **3.1 Estructura de la maqueta**

La maqueta construida es el resultado de una evolución de ideas, las cuales han confluído en un marco final. En un principio de bajaron diferentes disposiciones físicas.

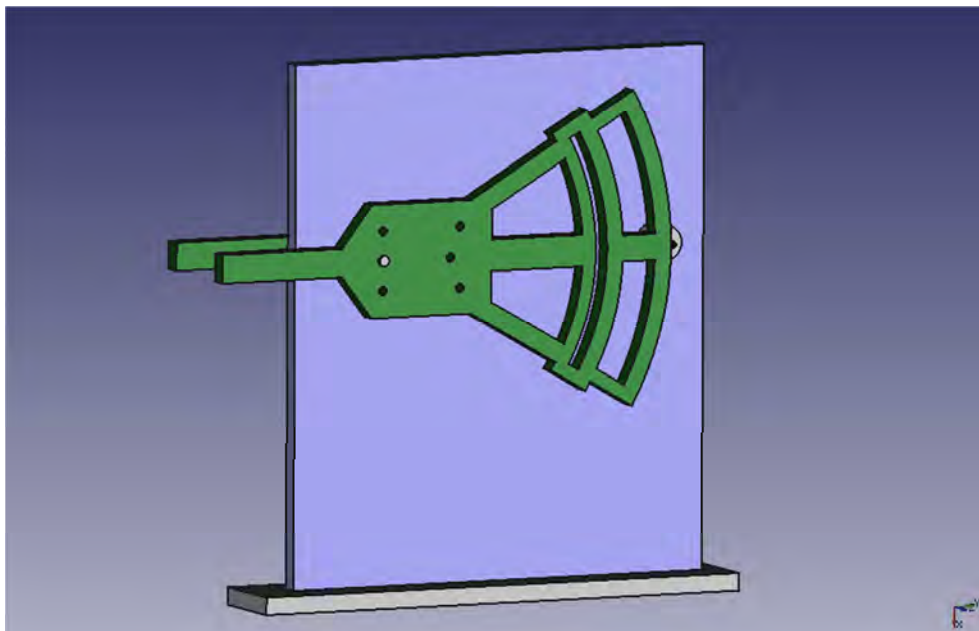
La primera idea fue colocar los mandos de la maqueta de forma horizontal. Esta idea era buena teniendo en cuenta la finalidad del sistema.

Pero este primer modelo se descartó debido al gran tamaño que finalmente han tenido los mangos. Este tamaño es el resultado de un obtener una buena relación de reducción directa entre los motores y los mangos. Esta relación de reducción se calculó de esta forma para facilitar el control. Por lo tanto, fue necesario elaborar un segundo modelo.



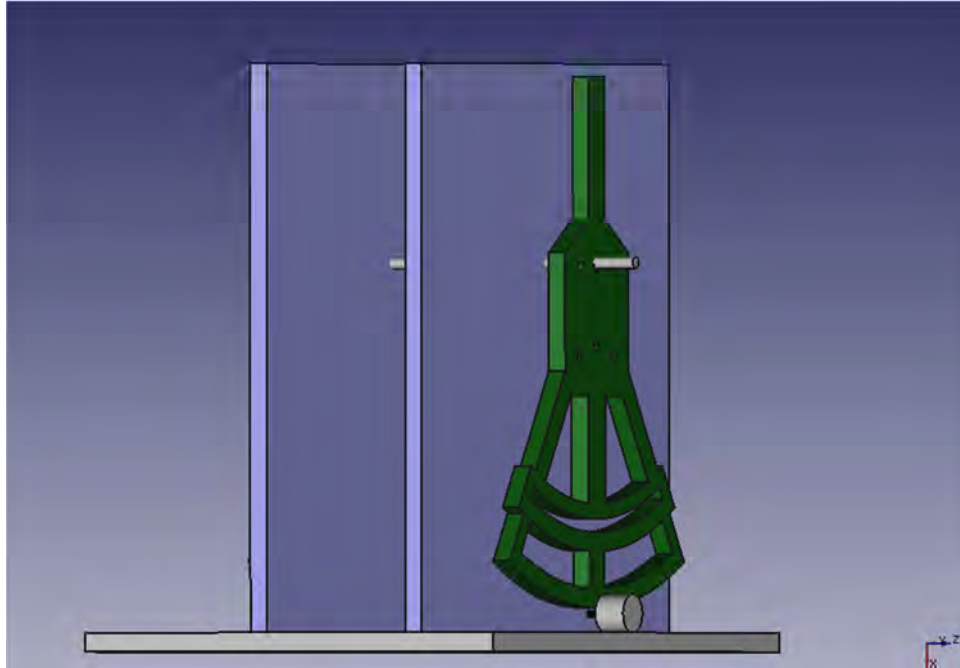
**Ilustración 6: Primer Modelo**

En este nuevo modelo se pensó situar los mangos en una nueva disposición que permitiera un mayor tamaño de los mismos sin aumentar el tamaño de la maqueta considerablemente. En este nuevo sistema los mandos estarían apoyados en una bancada vertical, ambos rotando alrededor de un mismo eje, perpendicular a la bancada vertical. Este sistema requería una compleja colocación de los motores.



**Ilustración 7: Segundo Modelo**

Los motores debían mantenerse en su posición si moverse ni cimbra, ya que de no ser así el sistema fallaría y los resultados no podrían ser correctos. De tal modo que se optó por poner los motores en la bancada de forma horizontal; así por fin se decidió disponer los mandos de la maqueta de forma vertical, de igual modo que en la maqueta de Stanford.



**Ilustración 8: Modelo Final**

De esta forma la maqueta queda mucho más estable y su montaje resulta mucho más simple. También se facilitó el emplazamiento de la electrónica de control y de potencia que quedaba situada entre las dos bancadas verticales que sustentan los mandos.

Otro de los beneficios de este sistema es la posibilidad de ajustar cada mando al motor de forma independiente. En el segundo sistema pensado, esto no era posible, ya que ambos mandos rotaban alrededor del mismo eje. En el sistema final, cada mando tiene un eje de giro independiente, y así podemos equilibrar convenientemente cada mando. Además, en este sistema se ha añadido una brida que garantiza la correcta posición de los sensores.

Por otro lado, como los mandos tenían un radio de giro bastante amplio se añadió una guía que ayudaría a equilibrar el mando. Gracias a esta guía se evitarían posibles oscilaciones y desajustes entre el mando y el motor.

## **3.2 Diseños en 3D**

Debido a lo específico de ciertos elementos empleados en el proyecto fue necesario emplear la impresión 3D. Esta hace referencia a un grupo de tecnologías de fabricación donde un objeto tridimensional es creado mediante la superposición de capas sucesivas de material. Esta es una solución barata, rápida y de fácil uso, comparada con otras tecnologías de fabricación tradicionales. [4]

Para poder obtener las piezas en 3D primero fue necesario diseñar las mismas en un software de diseño CAD.

### 3.2.1 Diseños CAD

El diseño asistido por ordenador (CAD: Computer-aided design), es el uso de un amplio rango de herramientas computacionales que asisten a diferentes profesionales como ingenieros, arquitectos y diseñadores. Estas herramientas se pueden dividir básicamente en programas de dibujo 2D y de modelado 3D. Las herramientas 2D se basan en entidades geométricas vectoriales como puntos, líneas, polígonos; con lo que se opera a través de una interfaz gráfica. En el caso de los sistemas 3D, se añaden superficies y sólidos.

En nuestro caso para diseñar las piezas se ha empleado Solid Edge. Este es un software parametrizado de diseño CAD de piezas tridimensionales. Permite el modelado de piezas de distintos materiales, doblado de chapas, ensamblado de conjuntos, soldaduras, etc. Además dispone de funciones de dibujo de plano para ingenieros.

Se optó por este software porque ya fue empleado en durante la carrera en la asignatura “Expresión Gráfica de la Ingeniería”, cursada en primero del Grado de Ingeniería en Tecnologías Industriales. Además, estaba disponible en todas las aulas informáticas de la Universidad, facilitando así el trabajo.

#### 3.2.1.1 Manivela

La Manivela es la pieza que está en contacto con el rotor de los motores. Se trata de una pieza con forma de arco con un alma que le da mayor resistencia. A medio camino, entre la parte superior y la parte inferior del sistema, encontramos una guía que ayuda a estabilizar la manivela.

El radio del arco superior es de 180 mm, el cual se ha calcula teniendo en cuenta que el radio del rotor de los motores es de 4 mm. Esto hace posible una relación de reducción de  $180mm/4mm = 45$ . De esta forma cuando el eje de la reductora girara a 1000 rpm el mango como mucho lo haría a 22,22 rpm, lo cual ayuda a controlar mejor el comportamiento de la maqueta. Para evitar que las piezas salieran de demasiado grandes con ese radio de giro, se diseñaron las piezas de tal forma que el eje de giro quedara en el mango.

Mencionar que se tuvieron que hacer dos piezas porque en la impresora empleada no cabían la manivela y el mango siendo una sola pieza.

El radio superior de la guía es de 140mm y el radio inferior de 133mm, dejando así un espacio de 7mm para la guía, que tendrá 6mm de diámetro.

Esta iba a ser una pieza que no soportaría grandes esfuerzos, pues solo tendría que apoyar en el rotor del motor. En un principio se optó por realizar una pieza más delgada, pero para garantizar un buen apoyo y un mayor rozamiento que garantizase un buen funcionamiento, se optó por diseñar la pieza con un ancho de 10mm.

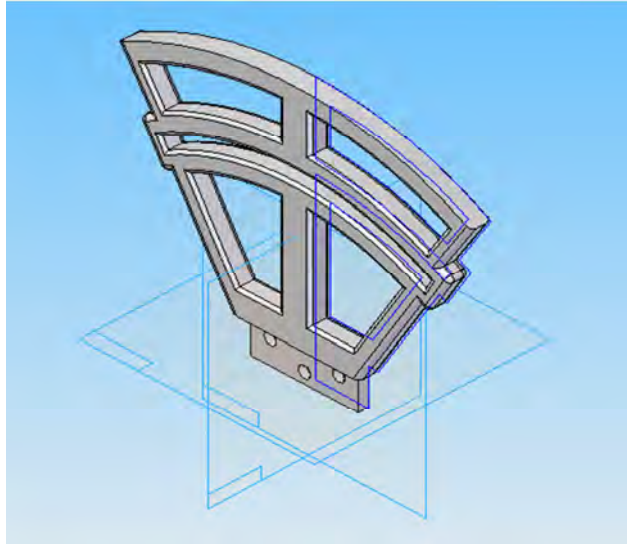


Ilustración 9: Manivela

### 3.2.1.2 *Mango*

Sobre el mando es sobre la pieza sobre la que se aplica la fuerza para poder experimentar las sensaciones hápticas. Se ha intentado realizar de la forma más ergonómica posible para así facilitar su uso y aumentar la comodidad.

Para realizar la unión entre el las dos piezas se barajaron diferentes alternativas. En primer lugar se optó por emplear una “Cola de Milano” con forma triangular, pero esta no nos proporcionaba la suficiente resistencia ni estabilidad.

La segunda opción que se barajó fue la de emplear una “Cola de Milano” esta vez rectangular. Con esta solución se obtenía una mayor resistencia y mayor estabilidad, pero después de realizar un prototipo se optó por cambiar de nuevo de diseño.

El resultado de este nuevo diseño fue es en elaborar un sistema en el que una de las partes encajara en la otra siendo posteriormente atornillada para aportar mayor resistencia.



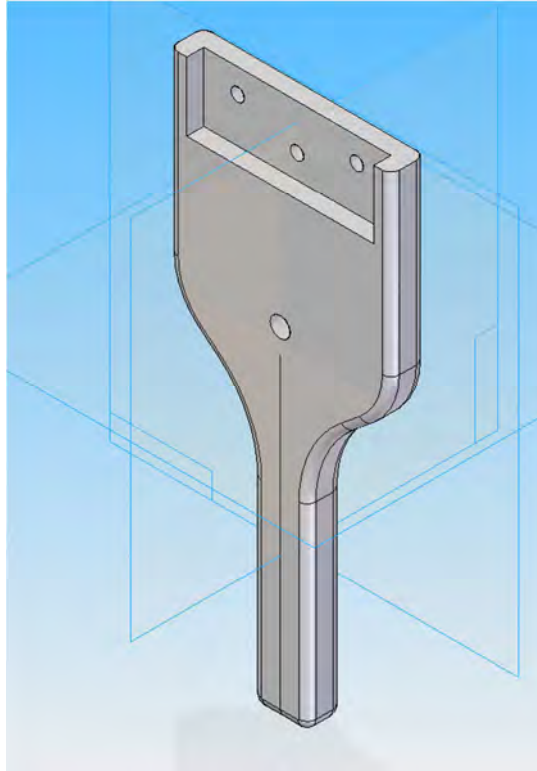


Ilustración 10: Mango

### 3.2.1.3 Porta Imanes

Para controlar el sistema convenientemente se barajaron dos posibles colocaciones para los sensores e imanes. La primera opción fue la de situar el sensor en el eje de rotación del motor. De esta forma se obtenía una gran información, pero el imán daría más de una vuelta, por lo tanto sería más complicado calcular la posición absoluta de los mandos. De esta forma estaba situado el imán en el sistema de Stanford.

La segunda opción fue la de situar el imán en el eje de giro del mango. De esta forma podemos obtener la posición absoluta de una forma mucho más sencilla. Esto nos ayudó en el control de la maqueta.

Para el diseño de esta brida se optó por elaborar un pórtico que se adaptara a las dimensiones del mando. En este punto debíamos tener en cuenta que en el eje de giro habría una arandela, que ayuda también a equilibrar los mandos. Por lo tanto el pórtico debería ser lo suficientemente largo como para evitar rozamientos con la arandela; y suficientemente alto como para no rozar con la tuerca autoblocante situada en el extremo del eje de giro.

En el centro del puente se practicó un taladro ciego de 6mm de diámetro en el que se insertó el imán una vez impresa la pieza. El agujero se hizo de 6mm de radio, al igual que el imán, por lo que para acoplar el imán al pórtico hubo que calentar el imán. De esta forma el imán quedaba perfectamente embutido en la brida.

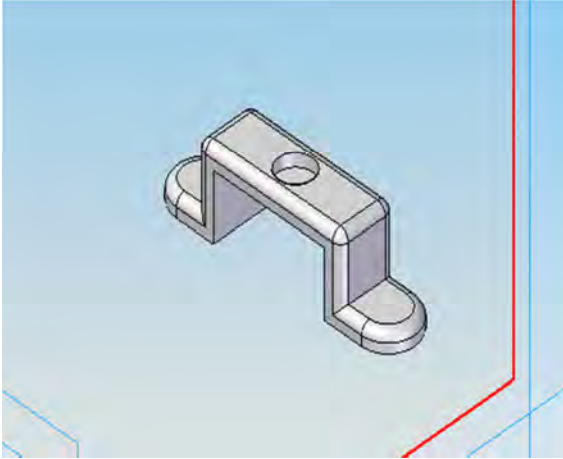


Ilustración 11: Brida Imán



Ilustración 12: imán de Neodimio

### 3.2.1.4 *Bridas PCB*

Una vez resuelto la colocación del imán, había que resolver cómo colocar los sensores de posición justo frente a los imanes. Para esto lo primero que se hizo fue colocar cada sensor, uno por mango, en una PCB, sobre la que se soldó el sensor. El hecho de colocar el sensor sobre la PCB ayudó a soldar sobre ella los cables de datos y de alimentación del sensor.

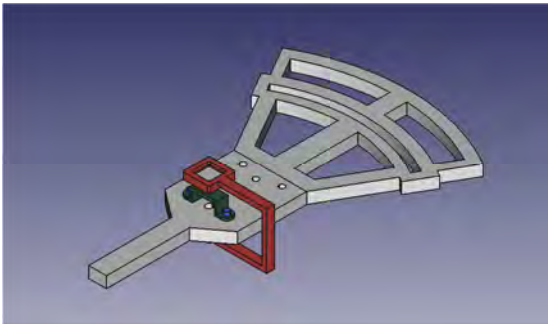


Ilustración 13: Sistema con prototipo brida PCB

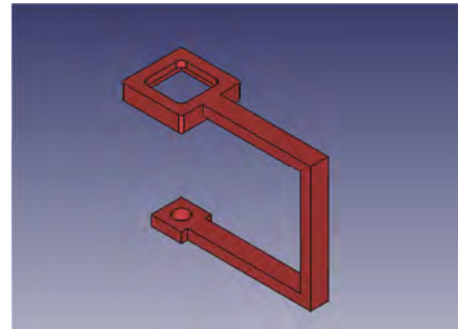


Ilustración 9: Prototipo brida PCB

En un primer momento se optó por un diseño que se sujetaba en el eje de rotación del mango, como se muestra en la ilustración 13 y en la ilustración 9.

Con este método se quería simplificar el montaje del sistema reduciendo el número de taladros que debían hacerse en la bancada vertical. Así la PCB con el sensor quedaría enfrentada al imán de forma rápida y sencilla. Para acoplar la PCB al soporte bastaría con insertar la PCB en la cavidad diseñada exclusivamente para ello.

El problema que plantaba este sistema eran las posibles oscilaciones que se pudieran generar durante el uso de la maqueta. Por lo tanto se pensó en otro sistema que otorgara una mayor resistencia y que además permitiera regular la distancia entre el imán y el sensor. Para hacer esto posible bastó con un diseño mucho más simple, pero más eficiente.

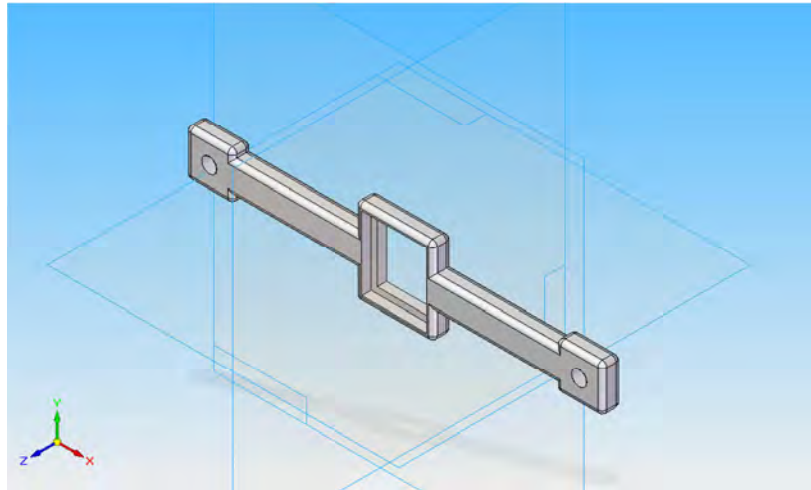


Ilustración 10: Brida PCB

La ventaja de este sistema respecto al anterior era tener dos puntos de apoyo en lugar de uno; y que estos dos puntos de apoyo se situaban en la bancada y no en el eje de rotación.

Se podrá ver en el sistema final que el hueco para la PCB en cada una de las bridas es diferente. Esto se debe a que cuando se soldaron los sensores a la PCB no se disponía de dos PCB del mismo tamaño. Como lo importante en este caso era el fin último y no el estético, se decidió emplear estas placas, ya que de esta forma nos asegurábamos el correcto funcionamiento sin una mayor demora de tiempo. Por lo tanto se decidió emplear una PCB de  $30\text{mm} \times 30\text{mm}$  y otra de  $20\text{mm} \times 20\text{mm}$ .

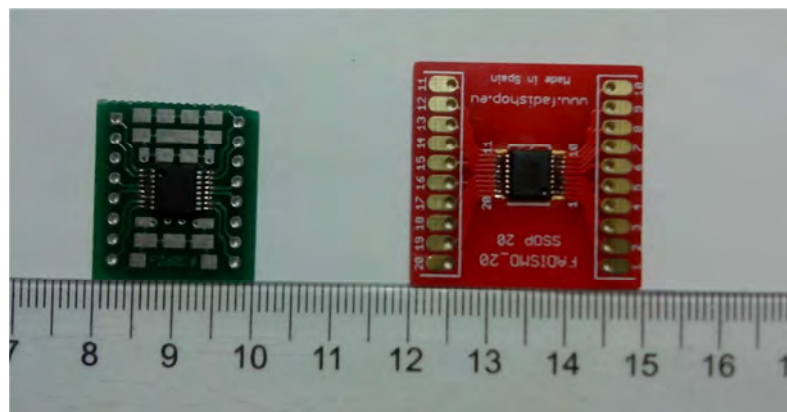


Ilustración 14: Sensores en PCB

Como iba a ser una pieza que no soportaría esfuerzos, se diseñó una pieza de ancho reducido. En concreto, para optimizar lo más posible se empleó un espesor de 5mm.

### 3.2.1.5 Brida de Ajuste

Uno de los inconvenientes de realizar la brida porta PCB de la forma final era la dificultad de alinear el imán con el sensor.

Para solucionar este problema se investigó y se buscaron soluciones en diferentes aplicaciones similares. Soluciones que de forma rápida y sencilla se pudieran aplicar a nuestro proyecto. El resultado se halló, de nuevo, en el dispositivo Hapkit desarrollado por la Universidad de Stanford.

En este sistema para ajustar el mando al rotor del motor se empleaba un sistema parecido a una palanca de segunda clase. Este sistema estaba formado por una brida, sobre la que se instalaba el mando del dispositivo háptico; con 3 orificios. Además de esto en la bancada del sistema háptico había otros 3 orificios, aunque con ciertas peculiaridades.

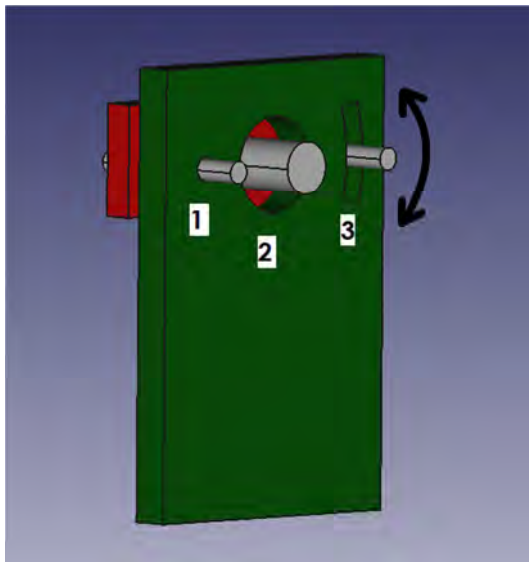


Ilustración 15: Sistema de ajuste A

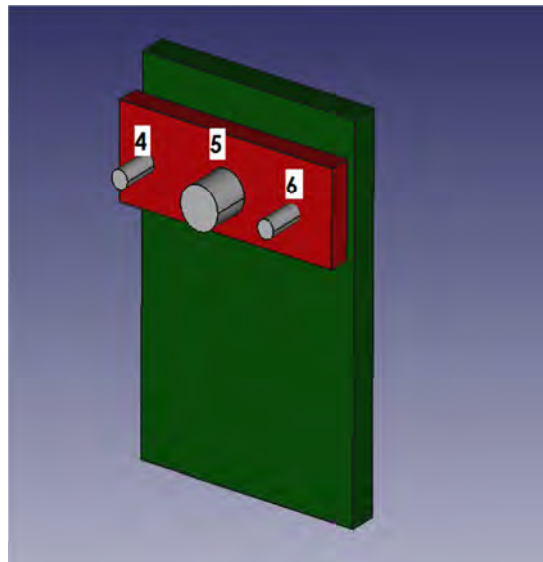


Ilustración 16: Sistema de ajuste B

Como se puede ver, en la brida (color rojo) se practican 3 orificios. En el orificio 5 se inserta el eje de giro del mango; mientras que en los orificios 4 y 6 se insertan elementos de sujeción, como pueden ser tornillos con tuerca y arandela.

En la bancada vertical (color verde) se practican también 3 orificios, pero en este caso se trata de unos orificios con diferente geometría.

- Así, el orificio Nº 1 es un orificio de igual métrica que el tronillo que se empleará para sujetar la brida. En ambos casos se emplea la misma métrica. Sobre este orificio rotará la brida, siendo este el punto de apoyo, si lo comparamos con una palanca de segunda clase.
- El orificio Nº 2 es un orificio de mayor diámetro que el del eje de giro del mango. Esto permite desplazar el mango verticalmente para poder ajustarlo al eje de rotación del

motor. Así nos aseguramos un correcto funcionamiento del mismo. Este segundo orificio sería la resistencia de la palanca de segundo grado.

- Por último encontramos el tercer orificio. Este es más bien una guía que se emplea para fijar el sistema. Cuando se haya encontrado el punto de equilibrio se ajustará este edificio para fijar la brida en la posición deseada. Así continuando con la comparativa con la palanca, este sería el punto de aplicación de potencia.

Una vez resuelto el problema se diseñó una brida con la mismas características en que el sistema de Stanford. Para realizar este diseño se tuvo en cuenta el campo de rotación del mango, para que una vez instalado no molestara.

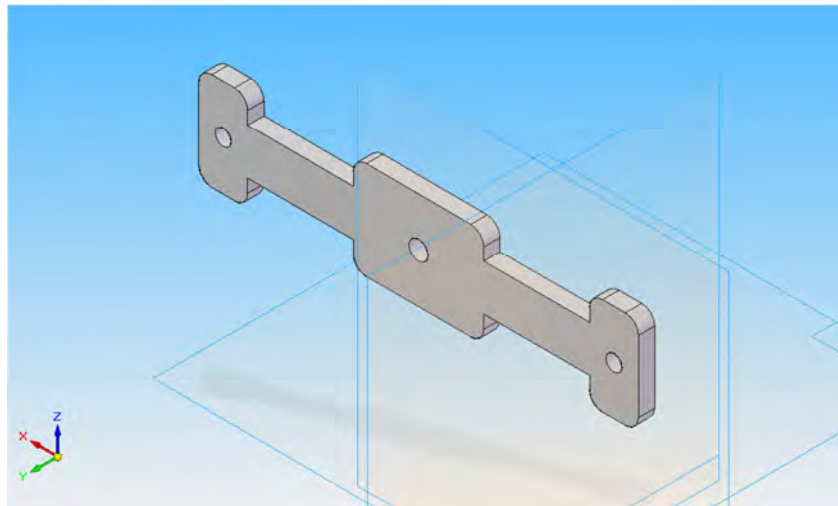


Ilustración 17: Brida de ajuste

### 3.2.2 Impresión 3D

Una vez fueron diseñadas todas las piezas, se procedió a la impresión de las mismas. En nuestro caso se empleó una impresora Prusa I3. Esta es una impresora desarrollada por el departamento de Innovación y Robótica de bq [5]. Con este diseño la empresa busca ofrecer un diseño de impresora robusto que soluciona algunas de las carencias que tenían las impresoras anteriores.

#### 3.2.2.1 Material

Uno de los puntos más importantes a la hora de realizar una impresión, es el material de impresión. Hoy en día existe una gran cantidad de plásticos y derivados para imprimir, cada uno de ellos con sus ventajas e inconvenientes. Tras analizar la finalidad de las piezas se barajaron dos materiales: ABS y PLA.

- ABS: Acrilonitrilo Butadieno Estireno. Este material era el más económico de los dos. Se trata de un termoplástico amorfo muy resistente a golpes. Su obtención es algo complicada, por lo que se le llama plástico de ingeniería. Su acrónimo se deriva de los 3 monómeros empleados para su fabricación: acrilonitrilo, butadieno y estireno. Se trata de un material con un gran tenacidad, duro y rígido con buena resistencia química. [6]
- PLA: Ácido Poliláctico o Poliláctido. Plástico biodegradable procedente de maíz o patata. Esto significa que, con el paso del tiempo y el efecto de los elementos, el plástico pierde las propiedades iniciales hasta su descomposición en elementos químicos simples, aunque esto puede tardar más de cien años. Además es reciclable, por lo que es posible su reutilización después de un proceso adecuado. Una de las ventajas del PLA es que no requiere de cama caliente para imprimir. Esto garantiza mejores acabados dimensionales. [7]



Ilustración 18: Estructura ABS [8]

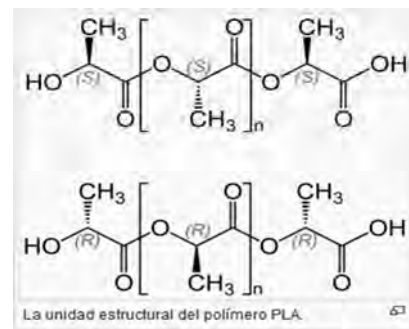


Ilustración 19: Estructura PLA [9]

	PLA	ABS
Temperatura Fusión	160-230°	215-250°
Cama Caliente	Temperatura ambiente, hasta 60°	90-103° (warping)
En Cama Caliente	Laca o cinta adhesiva	Laca o cinta adhesiva
Biodegradable	SI	NO
Reciclable	SI	SI
Estructura	Ilustración 14	Ilustración 13

Tabla 1: Características PLA y ABS

Tras tener en cuenta las características de cada material se opta por emplear el PLA. Se decide emplear este material por su menor temperatura de fusión, facilitando así una mayor velocidad de impresión. Además, carecer de cama caliente garantiza unas mejores tolerancias dimensionales. Por otro lado se trata de un material biodegradable, que junto con un menor consumo de energía, no precisa cama caliente y tiene menor temperatura de fusión, lo convierte en un material mucho más sostenible.

### 3.2.2.2 Parámetros de Impresión

Una vez decidido que material emplear queda realizar la impresión. Para realizar la impresión se empleó el programa CURA. Cura es un software que permite convertir los archivos STL que ya se han creado en pizas físicas en un solo entorno de trabajo. Se trata de un software libre que se puede descargar fácilmente de internet.

En este entorno de trabajo podemos configurar diferentes parámetros que definen como va a ser la impresión. En primer lugar, si no lo hemos hecho ya, debemos configurar nuestra impresora. Cuando ya se ha configurado lo que, al abrir el programa obtenemos la siguiente pantalla:

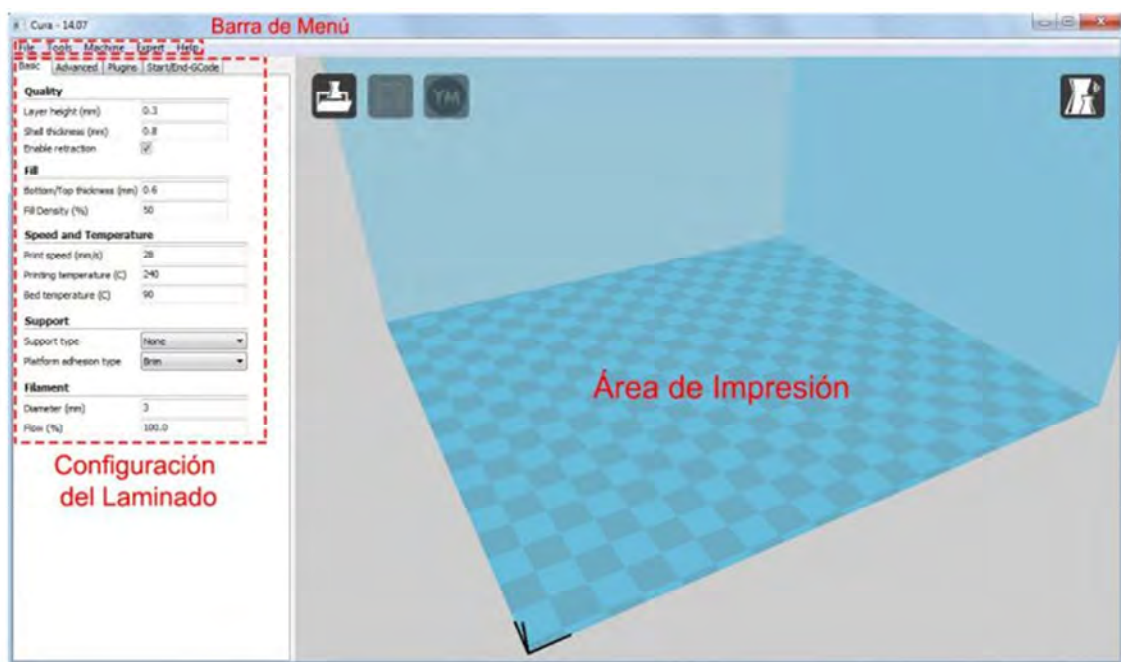


Ilustración 20: Interface cura [10]

En esta pantalla encontramos diferentes campos de interés. En nuestro caso centramos la atención en el área de configuración de laminado.

En esta sección podemos definir parámetros como la altura de capa, el ancho del objeto, la velocidad de impresión, la densidad, la temperatura del extrusor, la necesidad de material del apoyo; diámetro de la boquilla, espesor de la primera y última capa, ...

En nuestro caso dejamos todos los parámetros por defecto excepto los siguientes:

Print Speed (mm/s)	185 mm/s
Printing temperatura (C)	220 C
Fill Density (%)	40
Layer height (mm)	0.2
Shell Thickness	0.8

Tabla 2: Parámetros de impresión



### 3.2.2.3 Colocación y resultados

Tras haber configurado el entorno de trabajo de CURA, procedemos a cargar los archivos STL en el área de impresión. Para optimizar el tiempo y el área de impresión ajustamos las piezas procurando imprimir el mayor número de piezas en una misma impresión.

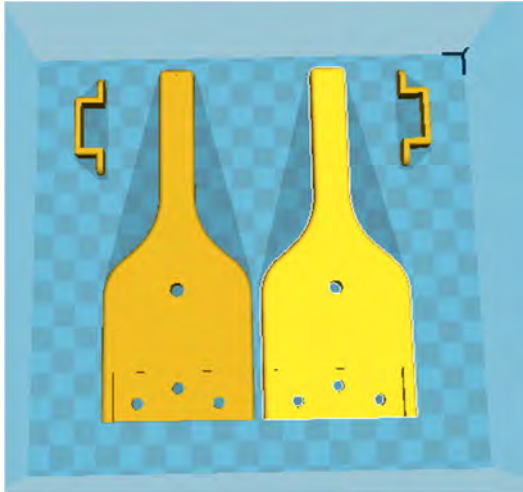


Ilustración 21: Primera impresión

Tiempo de Impresión	2 horas, 15 min
Metros de Filamento	9,56 m
Gramos de Material	76 gr

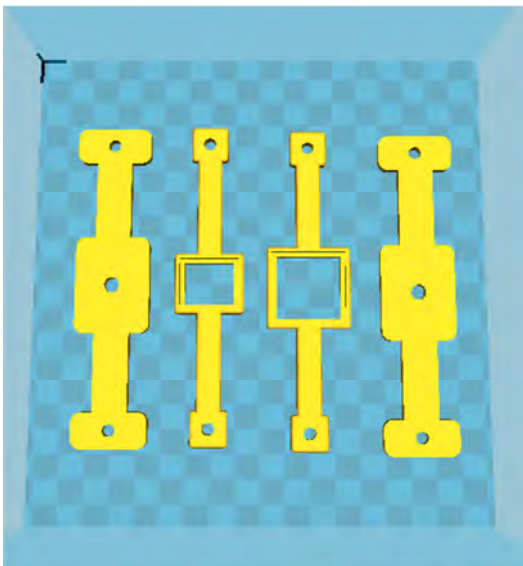


Ilustración 22: segunda impresión

Tiempo de Impresión	1 hora, 31 min
Metros de Filamento	4,88 m
Gramos de Material	39 gr

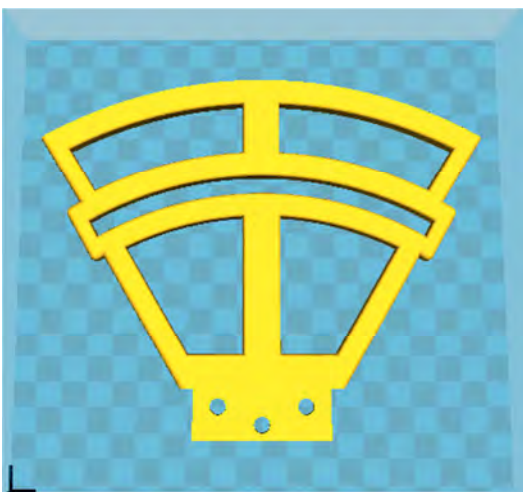


Ilustración 23: tercera y cuarta impresión

Tiempo de Impresión	2 horas, 15 min
Metros de Filamento	7,92 m
Gramos de Material	63 gr

Es este caso se deben imprimir dos piezas, luego los resultados han de multiplicarse por dos.



### 3.3 Montaje

#### 3.3.1 Materiales

Tras las realización de las piezas en 3D lo que queda ahora es montar la maqueta. Para el montaje de la misma serán necesarios los siguientes materiales.

Nombre	Cant	Descripción	Pieza
Escuadras de ángulo 50x50 mm	8	Montaje de la estructura	
Tornillos M4 x 20mm-DIN 965	34		
Tuercas M4-DIN 439	38		
Varilla M6 x 1000	1	Eje de Rotación y Guías.	
Tuercas Autoblocantes M6mm-DIN 985	12		
Arandela 6mm-DIN 440-R	16		
Tuerca Mariposa M6mm-DIN 315	4		
Tornillo M5 x 70- DIN 963	4		
Arandela M5mm-DIN 440-R	16		
Tuerca mariposa M5mm-DIN315	4		

Muelle M6 x 100	4	Diámetro Exterior: 6mm; Grosor Alambre: 0,45mm	
Tornillo M3 x 20- DIN 963	4		
Tuerca M3mm-DIN 439	4		
Arandela M4 mm-DIN 440-R	32		
Goma de Caucho	1	Suficiente para las dos manivelas	
Tablero de Contrachapado	4	140 x 260	

Tabla 3: Materiales

### 3.3.2 Herramientas

Nombre	Cant	Descripción	Pieza
Taladro Eléctrico	1		
Radial	1	Para preparar los ejes.	
Destornillador Estrella	1		
Soldador	1	Para embutir tuercas	
Alicates	1		
Pegamento Rápido	1	Para pegar la goma a la manivela	
Cola Blanca	1	Para pegar los tableros de contrachapado	



Brocas para Madera	4	M3mm; M4mm; M5mm; M6mm	
Metro	1		

Tabla 4: Herramientas

### 3.3.3 Pasos

#### 3.3.3.1 Preparación previa al montaje



Ilustración 24 : Tablero

1. Preparación de tableros: Antes de comenzar con el montaje es necesario prepara algunos elementos. En primer lugar se prepararán los tableros de contrachapado. En nuestro caso se emplearon tableros de 140 x 260. Para otorgar mayor resistencia, se pegaron los tableros dos a dos empleado cola blanca. Una vez pegados los tableros y seca la cola habrá que realizar los orificios en los que se han de colocar diferentes elementos. De acuerdo al croquis.

2. Preparación de ejes: En segundo lugar hay que a cortar la varilla de roscada con la radial para realizar los ejes de giro. Se necesitan 4 ejes de 60 mm que se contarán con la radial. En este punto debemos tener especial cuidado con los extremos. En caso contrario se estropearía la rosca y la tuerca no entraría. Para evitar esto se procede redondeando la cabeza con el disco de la radial (ilustración 25). El siguiente paso será introducir en cada uno de los ejes dos tuercas autoblocantes como indica la ilustración 26.



Ilustración 25: Detalle corte

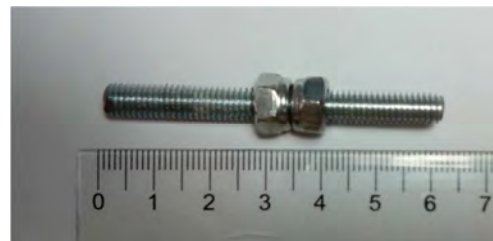


Ilustración 26: Detalle eje

3. Embutición de tuercas e imanes: Con el fin de facilitar el montaje y dar robustez al sistema se van a embutir en algunos elementos en las piezas impresas. En concreto se van a embutir los imanes en las y las tuercas que sujetaran las bridas de los imanes. Para ello se va a emplear el soldador y el destornillador.



Ilustración 27: Detalle embutición

4. Colocación de la goma de caucho: Para garantizar la fricción entre la manivela y el rotor del motor debemos pegar en la manivela una tira de goma de caucho. Esto lo hacemos porque el PLA en si no tiene coeficiente de rozamiento y el sistema patinaría. Para esto se ha empleado pegamento rápido. Otra opción era la de emplear acetona y realizar una soldadura química; pero tras realizar varias pruebas se observó que el PLA actuaba mejor con el pegamento rápido.

### 3.3.3.2 Instrucciones de montaje

1. Tablero: Cuando ya tenemos los tableros, con los taladros para los mundos y las bridas, procedemos a montar los tableros en la bancada. Para ello haremos uso del taladro con una broca para madera de  $\varnothing$  5mm. Con esta haremos los taladros que ajustarán las escuadras de ángulo. Para aportar mayor resistencia, para fijar cada escuadra se emplearán tornillos de M4 x 20 con tuerca y arandela. Para cada uno de los tableros se emplearán 4 escuadras y 12 tornillos. Con su correspondiente tuerca y arandela. El resultado será una estructura con forma de pórtico.



Ilustración 28: Detalle ensambleje

2. Mango y Manivela: Ahora se deben montar los mandos. Para montar cada uno de los mandos se necesitarán: 2 ejes, cada uno de ellos con 2 arandelas y una tuerca autoblocante (además de las dos que ya se han empleado); 3 tornillos de M4 x 20 y 3 tuercas M4. Lo primero que se debe hacer es introducir el eje en orificio central del mando interponiendo una arandela entre cada tuerca y el mando, de acuerdo a la ilustración 29. A continuación realizamos el mismo proceso con el segundo eje y la guía de la manivela. En ambos casos debemos tener cuidado y no apretar mucho los tuercas. De ser así el giro y el desplazamiento se verían impedidos.



Ilustración 29: Detalle ejes 2

A continuación unimos el mango con la manivela empleando tres tornillos que M4 x 20 con su correspondiente tuerca. En este punto, con el fin de añadir un toque estético se puede realizar un avellanado en empleando el soldador y el destornillador siguiendo los pasos descritos en el apartado de "Preparación Previa". Tras esto solo queda colocar la brida con el imán el mango.



Ilustración 30: Mando montado

Por último debemos colocar los mandos en su posición en la bancada. Para esto será necesario 4 arandelas de M6 y dos tuercas mariposas por mando. En este momento colocaremos la brida de calibración.



Ilustración 31: Mando montado 2



3. Colocación de los sensores: Para la instalación de cada sensor necesitaremos la brida del sensor, la PCB con el sensor, dos tornillos de M5 x 70mm, 8 arandelas, 2 tuercas mariposa y 2 muelles.

En primer lugar se han de adecuar los muelles al tamaño de los tornillos empleados. En este caso, para realizar este ajuste se ha elegido un tamaño un poco menor que el del tornillo, para asegurar una buena sujeción sin que llegue a ser muy costosa su instalación. Para tomar la medida al tornillo se le ha descontado el tamaño de la arandela y de la rosca.



Ilustración 32: Montaje brida PCB

Tras esto se introducen los tornillos en los orificios de las bridas introduciendo las arandelas correspondientes. A continuación se introducen los muelles y de nuevo una arandela. Por último debemos introducir los tornillos por los agujeros correspondientes, es decir por los taladros del tablero y por los de la brida. Tras esto último se introduce otra arandela y se asegura con una tuerca mariposa.

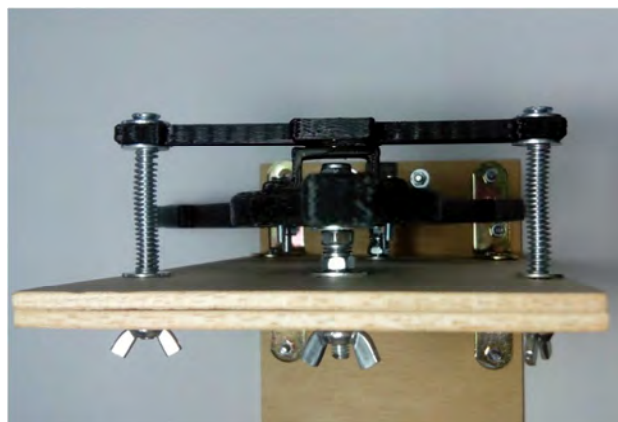


Ilustración 33: Colocación Brida PCB

Gracias a los muelles podemos ajustar la posición lineal de los sensores. Así rotando las mariposas a un lado u a otro la brida se acercará o alejará al imán. Por otro lado, gracias a la brida de calibración, el sensor y el imán siempre estarán alineados.

4. Instalación de los motores: Para la instalación de los motores en primer lugar debemos situar los motores en su posición. Para ello debemos colocar los mandos en posición totalmente vertical y hacer que el rotor del motor coincida con la posición más próxima al suelo de mando. Cuando se haya determinado esta posición, se colocará la abrazadera del motor sobre este y se marcarán los puntos en los que se deben hacer los taladros, en los que se sujetarán las bridas. Cuando se hayan realizado los taladros se fijarán los tornillos en los que se atornillarán las bridas. Estos tornillos estarán fijos y se

ajustarán con su correspondiente tuerca. El siguiente paso es colocar un trozo de goma de caucho en el centro junto entre los dos tornillos. Gracias a esta goma el motor no apoya directamente en la base de la maqueta y ayuda a sujetar el mismo. En caso de no estar la goma el motor se puede hundir en la madera. Además la goma ayuda a ajustar el la manivela y el eje del motor.



Ilustración 34: Detalle Caucho Motor

Por último solo queda colocar el motor sobre la goma y sobre este la abrazadera ajustando a nuestras necesidades con las tuercas. El beneficio de la goma es que nos permite apretar más o menos regulando, un poco, la posición del motor.

Para ajustar la posición de los motores y los mangos bastará con jugar con la brida de ajuste moviéndola de arriba abajo.

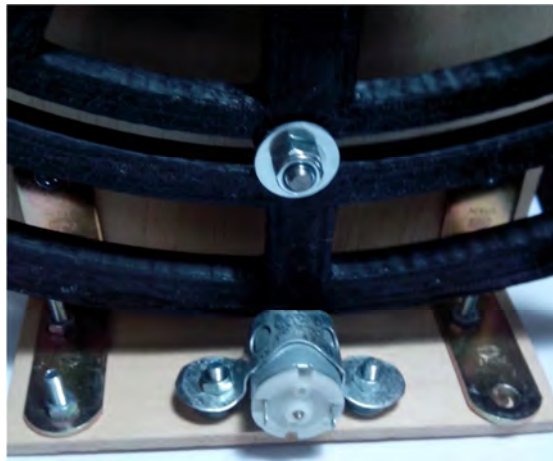


Ilustración 35: Colocación Motor

### 3.4 Ajustes

Tras la construcción de la maqueta, se procedió a la realización de diferentes pruebas para comprobar el funcionamiento. Tras ellas se concluyó que sería necesario realizar una serie de cambios, hasta llegar el resultado final. Dichos cambios fueron:

a. Eliminación de la brida de calibración:

Tras la realización de diferentes pruebas en la medida de los sensores se observó que resultaba más sencillo realizar el ajuste de los sensores si se eliminaba la brida de regulación. Esto se debe a que de esta manera la brida



del sensor se podía ajustar más rápidamente si se realizaba el ajuste en ausencia de la brida. Además, esto permitía ajustar la posición de los sensores sin tocar la posición de los mandos.

Además se modificaron sustancialmente los taladros realizados en los tableros verticales, siendo desde entonces ambos como muestra la imagen.

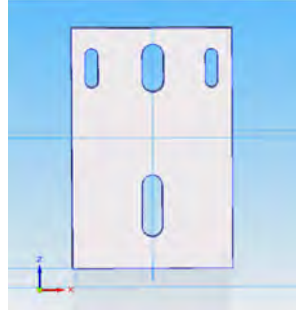


Ilustración 36: Croquis Tablero

b. Eliminación de la grúa de sujeción:

De igual modo que pasó con las la brida de sujeción se observó que la maqueta podía funcionar sin este elemento, luego se procedió a su eliminación con el fin de facilitar el montaje.

c. Cambio de los motores:

Tras finalizar la montaje tanto físico como electrónico, se comenzó a programar la maqueta empezando por la lectura de los sensores y continuando con la salida de los motores. En este punto se realizaron diversas pruebas con los motores sin reductora, pues se pensaba que de esa manera el sistema funcionaría mejor, porque la transmisión sería directa. Pero tras realizar las primeras pruebas se observó que en ciertos momentos los motores no reaccionaban bien. En un principio, cuando la diferencia angular era muy pequeña los motores no podían superar el par y el mando no se movía; y cuando la diferencia angular aumentaba, y el motor recibía mayor potencia el motor reaccionaba de forma brusca y se excedía el movimiento necesario. Esto se intentó solventar aplicando una precarga a la salida de los motores, pero aun así seguían fallando. Por lo tanto se optó por la utilización de la reductora de los motores.

Esto nos permitió aplicar mayor tensión a los motores, logrando un mejor seguimiento.

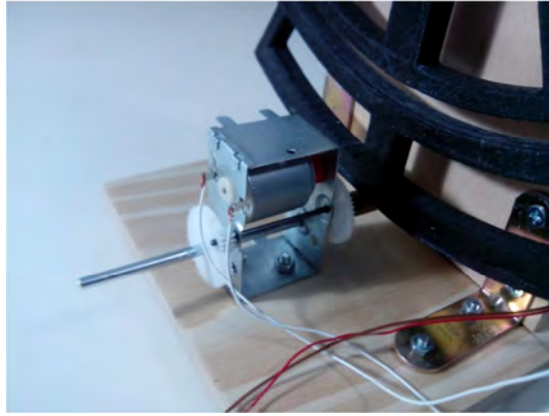


Ilustración 37: Motor con reductora

d. Cambio del tablero base:

Tras la realización de los cambios oportunos y con el fin de dar mayor estabilidad a la maqueta se procedió al cambio del tablero inferior por uno de mayor calidad y de mayor grosor.

Tablero de aglomerado chapado de 160x430x10

Tras estos pequeños cambios el sistema físico quedó finalizado.

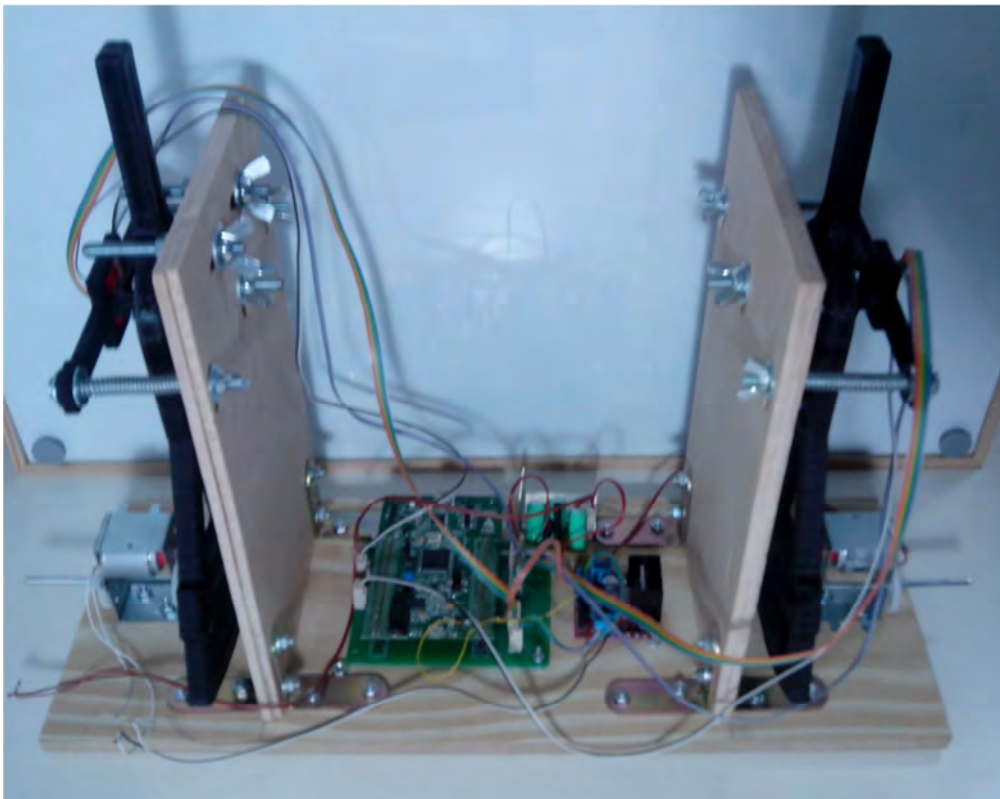


Ilustración 38: Maqueta Final 1

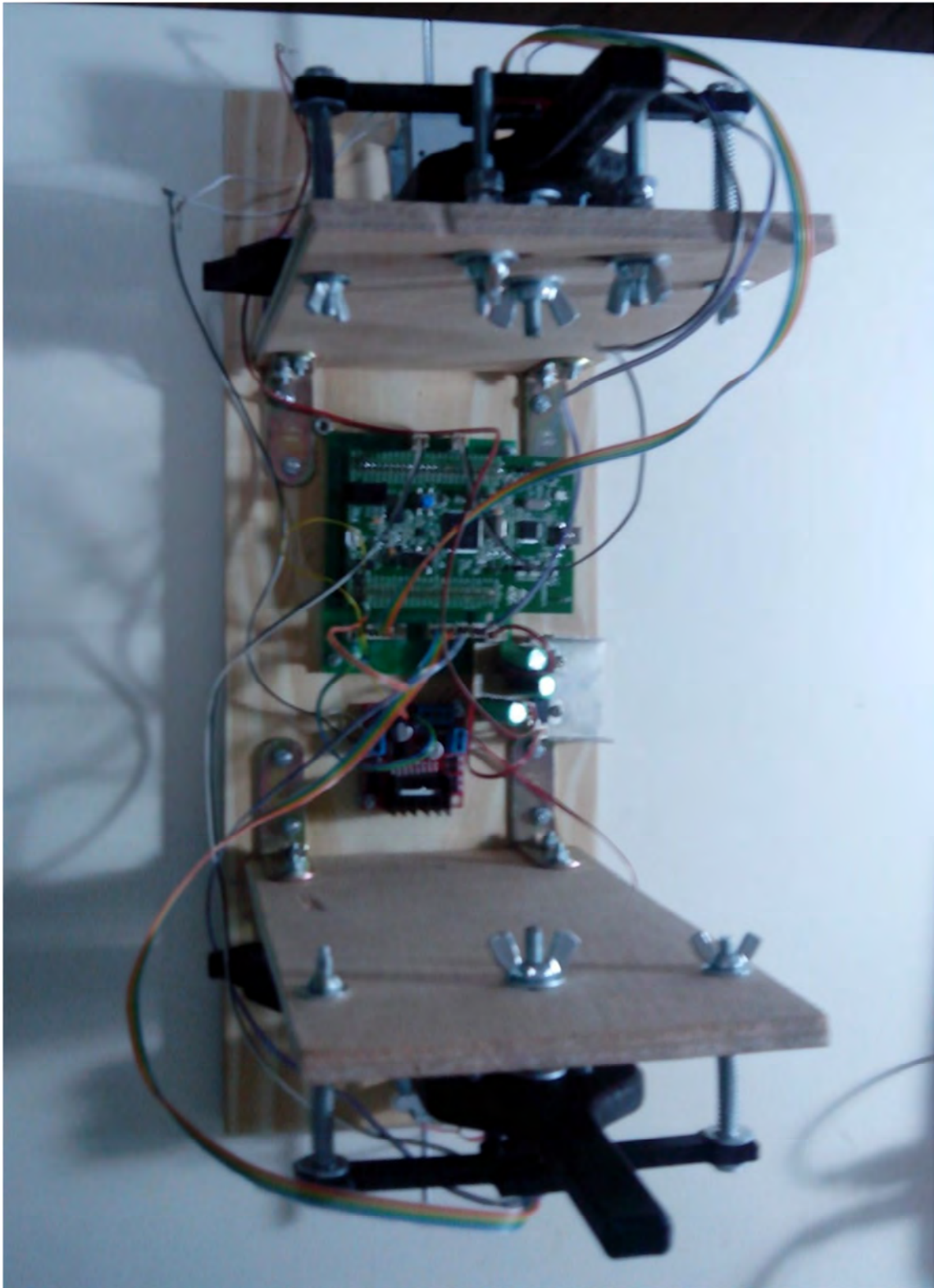


Ilustración 39: Maqueta Final 2

# 4 Sistema Electrónico

## 4.1 L298N

Se conoce por Puente en H al circuito electrónico que permite a un motor eléctrico DC girar en ambos sentidos, avance y retroceso. Son muy usados en robótica y en electrónica de potencia como convertidores. Los puentes H están disponibles como circuitos integrados, pero también pueden construirse a partir de componentes discretos [11].

El término "puente H" proviene de la típica representación gráfica del circuito. Un "puente H" se construye con 4 interruptores (mecánicos o mediante transistores).

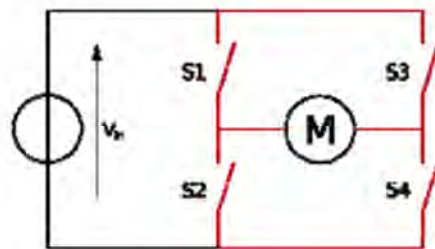


Ilustración 40: Puente H

Cuando los interruptores S1 y S4 están cerrados y S2 y S3 abiertos, se aplica una tensión positiva en el motor, haciéndolo girar en un sentido. Abriendo los interruptores S1 y S4 y cerrando S2 y S3, el voltaje se invierte, permitiendo el giro en sentido inverso del motor.

Con la nomenclatura que estamos usando, los interruptores S1 y S2 nunca podrán estar cerrados al mismo tiempo, porque esto cortocircuitaría la fuente de tensión. Lo mismo sucede con S3 y S4. Así, vamos a analizar los cuatro casos posibles, con los pares S1S4/S2S3:

S1	S2	S3	S4	RESULTADO
1	0	0	1	El motor gira en <i>avance</i>
0	1	1	0	El motor gira en <i>retroceso</i>
0	0	0	0	El motor se detiene bajo su inercia
0	1	0	1	El motor frena ( <i>fast-stop</i> )

Tabla 5: Casos Puente H

El L298N es un puente H doble de alta potencia con las siguientes características [12]:

- Corriente de operación hasta 2A por cada puente H.

- Fuente de alimentación hasta 46V.
- Fuente de alimentación para la parte lógica hasta 7V.
- Velocidad de conmutación de hasta 40KHz.
- 15 pines.

Puede trabajar con dos fuentes de alimentación diferentes, de ese modo la lógica puede trabajar con un voltaje más bajo (TTL estándar). En este integrado los puentes H pueden ponerse en paralelo para aumentar la corriente de operación. Es ideal para controlar motores de mediana potencia en proyectos de robótica. Este integrado contiene dos puentes H y por tanto con él se puede manejar hasta dos motores, como es nuestro caso.

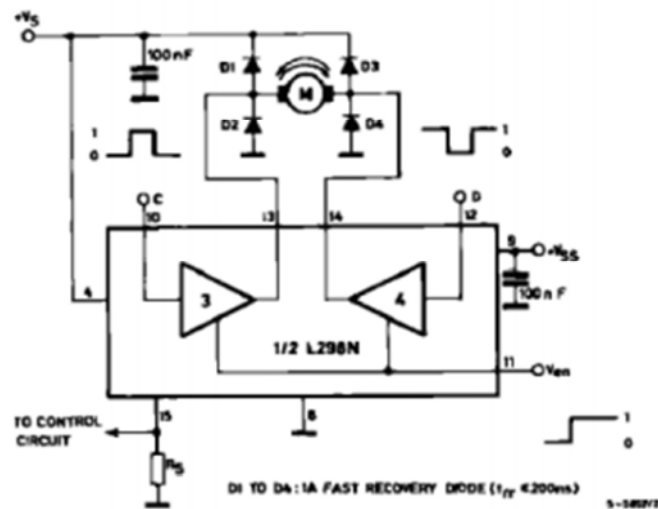


Ilustración 41: L298N

Inputs		Function
$V_{en} = H$	C = H ; D = L	Forward
	C = L ; D = H	Reverse
	C = D	Fast Motor Stop
$V_{en} = L$	C = X ; D = X	Free Running Motor Stop

L = Low

H = High

X = Don'tcare

Tabla 6: Casos L298N

El motor, gracias a este integrado, podrá ser alimentado a una tensión distinta de los 5V que emite el PIC, por lo que se resuelven los problemas propuestos. En el proyecto es utilizado como convertidor de potencia para proteger el microcontrolador frente posibles picos de corriente que pudieran dañarlo, ya que el L298N aguanta hasta 2A como se ha descrito previamente. Por este motivo es muy importante elegir unos motores que no superen esa corriente.



Esquema del integrado L298 donde se pueden ver los puentes en H y los pines:

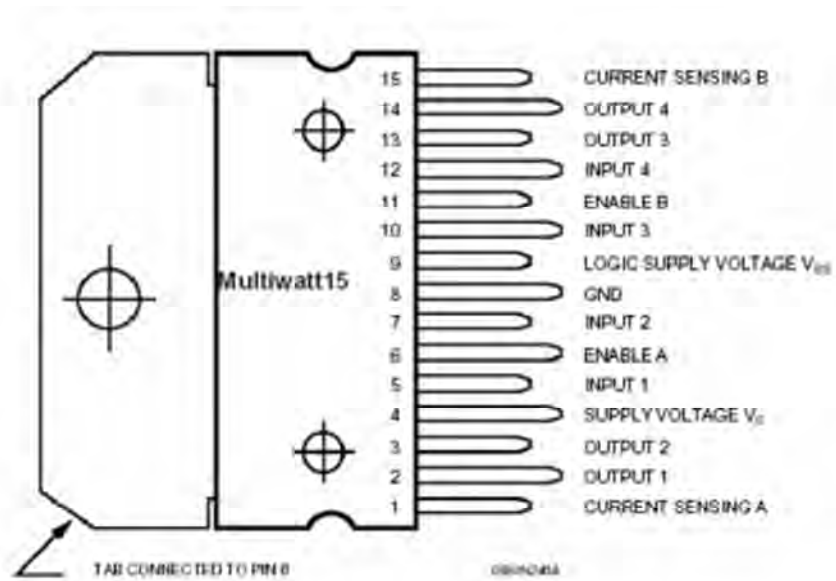


Ilustración 42: Patillaje del L298N [13]

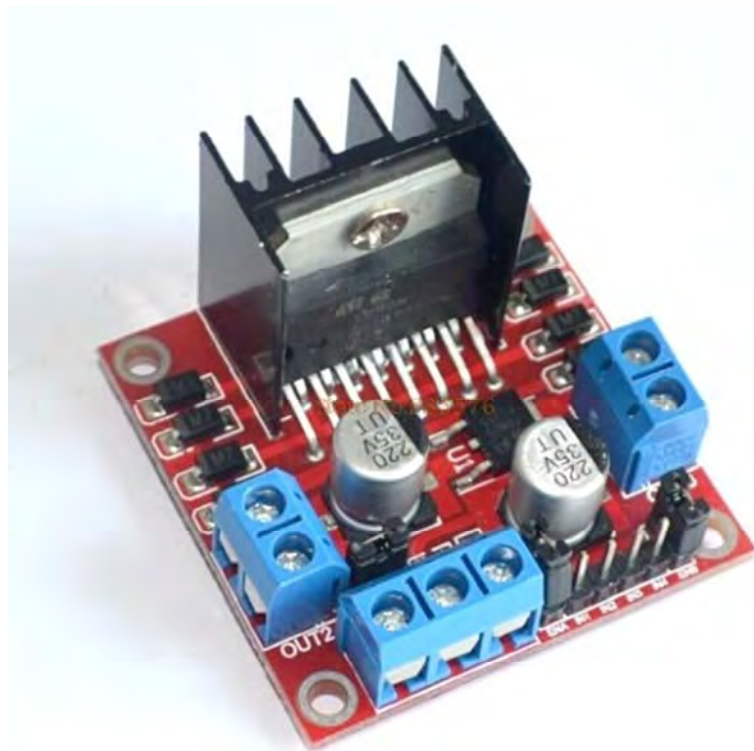


Ilustración 43: L298N

## 4.2 STM32F4 Discovery

### 4.2.1 Serie STM32

Una plataforma en general se define como aquella estructura, de software y/o hardware, sobre la cual algunas aplicaciones son capaces de ser ejecutadas. (Miranda, 2012). La serie STM32 es una familia de microcontroladores, desarrollados por *STMicroelectronics*. Las tarjetas STM32 se basan en los núcleos ARM RISC de 32 bits de *ARM Holdings*, tales como Cortex-M4F, Cortex-M3, Cortex-M0+ y Cortex-M0.

La familia STM32 está compuesta por siete series de microcontroladores: F4, F3, F2, F2, F0, L1, W. Esta familia es producida desde el año 2007 y es la tercera de *STMicroelectronics*. La siguiente tabla, resume las series de microcontroladores STM32:

Seris STM32	Núcleo CPU
F4 F3 J	ARM Cortex-M4F
F2 F1 L1 W J	ARM Cortex-M3
F0	ARM Cortex-M0

Tabla 7: Microcontroladores STM32

### 4.2.2 Descripción

La plataforma STM32F4 *Discovery* es una placa de evaluación de bajo costo para el rango de microcontroladores de STM32F4 ARM Cortex-M4. Basada en la STM32F407VGT6, que incluye una herramienta de depuración incrustada ST-LINK/V2, dos MEMS de ST, acelerómetro digital y micrófono 44 digital, un DAC de audio con el controlador de altavoz integrado clase D, LEDs y botones pulsadores y un conector USB OTG micro-AB.

La serie STM32F4 y la plataforma STM32F4 *Discovery* son producidas a partir de setiembre de 2011. Es el primer grupo basado en ARM Cortex-M4F.

### 4.2.3 Arquitectura ARM

ARM son las siglas de *Advanced RISC Machine*. La arquitectura RISC (*Reduced Instruction Set Computer*) es un tipo de diseño de CPU que se utiliza generalmente en microcontroladores. A continuación se presentan algunas características:

- Instrucciones de tamaño fijo, mostradas en un reducido número de formatos.
- Disponen de muchos registros de propósito general.
- Ejecutan instrucciones en paralelo y reducen los accesos a memoria.

El procesador ARM Cortex™-M4 es el último procesador incrustado, desarrollado por ARM, específicamente para hacer frente a los mercados de control de señales digitales que demandan un servicio eficiente y fácil de usar.

Algunas características de este procesador se enlistan a continuación:

- 1- *Floating Point Unit* o Unidad de Punto Flotante (FPU): componente de la CPU, especializado en el cálculo de operaciones en punto flotante.
  - (1) Precisión simple
  - (2) Facilidad de uso
  - (3) Mejor eficiencia de código
  - (4) Apoyo fácil para herramientas con meta-lenguaje (por ejemplo Matlab)
- 2- *Microcontroller Unit* o Unidad de Microcontrolador (MCU):
  - (1) Facilidad de uso de la programación
  - (2) Interrupción de la manipulación
  - (3) Ultra- baja potencia
- 3- *Digital Signal Processor* o Procesador de señales Digitales (DSP):
  - (1) Arquitectura *Harvard* (utiliza dispositivos de almacenamiento físicamente separados para datos e instrucciones)
  - (2) Instrucciones MAC de un ciclo

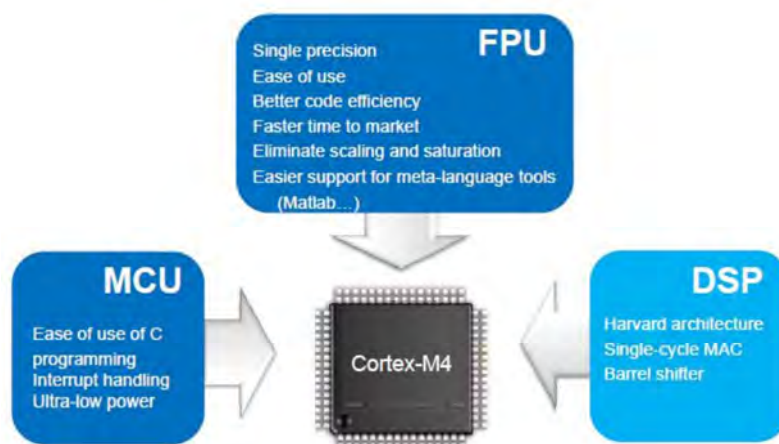


Ilustración 44: Procesador Cortex- M4 [14]



#### 4.2.4 Características de la plataforma STM32 F4 *Discovery*

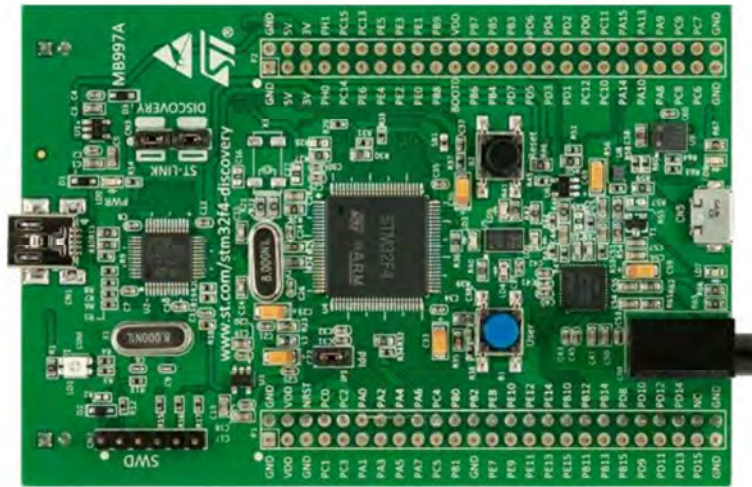


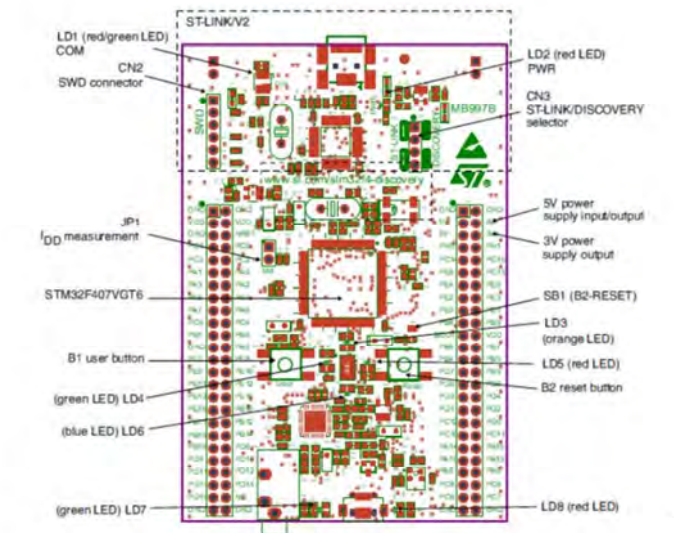
Ilustración 45: Placa STM32F4 *Discovery*

La placa STM32F4 *Discovery* ofrece las siguientes características:

- 1- Core ARM Cortex-M4F a una velocidad máxima de reloj de 180 MHz.
- 2- 192 KB de RAM estática, 64 KB de CCM (del inglés *Core Coupled Memory*), 4 KB batería de respaldo, 80 bytes de batería de respaldo con detección de manipulación de borrado.
- 3- Microcontrolador STM32F407VGT6 con 1 MB de memoria flash, que consta de 512/1024/2048 KB de uso general, 30 KB sistema de arranque, 512 bytes programables una vez (*One Time Programmable*), 16 bytes de opciones. Encapsulado LQFP100.
- 4- Cada chip tiene un número identificador de dispositivo único de 96 bits, programado de fábrica.
- 5- ST-LINK/V2 incorporado con selector usar el kit como un STLINK/V2 independiente (con conector SWD para programación y depuración).
- 6- Fuente de alimentación: a través del bus USB o desde una fuente de alimentación externa de 5 V o 3,3 V.
- 7- Sensor de movimiento ST MEMS LIS302DL, acelerómetro con salida digital de 3 ejes.
- 8- Sensor de audio ST MEMS MP45DT02, micrófono digital omnidireccional.
- 9- Audio DAC CS43L22 con controlador integrado de altavoz clase D.
- 10- 16 DMA, 6 USART, 3 SPI, 3 I2C, 3 ADC, 2 DAC, RTC, unidad CRC, Ethernet, 2 USB OTG, varios temporizadores, 2 *watchdogs* independientes.
- 11- Ocho LEDs:
  - LD1 (rojo / verde) para la comunicación USB;
  - LD2 (rojo) alimentación 3,3 V
  - Cuatro LEDs de usuario, LD3 (naranja), LD4 (verde), LD5 (rojo) y LD6 (azul)
  - 2 LEDs USB OTG LD7 (verde), VBus y LD8 (rojo)
- 12- Dos pulsadores (usuario y reset).

- 13- USB OTG FS con conector micro-AB.
- 14- Cabecera de extensión para todos LQFP100 E / S para conexión rápida de placa de prototipo y fácil sondeo.
- 15- Rango de tensión de operación: 1,8 V hasta 3,6 V.

#### 4.2.5 Descripción de componentes y pines



Nota: El pin 1 de CN2, CN3, JP1, y de los conectores P1 y P2 están indicados mediante un cuadro.

Ilustración 46: Descripción de componentes y pines

- 1- 16 DMA: canales de acceso directo a memoria
- 2- 6 USART: Transmisor/receptor asíncrono síncrono universal
- 3- 3 SPI: protocolo de datos en serie utilizado por microcontroladores para comunicarse con uno o más dispositivos periféricos rápidamente en distancias cortas, o bien para la comunicación entre dos microcontroladores.
- 4- 3 I2C: es un bus de comunicaciones en serie. Su nombre viene de *Inter-Integrated Circuit* (Inter-Circuitos Integrados). Utilizado en la industria, principalmente para comunicar microcontroladores y sus periféricos integrados (*embedded systems*) y para comunicar entre sí circuitos integrados alojados en el mismo circuito impreso.
- 5- 3 ADC: convertidor analógico- digital
- 6- 2 DAC: convertidor digital- analógico
- 7- RTC: reloj en tiempo real
- 8- Unidad CRC: inspector cíclico de redundancia
- 9- Ethernet
- 10- 2 USB OTG
- 11- Varios temporizadores
- 12- 2 *watchdogs* independientes: temporizador electrónico, cuyo funcionamiento es detectar y recuperarse de fallos.

Una de las ventajas que presenta el microcontrolador es que cada uno de los pines trabaja con una base de reloj, se puede elegir entre tres bases de reloj, cada patilla se puede inicializar y trabajar por separado. Cada una de las patillas se puede inicializar como entrada o salida de propósito general, bajo el estándar I/O o bien como GPIO.

Cada patilla tiene varios estados en los que se puede trabajar, por ejemplo: entrada, salida, alto, bajo, alta impedancia, con potencia y sin potencia, entre otras. En este proyecto, se hará uso de algunos de estos estados, tales como: alto, bajo, salida analógica asociada a un DAC, entrada analógica asociada a un ADC, entre otras.

#### 4.2.6 Características de alimentación

La plataforma de trabajo STM32F4 *Discovery*, fue diseñada para ser alimentada por medio de una conexión USB o desde una fuente de alimentación externa de 5V o 3,3V.

#### 4.2.7 Modelos STM32F4

En la siguiente ilustración se muestra un cuadro resumen con los modelos de la serie STM32F4 y las características más importantes de cada uno de ellos. El modelo STM32F407 es el que se ha empleado en este trabajo.

Main common features		STM32F429/439					
<ul style="list-style-type: none"> <li>Cortex™-M4 (DSP + FPU)</li> <li>• Up to 2x USB 2.0 OTG FS/HS</li> <li>• SDIO</li> <li>• USART, SPI, I<sup>2</sup>C</li> <li>• 2x I<sup>2</sup>S + audio PLL</li> <li>• 16- and 32-bit timers</li> <li>• Up to 3x 12-bit ADC (0.41 μs)</li> <li>• Low voltage 1.7<sup>1</sup> to 3.6 V</li> </ul>	180 MHz	Serial features as for STM32F427/437	TFT LCD with Chrom-ART™ Accelerator	SDRAM interface FMC	Serial audio interface (SAI)	2x12-bit DAC	
	STM32F427/437		168 MHz	Ethernet IEEE 1588	Camera interface	Hardware Crypto/Hash RNG <sup>2</sup> FSMC	2x12-bit DAC
	STM32F407/417		168 MHz	Ethernet IEEE 1588	Camera interface	Hardware Crypto/Hash RNG <sup>2</sup> FSMC	2x12-bit DAC
	STM32F405/415		168 MHz		Hardware Crypto/Hash RNG <sup>2</sup> FSMC		2x12-bit DAC
	STM32F401		84 MHz	<ul style="list-style-type: none"> <li>• Low power: <ul style="list-style-type: none"> <li>• Run mode down to 140 μA/MHz</li> <li>• Stop mode down to 11 μA typ</li> </ul> </li> <li>• Small form factor: down to 3 x 3 mm</li> </ul>			

Note: <sup>1</sup>/1.7 V min on WLCS90 and BGA100 package only and 1.8 min for all other packages. <sup>2</sup>/Hardware crypto/hash on F417 and F437 only




Ilustración 47: Cuadro comparativo modelos STM32 F4

- 1- STM32F401 -84 MHz CPU/105 DMIPS. Nivel de entrada de la serie STM32 F4, ofreciendo la capacidad de potencia inferior frente a los otros miembros de esta serie.
- 2- STM32F405/415 - 168 MHz CPU/210 DMIPS, hasta 1 MB de flash con conectividad avanzada y encriptación.
- 3- STM32F407/417 - 168 MHz CPU/210 DMIPS hasta 1 Mbyte de flash añadiendo Ethernet MAC y la interfaz de la cámara a la STM32F405/415.
- 4- STM32F427/437 - 168 MHz CPU/210 DMIPS, hasta 2 MB de flash añadiendo más conectividad y funciones de encriptación a la STM32F407/F417.
- 5- STM32F429/439 - 180 MHz CPU/225 DMIPS, hasta 2 Mbytes de flash, con interfaz SDRAM, pantalla LCD del controlador, Chrom-ART Accelerator, interfaz de audio, mayor rendimiento y menor consumo de energía estático en comparación al STM32F4x7 / F4x5.

#### 4.2.8 Comparación con las plataformas Arduino Uno y Raspberry Pi

Para estudiar las ventajas que presenta la plataforma STM32F4 Discovery, es importante compararla contra otras plataformas de uso cotidiano, tales como el microcontrolador Arduino Uno y el Raspberry Pi. A continuación se describen las características más importantes de las plataformas mencionadas.

- **Arduino Uno**

El Arduino UNO es un microcontrolador equipado con la placa ATmega328, posee 14 pines digitales de Entrada/Salida (seis de los cuales pueden ser usados como salidas PWM), seis entradas analógicas, un cristal que brinda una frecuencia de operación de 16 MHz, posee un puerto de conexión USB, cabecera ICSP y un botón de *reset*. Cuenta con el ATmega16U2 programado como un convertidor *USB-to-serial*. [15]

- **Raspberry Pi**

El *Raspberry Pi* es un sistema incrustado (del inglés *embedded*) que incluye un procesador central ARM1176JZF-S a 700 MHz, un procesador gráfico (GPU) y 512 MB de memoria RAM. No posee disco duro ni memoria flash, puesto que utiliza una tarjeta SD como método de almacenamiento.

A continuación, a manera de comparación, se presenta un cuadro que resume los datos más importantes de cada una de las plataformas.

Características	STM32F4 Discovery	Arduino Uno	Raspberry Pi
Microcontrolador	STM32F407VGT6	ATMega328	ARM1176JZF-S
Memoria Flash	1 MB	32 KB	SD exterior
Memoria SRAM	192 KB	1536 KB	512 MB
Pines	80 de propósito general	14 (6 PWM)	17 de propósito general
Alimentación	5 V (USB)	5 V (USB)	5 V (USB)
Tensión de operación	1,8 V- 3,6 V	5 V	3,3 V- 5 V
Frecuencia de operación	168 MHz	16 MHz	700 MHz
Compilador	Gratis	Gratis	Gratis

Tabla 8: Comparación entre STM32F4 Discovery, Arduino Uno y Raspberry Pi

Además de todo lo mencionado, se debe resaltar que la placa de control seleccionada, aunque tenga menos potencia de cálculo posee una latencia menor. Esto reduce el retardo producido por la demora en lectura y escritura de los periféricos de entrada y salida.

### 4.3 Motores

Hoy en día, en todos los ámbitos de la vida moderna es fácil encontrar dispositivos y equipos que emplean motores eléctricos para realizar un determinado trabajo. Todos ellos funcionan o por corriente continua o bien por corriente alterna. Para la aplicación desarrollada en este trabajo, se han empleado motores de corriente continua, porque el sistema requería poca potencia para poner en funcionamiento el mecanismo.

Un motor de corriente continua basa su funcionamiento en la “Ley de Fuerza de Lorentz”. Esta ley postula que cuando una partícula cargada eléctricamente se mueve dentro de un campo magnético, experimenta una fuerza perpendicular a la dirección de ese movimiento y perpendicular, a su vez, a la dirección de flujo del campo magnético [16]. Esta ley se puede demostrar empleando la “Regla de la mano izquierda” propuesta por el físico británico John Ambrose Fleming.



Ilustración 48: Motores



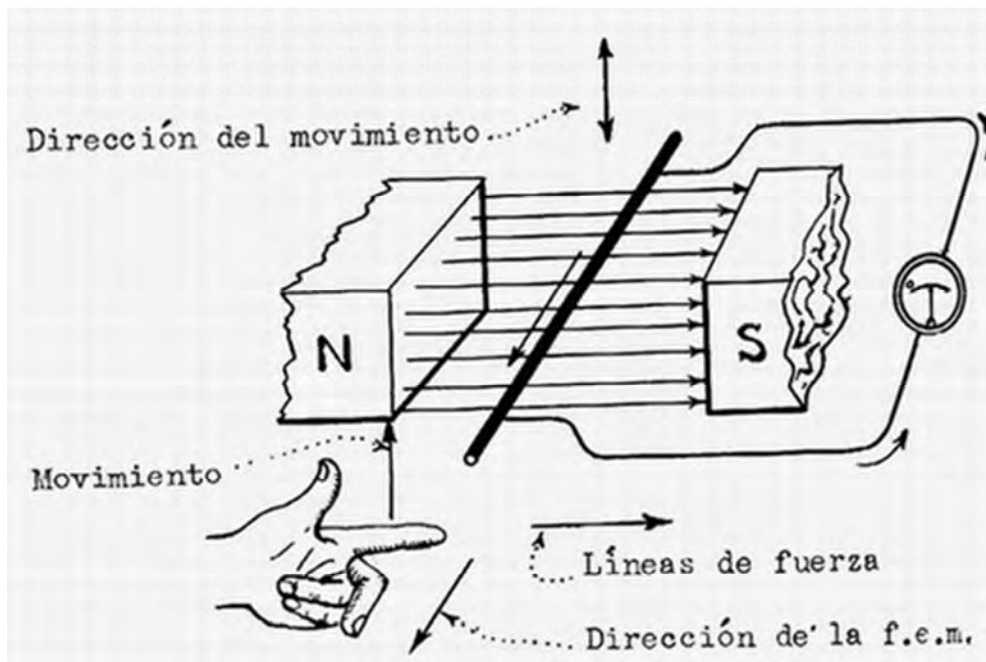


Ilustración 49: Regla mano Izquierda

En la imagen superior se puede ver como el dedo índice de la mano izquierda señala en la dirección que tienen las líneas del flujo magnético " $\phi$ " del imán permanente (del polo norte al polo sur); el dedo medio señalando el sentido de circulación de la corriente eléctrica " $I$ " a través del cable conductor creando un campo electromagnético a su alrededor; y, finalmente, el dedo pulgar señalando en la dirección " $F$ " en la que se moverá el cable cuando su campo electromagnético interactúe con el campo magnético del imán permanente. De esta forma, cuando el sentido de las líneas del campo electromagnético alrededor del cable coincidan con la dirección que tienen las líneas de fuerza del campo magnético del imán permanente (del polo norte al polo sur), el cable será rechazado (o empujado) hacia arriba o hacia abajo, de acuerdo con la Ley de la Fuerza de Lorentz. Si aplicamos la "Regla de la mano izquierda" podemos determinar la dirección en la que se moverá el cable. De esta forma podemos determinar en qué sentido girará el rotor del motor basándonos en la repulsión que ejercen los polos magnéticos de un imán permanente cuando, de acuerdo con la Ley de Lorentz, interactúan con los polos magnéticos de un electroimán que se encuentra montado en un eje. Este electroimán se denomina "rotor" y su eje le permite girar libremente entre los polos magnéticos norte y sur del imán permanente, situado dentro de la carcasa o cuerpo del motor.

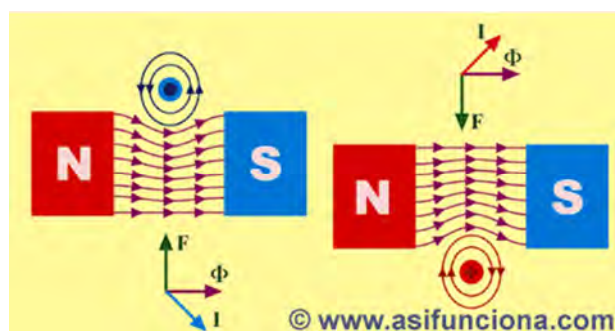


Ilustración 50: Ley de Lorentz [17]

Para la este proyecto, tras la realización de diferentes pruebas, se han empleado motores de corriente continua con una tensión de entre 1,5V y 12V, con una reductora de 23-1, para facilitar así el control de la maqueta. Gracias a esto se ha podido emplear una alimentación de 12V obteniendo así un mayor par de salida. Dicha reductora tiene un eje de trabajo doble de 40 mm con un diámetro de 4mm, al que se le ha acoplado una goma de 8mm que facilita la fricción y el movimiento entre una superficie y otra.

Esto hace que la relación de reducción final sea de 1035-1 entre el motor y el mango.

$$\text{Reductora} = 23 - 1$$

$$\text{Red. Eje} - \text{Mango} = 45 - 1$$

## 4.4 AS5045B

### 4.4.1 Introducción

El AS5045B es un sensor de posición de giro de 12 bits, anteriormente conocido como encoder, empleado para realizar mediciones angulares con una salida incremental sobre un giro completo de 360º, basado en la tecnología magnética sin contacto. Este dispositivo se adapta fácilmente a sistemas en los que en un eje de rotación se puede acoplar el imán bipolar, de tal forma que dicho imán gire sobre o bajo el sensor; pudiendo ser la posición cero programada.

Este sistema permite una resolución de 12bit, lo que da un total de 4096 estados o posiciones por revolución. Esto nos permite calcular la posición absoluta del motor con una resolución de 0,0879º. Esta medida física se traduce en una señal digital que puede ser obtenida como una serie de BITS, o como una señal PWM.

Este sensor tiene gran cantidad de aplicaciones en campos como la automática, la robótica, el control de motores...

Algunas de las características de este sensor son:

- Alta resolución para en giros superiores a 360º.
- Dos señales de 12-bits de salida:
  - (a) Como cadena de BITS.
  - (b) Como señal moduladora PWM.
- Posibilidad de programar la posición cero.
- Modo de detección de errores.
- Tolerancia a desalinización del imán.
- Pequeño encapsulado: SSOP 16 (5.3mm x 6.2m)

#### 4.4.2 Asignación de Pines

En la siguiente tabla se muestra la función de cada uno de los pines que presenta el empaquetado SSOP16:

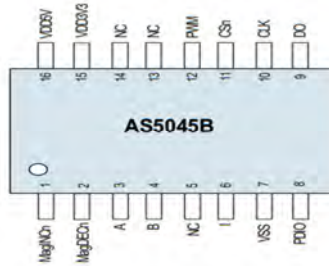


Ilustración 51: Esquema Pines

Pin Name	Pin Number	Pin Type	Description
MagINCn	1	Digital output open drain	Magnet Field Magnitude Increase. Active low. Indicates a distance reduction between the magnet and the device surface.
MagDECn	2	Digital output open drain	Magnet Field Magnitude Decrease. Active low. Indicates a distance increase between the device and the magnet.
A	3	Digital output	Quadrature output A (1024 Pulses)
B	4	Digital output	Quadrature output B (1024 Pulses)
NC	5	-	Must be left unconnected
I	6	Digital output	Index signal for the quadrature output.
Vss	7	Supply pin	Negative Supply Voltage (GND)
PDIO	8	Digital input pull-down	OTP Programming Input and Data Input for Daisy Chain mode. Pin has an internal pull-down resistor (74kΩ). Connect this pin to VSS if programming is not required
DO	9	Digital output/ tri-state	Data Output of Synchronous Serial Interface
CLK	10	Digital input, SchmittTrigger input	Clock Input of Synchronous Serial Interface; Schmitt-Trigger input
CSn	11	Digital input pulldown, Schmitt-Trigger input	Chip Select. Active low. Schmitt-Trigger input, internal pull-up resistor (50kΩ)
PWM	12	Digital output	Pulse Width Modulation
NC	13	-	Must be left unconnected
NC	14	-	Must be left unconnected
VDD3V3	15	Supply pin	3V-Regulator Output, internally regulated from VDD5V. Connect to VDD5V for 3V supply voltage. Do not load externally
VDD5V	16	Supply pin	Positive Supply Voltage, 3.0V to 5.5V

En este este proyecto se van a emplear los pines:

- 3 y 4: Estos pines son los que se emplean para obtener una diferencia incremental del ángulo.



- 7 y 8: el pin 7, como indica la tabla es el pin por que se establece la tierra del sensor. El pin está puenteado con el pin 8, que es el pin que se usa para programar la posición CERO.
- 9: salida síncrona del sistema.
- 10: señal de entrada síncrona al sistema.
- 11: Este pin (active low) selecciona un “device” en el sensor e inicia la transferencia de datos en serie. Cuando acontece un pulso a nivel alto (logic high) cambia la configuración de pin de salida (DO,9-> tri-state) y finaliza la transmisión de datos. Este pin también es empelado en el modo de alineación y el modo de programación.
- 15 y 16: Estos son los pines de alimentación. Para un correcto funcionamiento se ha puenteado.

#### 4.4.3 IMÁN

El imán empleado para generar las señales se un imán de neodimio de D6x2.5mm.



Ilustración 52: Imán de Neodimio

#### Magnetic Characteristics

Material: NdFeB  
 Energy grade: 35 MGOe  
 Temperature coeff. of  
 magnetic field strength:  $-0.15\% \text{ } ^\circ\text{C}^{-1}$   
 Max. Operating temp:  $120^\circ\text{C}$   
 Coating: Nickel plating

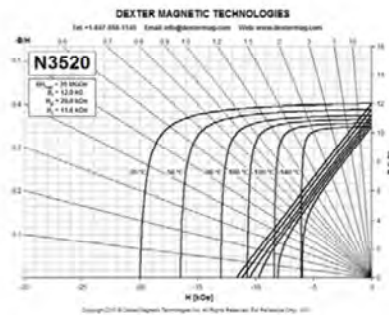


Ilustración 53: Características magnéticas

### **Magnetic Performance Data**

Magnetic flux density plot for

Use with AMS On-axis sensors:

(max Bz on line at 1.1mm from center of magnet)

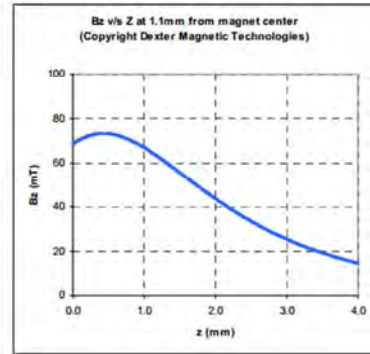


Ilustración 54: Actuación magnética

## 4.5 Conexiones

Para facilitar las conexiones en la *STM32F4 Discovery* se ha realizado una PCB en Eagle. El fin de la PCB es el hacer el sistema más robusto y sencillo. Algunas de las ventajas obtenidas han sido:

- Ausencia de soldaduras en la placa de control. Esto facilita la sustitución de la placa de control en caso de avería.
- Claridad en las uniones.
- Facilidad de realizar diferentes conexiones en un mismo PIN de la placa de control.

Para esto se ha empleado los pines que se indican en la siguiente tabla:



Motor1			
Motor2			
x10-1	gnd para LM 298		
X10-2	5V para LM 298		
X7-1	GND GENERAL		
X7-2	5V GENERAL		
X8-1	Miso M-2		
X8-2	Mosi M-2		
X8-3	SCLK M-2		
X8-4	Sensor M2	Chip select	
X9-1	Miso M-1		
X9-2	Mosi M-1		
X9-3	SCLK M-1		
X9-4	SENSOR M1	Chip select	
X3-1	VELOCIDAD MOTOR 1	Encoder relativo motor1	encoder relativo a
X3-2	VELOCIDAD MOTOR 1		encoder relativo b
X4-1	VELOCIDAD MOTOR 2	Encoder relativo Motor 2	Encoder relativo a
X4-2	VELOCIDAD MOTOR 2		encoder relativo b
X5-1	Sentido giro M1		
X5-2	Sentido giro M2		
X6-1	PWM M1		
X6-2	PWM M2		
X2-1	CND Sensor 1		
X2-2	5V Sensor 1		
X12-1	GND Sensor 2		
X12-2	5V Sensor 2		

Tabla 10: Leyenda Conexiones STM32F4 Discovery

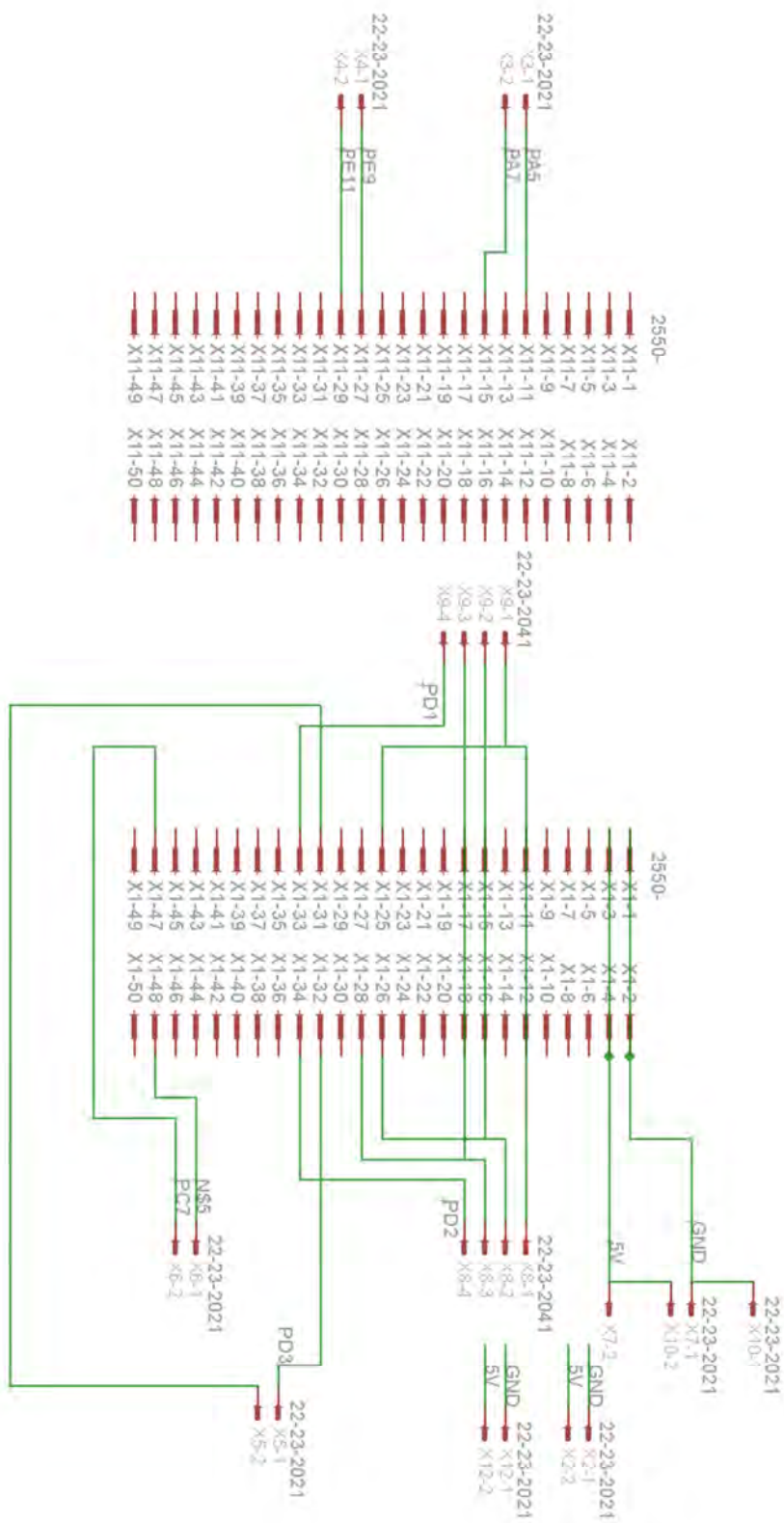


Ilustración 55: Esquema Conexiones STM32F4 Discovery

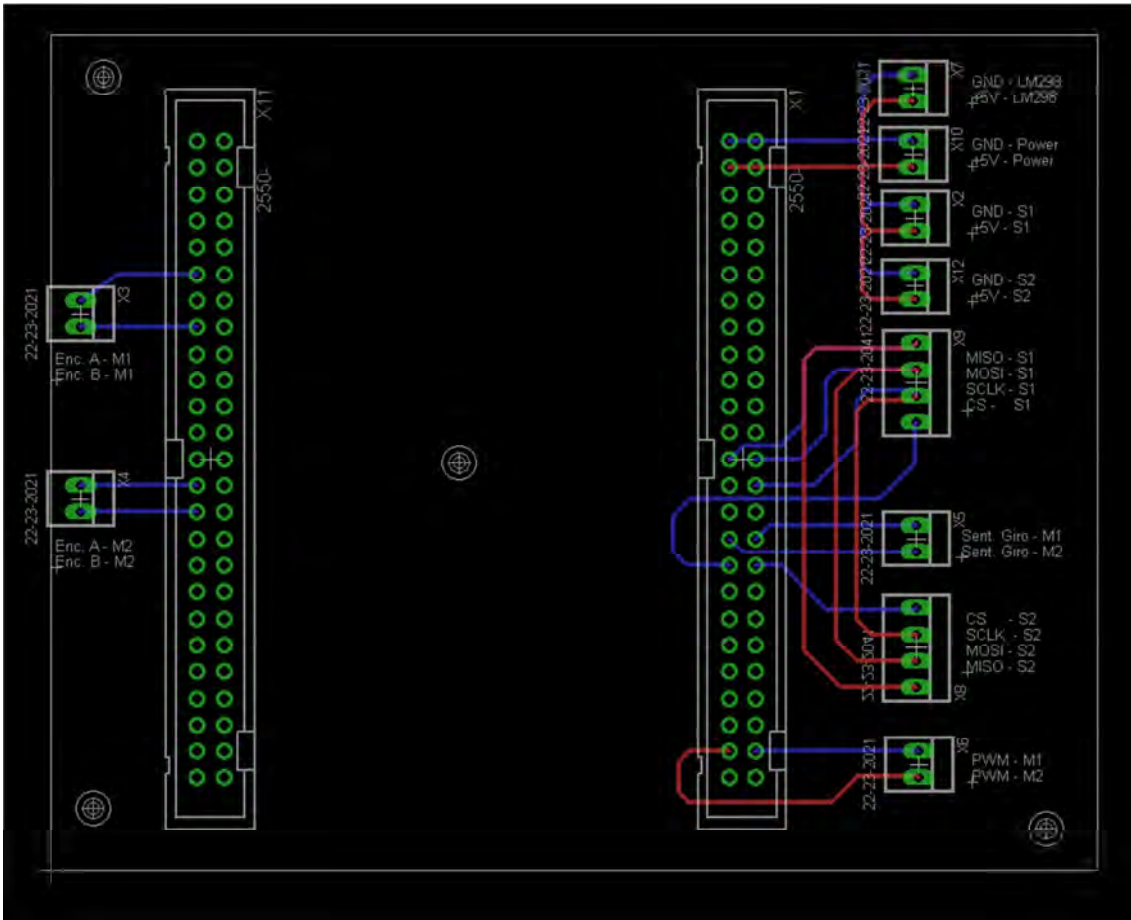


Ilustración 56: Interface PCB

# **5 Sistema de Control**

## **5.1 MATLAB/SIMULINK**

Para la programación del sistema se ha optado por emplear lenguajes de alto nivel de abstracción, que resulta ser más sencillo en comparación con los lenguajes textuales tradicionales. Este programa es *Matlab*.

En concreto se ha empleado una extensión de software de *Matlab* denominada *Simulink*. Esta proporciona un lenguaje síncrono basado en gráficos apoyado internamente por la funcionalidad del lenguaje textual *Matlab* [1]. Este software es adecuado para el desarrollo de sistemas de control en Tiempo-Real y Tiempo-Continuo. También está preparado para desarrollar modelos matemáticos, mediante lenguaje gráfico, de sistemas reales. Este software cubre una gran cantidad de dominios, lo que permite crear simulaciones para comprobar el funcionamiento de un sistema sin la necesidad de fabricarlo. Estas simulaciones también pueden programarse en un sistema de control hardware, de forma que se tenga un sistema virtual perteneciente a cualquier dominio físico.

Matlab fue creado por Cleve Morlen en 1984 basado en lenguaje M. Este lenguaje data de 1970; y se ideó para proporcionar un gran nivel de abstracción en el uso de las bibliotecas de funciones matemáticas. Simulink aparece por primera vez en 1990, bajo el nombre de Simulab, cambiando de nombre en 1992 [18].

La ventaja que proporciona este lenguaje de programación es la rapidez con la que es posible programar. La velocidad de aprendizaje es considerada y se trata de un lenguaje muy intuitivo. Esto ha propiciado una programación considerablemente rápida que nos ha otorgado más tiempo para realizar pruebas y calibraciones. Lo que ha ayudado a la optimización sistema.

## **5.2 Controlador PID**

### **5.2.1 Introducción**

Un controlador PID es un mecanismo de control por realimentación ampliamente empleado en sistemas de control industrial. Este controlador permite calcular la desviación o error entre un valor medio y un valor deseado.

El algoritmo del control PID depende de tres parámetros: el proporcional, el integral y el derivativo.

- El Valor Proporcional depende del error actual.
- El Valor Integral depende de los errores pasados.

- El Valor Derivativo es una predicción de los errores futuros.

La suma de estas tres acciones es usada para ajustar al proceso por medio de un elemento de control como la posición de una válvula de control o la potencia suministrada a un calentador.

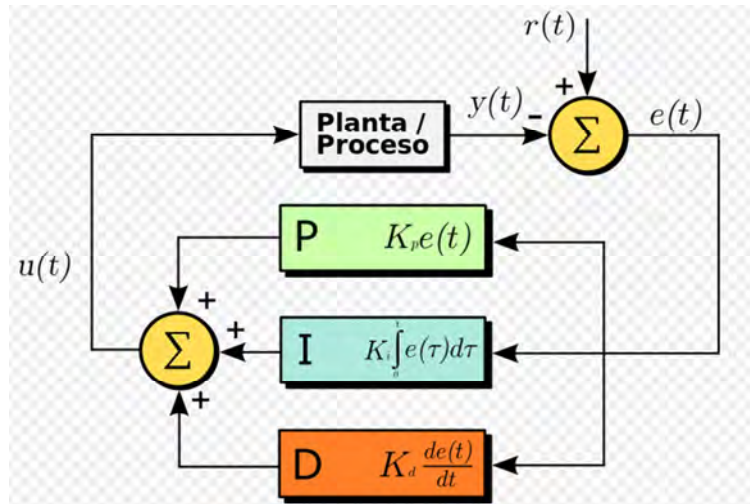


Ilustración 57: Esquema PID

Cuando no se tiene conocimiento del proceso, históricamente se ha considerado que el controlador PID es el controlador más adecuado. En este controlador, ajustando estas tres variables en el algoritmo de control del PID, el controlador puede proveer una acción de control diseñado para los requerimientos del proceso en específico. La respuesta del controlador puede describirse en términos de la respuesta del control ante un error, el grado el cual el controlador sobrepasa el punto de ajuste, y el grado de oscilación del sistema. No obstante el uso del PID para control no garantiza control óptimo del sistema o la estabilidad del mismo. [19]

### 5.2.2 Funcionamiento

Para el correcto funcionamiento de un controlador PID que regule un proceso o sistema se necesita, al menos:

- Un sensor, que determine el estado del sistema (termómetro, caudalímetro, manómetro, etc).
- Un controlador, que genere la señal que gobierna al actuador.
- Un actuador, que modifique al sistema de manera controlada (resistencia eléctrica, motor, válvula, bomba, etc).



El sensor proporciona una señal analógica o digital al controlador, la cual representa el punto actual en el que se encuentra el proceso o sistema. La señal puede representar ese valor en tensión eléctrica, intensidad de corriente eléctrica o frecuencia.

El controlador lee una señal externa que representa el valor que se desea alcanzar. Esta señal recibe el nombre de punto de consigna (o punto de referencia), la cual es de la misma naturaleza y tiene el mismo rango de valores que la señal que proporciona el sensor. El controlador resta la señal de punto actual a la señal de punto de consigna, obteniendo así la señal de error, que determina en cada instante la diferencia que hay entre el valor deseado (consigna) y el valor medido. La señal de error es utilizada por cada uno de los 3 componentes del controlador PID. Las 3 señales sumadas, componen la señal de salida que el controlador va a utilizar para gobernar al actuador. La señal resultante de la suma de estas tres se llama variable manipulada y no se aplica directamente sobre el actuador, sino que debe ser transformada para ser compatible con el actuador utilizado.

Las tres componentes de un controlador PID son: parte Proporcional, acción Integral y acción Derivativa. El peso de la influencia que cada una de estas partes tiene en la suma final, viene dado por la constante proporcional, el tiempo integral y el tiempo derivativo, respectivamente. Se pretenderá lograr que el bucle de control corrija eficazmente y en el mínimo tiempo posible los efectos de las perturbaciones.

1. Acción de Control Proporcional: da una salida del controlador que es proporcional al error, es decir:  $u(t) = KP * e(t)$ , que descrita desde su función de transferencia queda:  $C_p(S) = K_p$ , donde  $K_p$  es una ganancia proporcional ajustable. Un controlador proporcional puede controlar cualquier planta estable, pero posee desempeño limitado y error en régimen permanente.
2. Acción de Control Integral: da una salida del controlador que es proporcional al error acumulado, lo que implica que es un modo de controlador lento. La señal de control  $u(t)$  tiene un valor diferente de cero cuando la señal de error  $e(t)$  es cero. Por lo tanto se concluye que dada una referencia contante, o perturbaciones, el error en régimen permanente es cero. [20]

$$u(t) = K_i \int_0^t e(t)Dt$$

$$C_i(s) = \frac{K_i}{s}$$

3. Acción de control proporcional-integral: se define mediante:

$$u(t) = K_p e(t) + \frac{K_p}{T_i} \int_0^t e(t)dt$$

donde  $T_i$  se denomina tiempo integral y es quien ajusta la acción integral. La función de transferencia resulta:

$$C_{PI}(s) = K_p \left( 1 + \frac{1}{T_{is}} \right)$$

Con un control proporcional, es necesario que exista error para tener una acción de control distinta de cero. Con acción integral, un error pequeño positivo siempre nos dará una acción de control creciente, y si fuera negativo, la señal de control será decreciente. Este razonamiento sencillo nos muestra que el error en régimen permanente será siempre cero.

Muchos controladores industriales tienen solo acción PI. Se puede demostrar que un control PI es adecuado para todos los procesos donde la dinámica es esencialmente de primer orden.

4. Acción de control proporcional-derivativa: se define mediante:

$$u(t) = K_p e(t) + K_p T_d \frac{de(t)}{dt}$$

donde  $T_d$  es una constante denominada tiempo derivativo. Esta acción tiene carácter de previsión, lo que hace más rápida la acción de control, aunque tiene la desventaja importante que amplifica las señales de ruido y puede provocar saturación en el actuador. La acción de control derivativa nunca se utiliza por sí sola, debido a que solo es eficaz durante periodos transitorios. La función transferencia de un controlador PD resulta:

$$C_{PD} = K_p + sK_p T_d$$

Cuando una acción de control derivativa se agrega a un controlador proporcional, permite obtener un controlador de alta sensibilidad, es decir que responde a la velocidad del cambio del error y produce una corrección significativa antes de que la magnitud del error se vuelva demasiado grande. Aunque el control derivativo no afecta en forma directa al error de estado estacionario, añade amortiguamiento al sistema y, por tanto, permite un valor más grande que la ganancia  $K$ , lo cual provoca una mejora en la precisión en estado estable. [21]

5. Acción de control proporcional-integra-derivativa: esta acción combinada reúne las ventajas de cada una de las tres acciones de control individuales. La ecuación de un controlador con esta acción combinada se obtiene mediante:

$$u(t) = K_p e(t) + \frac{K_p}{T_i} \int_0^t e(t) dt + K_p T_d \frac{de(t)}{dt}$$

y su función transferencia resulta:

$$C_{PID}(s) = K_p \left( 1 + \frac{1}{T_i s} + T_d s \right)$$

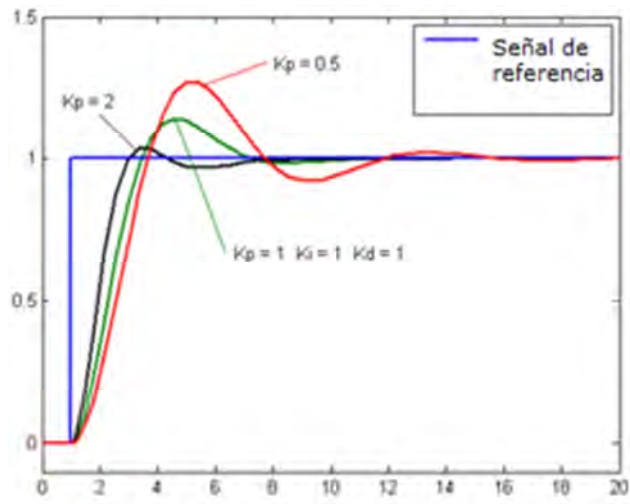


Ilustración 58: Proporcional

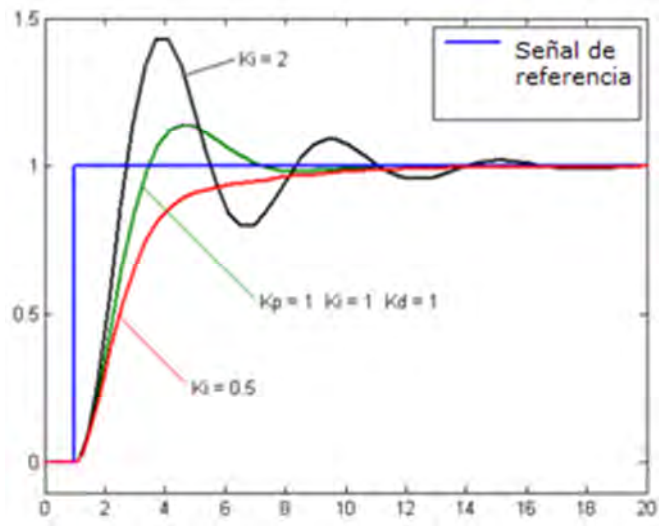


Ilustración 59: Integral

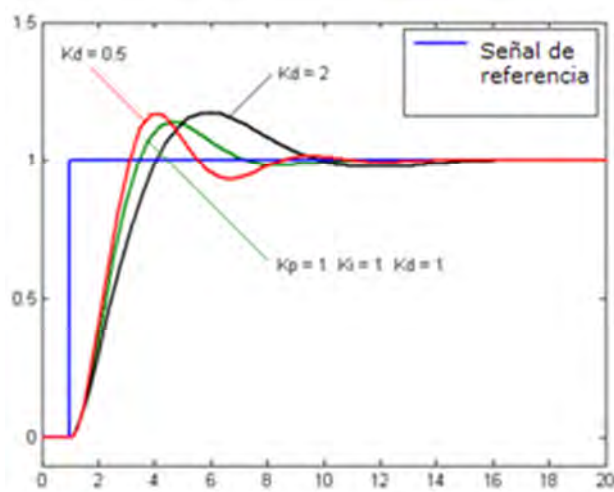


Ilustración 60: Derivativo

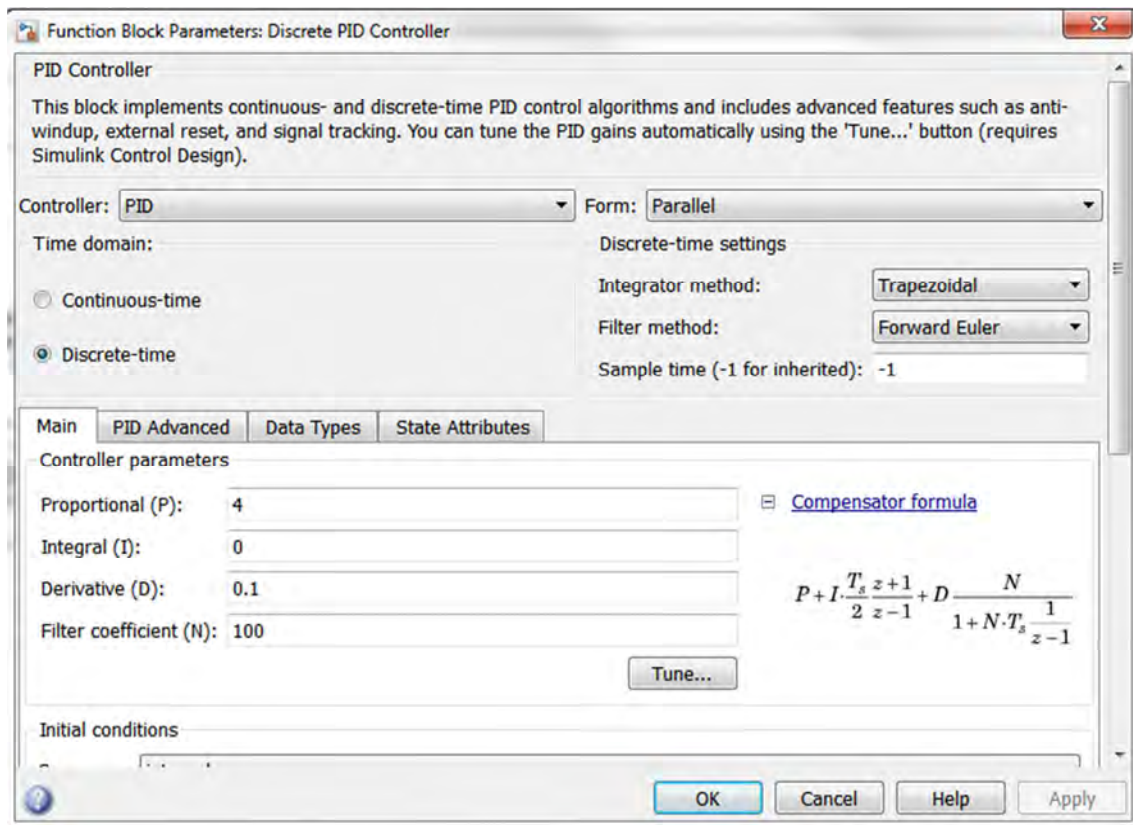


Ilustración 61: Regulador PID

## 5.3 Programa

En este apartado se procederá a la explicación del código. Para ello lo primero que debemos conocer es como trabaja Simulink. Como ya se ha mencionado en el apartado anterior, Simulink es una herramienta de gran utilidad para la simulación de sistemas dinámicos. Principalmente, se trata de un entorno de trabajo gráfico, en el que se especifican las partes de un sistema y su interconexión en forma de diagramas de bloque.

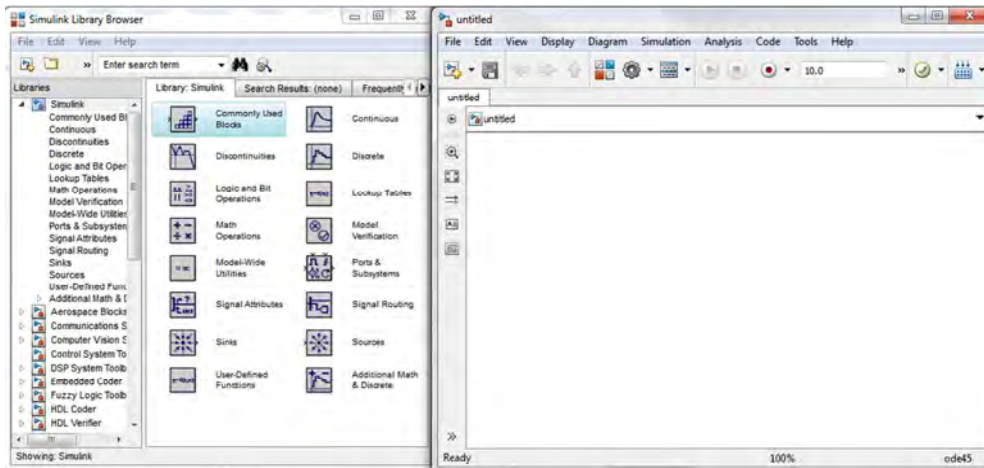
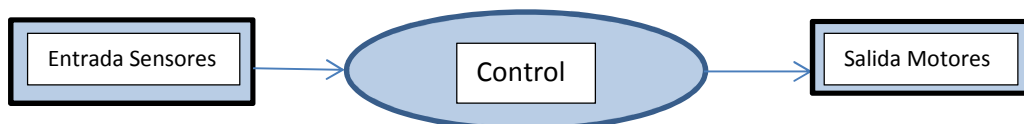


Ilustración 62: Entorno de Trabajo Simulink

En el espacio de trabajo Simulink se pueden ir creando modelos insertando los bloques correspondientes, los cuales se encuentran la amplia de librería de Simulink. Una de la grandes ventajas de Simulink es que puedes generar tus propios bloques de trabajo, de igual forma que puedes generar librerías en otros entornos de programación, por ejemplo C++. Esto facilita mucho la labor del programador, que puede crear estándares que podrá usar en trabajos futuros. [22]

Una vez introducido y conocido el entorno de trabajo se procedió a la programación del sistema. Para ello se comenzó creando un modelo de entrada y otro de salida, entre los que se realizaría el control. Así pues, dado una señal de entrada, tras un proceso de síntesis, se obtiene una señal de salida.



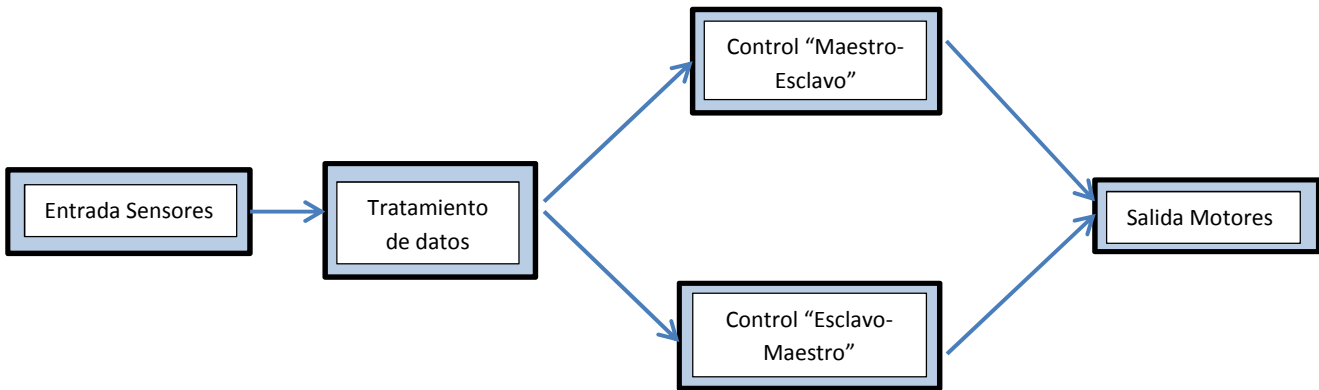
De este modo en la maqueta tenemos “Host Serial Rx” y “Host Serial Tx”; los cuales son los bloques de entrada y salida respectivamente; y entre estos dos bloques está el resto del sistema.

### 5.3.1 Estructura del programa

Dada las características de lo requerido se ha estructurado la programación en tres bloques.

- En un primer bloque se procede a la toma de datos y al tratamiento de esos datos.
- En un segundo bloque se controla la comunicación “Maestro-Esclavo”.
- En el tercer bloque se controla la comunicación “Esclavo-Maestro”.

Así el esquema del programa quedaría:



### 5.3.2 Captura y Tratamiento

En este bloque se muestrean los datos de los sensores y se transforman a grados para facilitar el trabajo con ellos. Para ello se empleó una máscara desarrollada por Dr. Antonio Flores Caballero. Dicha máscara proporciona dos tipos de datos, pero en este proyecto nos quedaremos con los datos de "Display1", que es el que proporciona valores absolutos de posición.

Los datos capturados por el bloque de entrada, son enviados al bloque de tratamiento donde son separados, cada sensor por una línea. En este punto llegamos a los bloques de conversión:

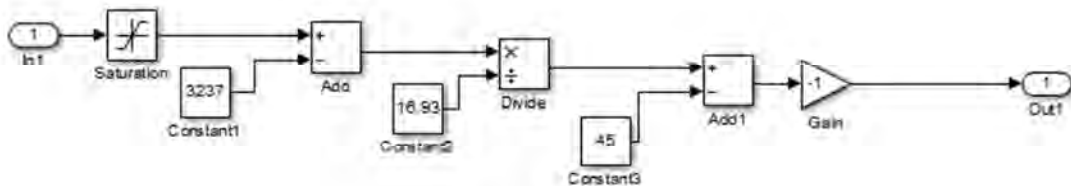


Ilustración 63: Conversión a grados 1

Es este bloque se pasa de pulsos a grados. Para ello se acota superior e inferiormente la entrada con un bloque de saturación. Para determinar estos datos será necesario realizar una calibración. Esto evitará tener valores fuera de rango en caso de que altere algún elemento, como por ejemplo, la posición de los sensores sin querer. De esta forma si entra algún valor por encima o por debajo de los establecidos en la saturación el control tomara el valor de la cota, ya sea superior o inferior.

A continuación, a la señal de entrada se le resta el "Lower limit". De esta forma obtenemos un "cero" en caso de que la entrada se mínima; y un valor igual a la diferencia entre "Upper Limit" y "Lower Limit" en caso de que la entrada sea máxima.

En el siguiente bloque al resultado del bloque anterior se le divide por una constante que es calcula de acuerdo a la fórmula:

$$Const = \frac{Upper\ limit - Lower\ limit}{45}$$

Tras esto se obtiene una salida transformada en grados con la que se puede trabajar con mayor comodidad. Los dos últimos bloques de “Conversión a grados 1” son bloques de ajuste que no están presentes en “Conversión a grados 2”.

Por último se comparan los valores de los dos sensores y se genera una señal de salida con la diferencia entre las dos señales, restando la señal del sensor “Rojo” a la señal del sensor “Verde”. Esta señal será usada por los dos próximos bloques que controlarán la comunicación entre los dos elementos del sistema en ambas direcciones.

### 5.3.3 Maestro-Eslavo

Este bloque es el encargado de asegurar el correcto seguimiento de mando esclavo cuando el mando maestro sea manipulado por el usuario. Este bloque genera una señal de control que gobernará el motor del mando esclavo cuando se genere una diferencia entre las posiciones de los dos mandos.

El funcionamiento de este se basa en un regulador PID que realimentado. Este regulador, genera una salida de Modulación en función de la entrada de datos. Si esa entrada es mayor que una constante, la salida del regulador se habilita y se le aplica tensión al motor. Dicha tensión está regulada por la señal de salida.

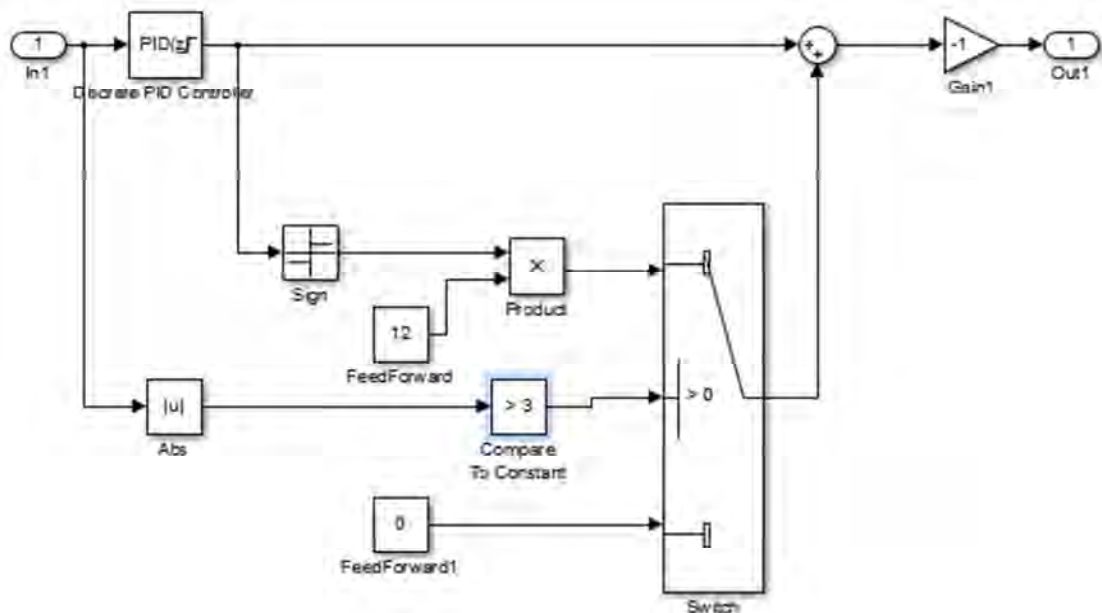


Ilustración 64: Controlador

Para finalizar este bloque encontramos un bloque llamado “Rate Transition” que maneja transferencia de datos entre los puertos que operan a diferentes muestreos.

### 5.3.4 Esclavo-Maestro

Este bloque es el que nos da la señal que gobierna el mando “Maestro” en caso de que el “Esclavo” encuentre un obstáculo.

Para la implementación de este control se ha empleado el mismo control que en el bloque “Maestro-Esclavo” al que se le ha añadido una condición a la entrada. Esta condición activará la salida del motor del “Maestro” cuando la diferencia entre el valor de los motores supere un valor determinado. En caso contrario la señal será “0” y el regulador no generará ninguna salida.

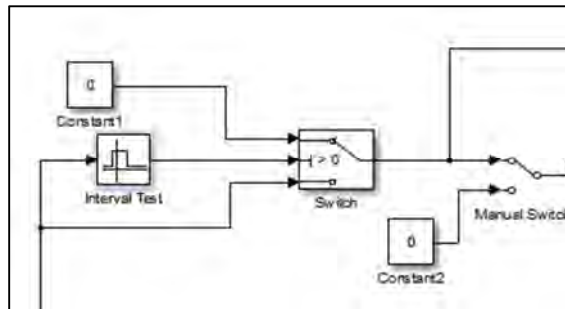


Ilustración 65: Condición de paso

## 5.4 Calibración y puesta a punto

A continuación se comentará el proceso necesario para poner en marcha el sistema. Antes poder operar con nuestra maqueta lo que se debe hacer es reequilibrar la medida de los sensores. Este es un proceso que resulta tedioso y rutinario, pero garantiza un correcto funcionamiento.

Para realizar esta labor, lo que primero que se debe hacer es conectar la maqueta a un ordenador sin conectar la fase de potencia. Tras esto se iniciará el programa y se seleccionará el puerto COM asociado a la placa. Para ello debemos ir a “Equipo” y en “Propiedades” tenemos que pulsar en “Administrador de dispositivos”. En la siguiente pantalla localizaremos “Puertos (COM y LPT)”. En este submenú deberá aparecer nuestra maqueta y tras su nombre el número de puerto asociado. Cuando se conozca el puerto asociado se debe ir al programa en Simulink y clicar sobre “Host Serial Rx”. En este momento se abrirá un cuadro de diálogo en que se podrá seleccionar el puerto. Tras realizar esta labor, el ordenador se comunicara correctamente con la placa de control.



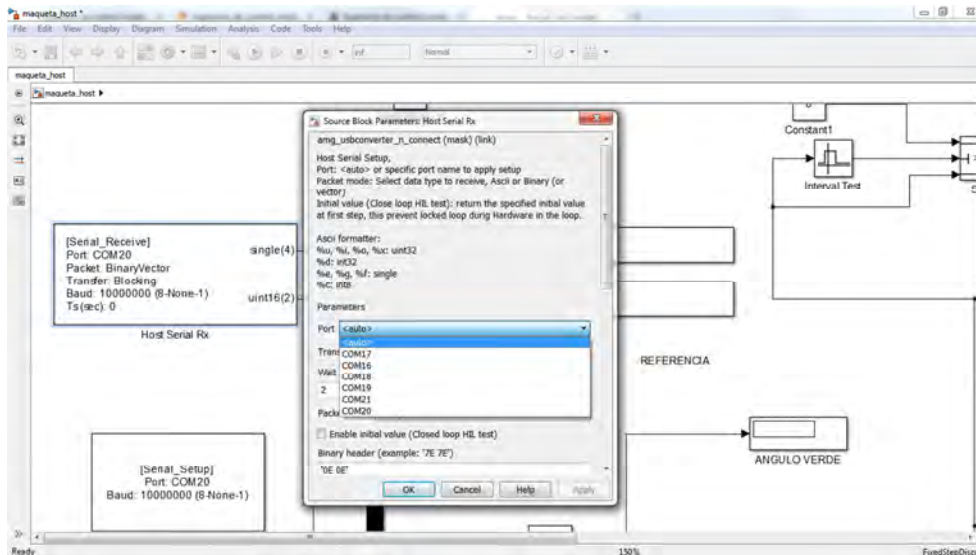


Ilustración 66: Selección Puerto COM

En el siguiente paso habrá que calibrar los sensores. Se realizarán mediciones de los sensores llevando los mandos hasta los extremos. Con esto se podrá calcular la medida de los sensores en cada extremo. Tras realizar esta medida un par de veces por cada mando y extremo se realizará un cambio de las mismas dentro del programa para adecuar el software a nuestro hardware. Estas medidas serán introducidas en una tabla de Excel que calcula la diferencia entre las dos medidas y generar una constante con la que se pasa de “pulsos” a “grados”. Esta contante también debe ser introducida en las variables de la maqueta.

Para cambiar las medidas hay que ir a los módulos de “Conversión a grados 1” y “Conversión a grados 2” y dentro de cada uno de estos se debe acceder a “Saturación”. En la ventana desplegada es donde se deberá cambiar los valores obtenidos tras la calibración. Además de los valores de “Saturación”, también hay que modificar “Constant1” y “Constant2”.

- Constant1: En ambos casos se deberá reemplazar con el nuevo “Lower Limit” obtenido tras la calibración.
- Constant2: Se reemplazará por la contante calculada en la hoja de Excel según la fórmula:

$$Const = \frac{Upper\ limit - Lower\ limit}{45}$$

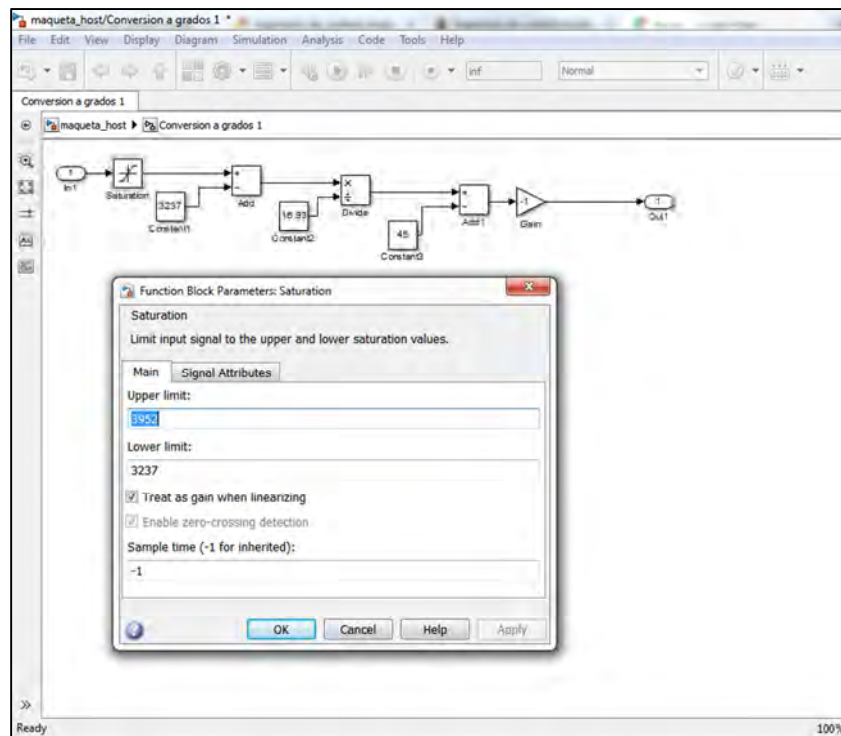


Ilustración 67: Calibración Sensores

Tras esto, se podrá conectar la fase de potencia y se podrá emplear la maqueta sin ningún problema.

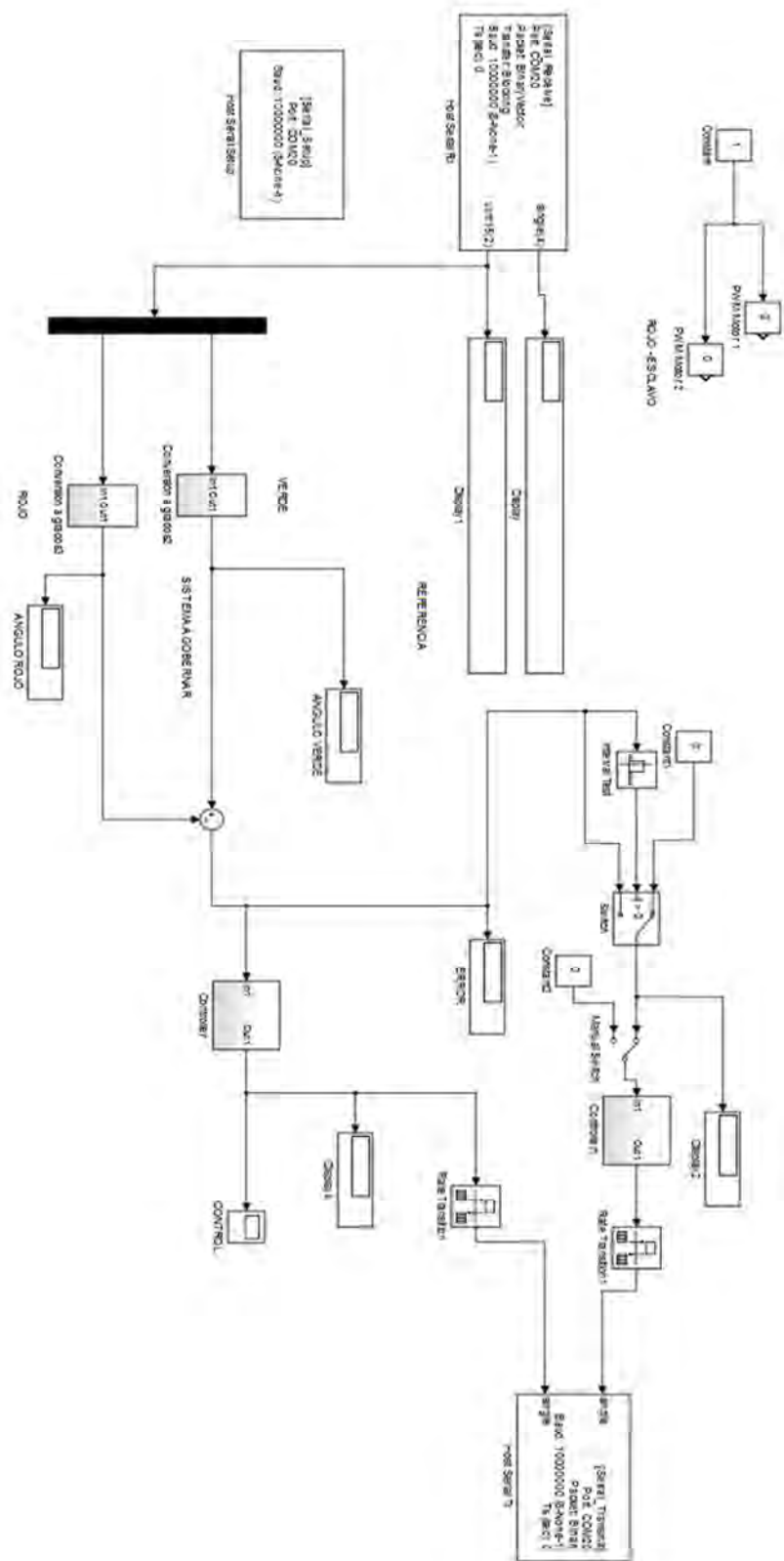


Ilustración 68: Programa

# **6 Conclusión y trabajos futuros**

## **6.1 Conclusiones**

Tras la realización de este Trabajo de Fin de Grado se puede afirmar que se ha logrado el objetivo que se marcó al inicio: diseñar, construir y controlar un sistema háptico que nos permitiera comprender como funcionan las interfaces hápticas. Este sistema es capaz de procesar información en dos direcciones, lo que le permite al usuario interactuar, en un mayor grado, con el dispositivo.

Tras la finalización se puede concluir que se han alcanzado los siguientes objetivos:

- Diseño y construcción del sistema partiendo desde “cero” empleando para ello materiales “domésticos”.
- Dotación de la electrónica necesaria que nos proporciona la información que necesitamos para realizar el control.
- Empleo de la placa de control STM32F4, lo que ha derivado en su conocimiento así como el de su entorno de trabajo.
- Programación del sistema siendo capaz de controlar los motores mediando una programación en “Simulink” empleando un controlador PID.
- Realización de los cambios que se consideraron oportunos tras la realización de las pruebas realizadas en cada fase del proyecto.

No obstante, durante la elaboración del proyecto también han surgido problemas y dificultades.

- Se establecieron unas pautas para la realización del proyecto y realizó un estudio previo de cómo resolver el problema, ya que los conocimientos sobre la materia eran casi inexistentes.
- Al haber empleado materiales “poco ortodoxos” el sistema resultante no posee una gran precisión. Esto se observa cuando en el camino del esclavo se interpone un obstáculo y se obtiene una respuesta en el maestro que no es todo lo clara que se puede esperar.
- Antes del uso, es necesario realizar una calibración en los sensores para obtener una respuesta aceptable.

## **6.2 Trabajos futuros**

Tras los resultados obtenidos en este proyecto se proponen como futuros trabajos:

- Rediseñar la maqueta para desarrollar una aplicación más pequeña y compacta; empleando otro método de transmisión y estableciendo otra disposición; utilizando otros materiales que nos aporten mayor precisión en los resultados.
- Realizar un estudio sobre la viabilidad de aplicar la tecnología estudiada a un sistema real como pudiera ser el control teleoperado de un brazo robótico.

## 7 Costes del Proyecto

En este capítulo se describe de forma aproximada los costes en los que ha incurrido el proyecto. Para ello se han desglosado las tareas ordenando cada una de ellas en función del tiempo.

En la siguiente tabla se observa el tiempo que ha llevado cada una de las fases del proyecto.

Fases	Tiempo (h)
Estudio del problema a resolver	15
Diseño del Sistema Físico	45
Impresión Piezas	6
Montaje del sistema mecánico	15
Montaje del sistema electrónico	5
Programación del sistema	10
Pruebas y Calibración	40
Mejora	25
Memoria del proyecto (realizada en paralelo)	150
<b>Total</b>	<b>316</b>

Tabla 11: Tiempo empleado en el proyecto

Suponiendo un coste de 25 €/hora trabajada, el precio total de las horas empleadas es de:

Concepto	Cantidad	Coste/und (€)	Total
Horas de Trabajo	316	25	7900 €

A continuación se calcula el coste de los elementos físicos y electrónicos empleados:

Concepto	Cantidad	Coste/und(€)	Total
Escuadras de ángulo 50x50 mm	8	0,25	2 €
Tornillos M4 x 20mm-DIN 965	34	0,05	1,7 €
Tuercas M4-DIN 439	38	0,05	1,9 €
Varilla M6 x 1000	1	2	2 €
Tuercas Autoblocantes M6mm-DIN 985	12	0,10	1,2 €
Arandela 6mm-DIN 440-R	16	0,05	0,8 €
Tuerca Mariposa M6mm-DIN 315	4	0,15	0,60 €
Tornillo M5 x 70- DIN 963	4	0,10	0,40 €
Arandela M5mm-DIN 440-R	16	0,05	0,8 €
Tuerca mariposa M5mm-DIN315	4	0,10	0,40 €
Muelle M6 x 100	4	0,15	0,60 €
Tornillo M3 x 20- DIN 963	4	0,10	0,40 €

Tuerca M3mm-DIN 439	4	0,05	0,20 €
Arandela M4 mm-DIN 440-R	32	0,05	1,6 €
Tablero de Contrachapado	4	2	8 €
Tablero de Pino para Base	1	5	5 €
Impresión Piezas 3D	178 gr	0,03 €/gr	6 €
<b>Total Físico</b>	33,6 €		
Motores CC Reductora 23-1 3v-12	2	7	14 €
L298N	1	3	3€
STM32F4 Discovery	1	24,16	24,16 €
AS5045B	2	0	0 €
<b>Total Electrónico</b>	41,16 €		
<b>Total</b>	74,76 €		

Tabla 12: Coste Materiales

En la siguiente tabla se detalla el coste de los programas y equipos empleados:

Concepto	Coste Total (€)	Utilización (%)	Dedicación (meses)	Periodo de depreciación (meses)	Coste
Ordenador	530	100	4	66	48,50 €
Prusa I3	499	100	0,1	12	4,15 €
MatLab R2011a	6.000	90	2	54	200 €
Microsoft Office 2010	550	100	3	42	39,28 €
Solid Edge V19	600	100	2	60	19,99 €
Cura 15.04	0	100	1	-	0 €
<b>Total</b>	311,92 €				

Tabla 13: Costes de equipo y software

Para el cálculo de estos costes se ha tenido en cuenta la siguiente expresión:

$$C = \frac{t}{T} * P * D$$

donde:

$t$  es el número de meses de uso.

$T$  es el periodo de depreciación.

$P$  es el precio de compra de software o equipo.

$D$  es la utilización del equipo o software en porcentaje.

Así el coste total, añadiendo los costes indirectos (15% del total) asciende a un total de:

<b>Concepto</b>	<b>Coste (€)</b>
Horas de trabajo	7900
Coste Matreiales	74,76
Costes de equipo y software	311,92
Total (sin Indirectos)	8918,68
Costes Indirectos	1243
<b>Total</b>	<b>9529,70</b>



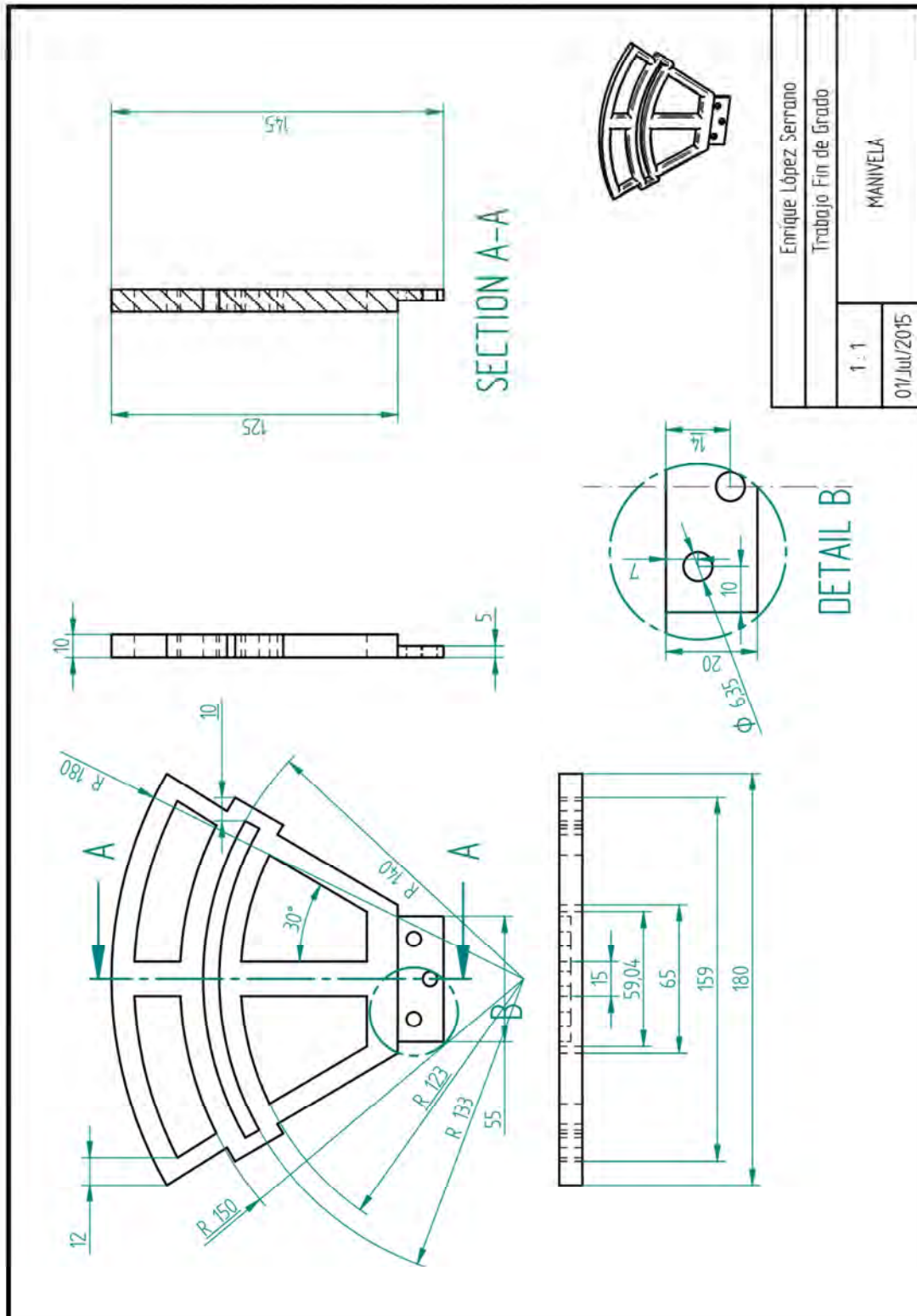
## 8 Bibliografía

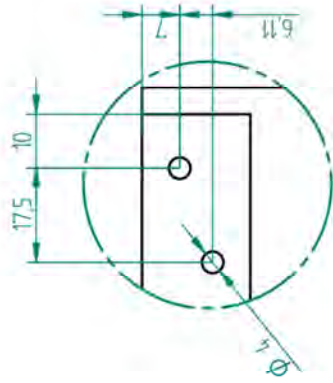
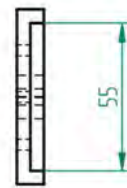
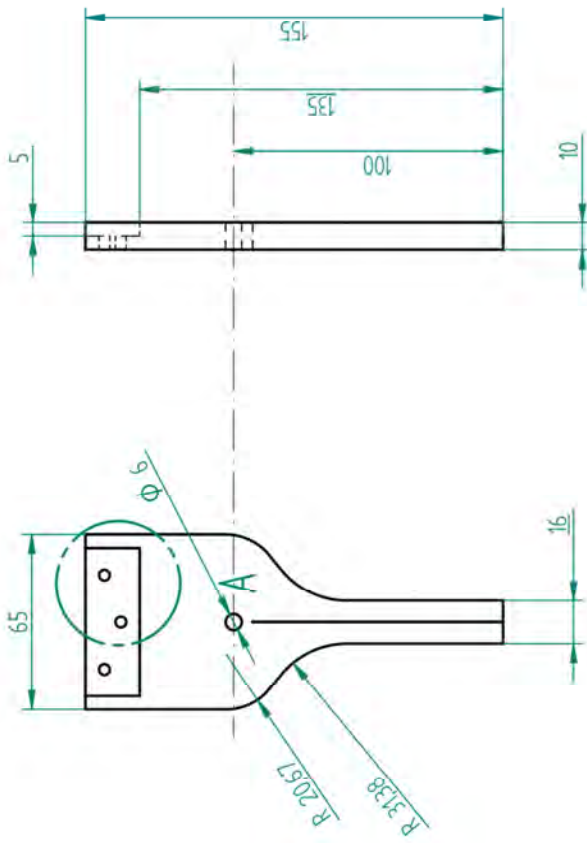
- [1] Antonio Flores Caballero, Antonio. “*SISTEMA AVANZADO DE PROTOTIPADO RAPIDO PARA CONTROL EN EXOESQUELETOS Y DISPOSITIVOS MECATRONICOS.*” Tesis doctoral. Universidad Carlos III de Madrid. 2014
- [2] Vélez Escorial, Andrés. “*Diseño mecánico de un interfaz Háptico para Realidad Virtual*”, Proyecto Fin de Carrera, 2011
- [3] MASSIE T.H., SALISBURY J.K. “*THE PHANTOM HAPTIC INTERFACE: A DEVICE FOR PROBING VIRTUAL OBJECTS*” Proceedings of the ASME Winter Annual Meeting, symposium on Haptic interfaces for virtual environments and teleoperator systems. 1994.
- [4] Ptonline.com, (2015). *3D Printers Lead Growth of Rapid Prototyping*. [online] Available at: <http://www.ptonline.com/articles/3d-printers-lead-growth-of-rapid-prototyping> [Accessed 3 May.2015].
- [5] Reprap.org, (2015). *Prusa i3 Hephestos/es - RepRapWiki*. [online] Available at: [http://reprap.org/wiki/Prusa\\_i3\\_Hephestos/es](http://reprap.org/wiki/Prusa_i3_Hephestos/es) [Accessed 6 Jul. 2015].
- [6] K. WEISSERMEL; H. ARPE, “*QUIMICA ORGANICA INDUSTRIAL*”, REVERTE, 1981. ISBN 9788429179897
- [7] K. WEISSERMEL; H. ARPE, “*QUIMICA ORGANICA INDUSTRIAL*”, REVERTE, 1981. ISBN 9788429179897
- [8] Es.wikipedia.org, (2015). *Acrilonitrilo butadieno estireno*. [online] Available at: [https://es.wikipedia.org/wiki/Acrilonitrilo\\_butadieno\\_estireno](https://es.wikipedia.org/wiki/Acrilonitrilo_butadieno_estireno) [Accessed 8 Sep. 2015].
- [9] Es.wikipedia.org, (2015). *Poliácido láctico*. [online] Available at: [https://es.wikipedia.org/wiki/Poli%C3%A1cido\\_l%C3%A1ctico](https://es.wikipedia.org/wiki/Poli%C3%A1cido_l%C3%A1ctico) [Accessed 8 Sep. 2015].
- [10] Diosdado, R. (2015). *Zona Maker - Manual de CURA*. [online] Zonamaker.com. Available at: <http://www.zonamaker.com/impresion-3d/software-imp3d/manual-de-cura> [Accessed 8 May. 2015].
- [11] Es.wikipedia.org, (2015). *Puente H (electrónica)*. [online] Available at: [https://es.wikipedia.org/wiki/Puente\\_H\\_%28electr%C3%B3nica%29](https://es.wikipedia.org/wiki/Puente_H_%28electr%C3%B3nica%29) [Accessed 10 Aug. 2015].
- [12] L298N, D. (2015). *Driver alta potencia - L298N*. [online] Bricogeek.com. Available at: <http://www.bricogeek.com/shop/182-driver-alta-potencia-l298n.html> [Accessed 13 Jul. 2015].
- [13] Espinosa Polo, Rubén. “*Mini-Moto controlada por computador: Arquitectura Hardware*” Trabajo de fin de Carrera. Universidad Carlos III de Madrid. 2010.
- [14] St.com, (2015). *STMicroelectronics*. [online] Available at: <http://www.st.com/web/en/home.html> [Accessed 8 May. 2015].

- [15] Blum, Jeremy. *“Arduino a Fondo”* Anaya Multimedia, 2014. ISBN 978-84-415-3652-4
- [16] Asifunciona.com, (2015). *FUNCIONA EL MOTOR DE CORRIENTE DIRECTA (C.D.) O CONTINUA (C.C.)*. [online] Available at: [http://www.asifunciona.com/electrotecnia/af\\_motor\\_cd/af\\_motor\\_cd\\_3.htm](http://www.asifunciona.com/electrotecnia/af_motor_cd/af_motor_cd_3.htm) [Accessed 10 Jun. 2015].
- [17] Asifunciona.com, (2015). *FUNCIONA EL MOTOR DE CORRIENTE DIRECTA (C.D.) O CONTINUA (C.C.)*. [online] Available at: [http://www.asifunciona.com/electrotecnia/af\\_motor\\_cd/af\\_motor\\_cd\\_4.htm](http://www.asifunciona.com/electrotecnia/af_motor_cd/af_motor_cd_4.htm) [Accessed 10 Jun. 2015].
- [18] Lcrc.anl.gov, (2015). *Matlab | Laboratory Computing Resource Center*. [online] Available at: <http://www.lcrc.anl.gov/installed-software/MATLAB> [Accessed 8 Sep. 2015].
- [19] Karl J. Aström, *“CONTROL PID AVANZADO”*, Prentice Hall, 1997. ISBN 9788483225110
- [20] Karl J. Aström and Björn Wittenmark. *“COMPUTER CONTROLLED SYSTEMS, THEORY AND DESIGN.”* Prentice Hall, 1997. ISBN 0133148998
- [21] Graham C. Goodwin, Stefan F. Graebe, and Mario E. Salgado. *“CONTROL SYSTEM DESIGN”* Prentice Hall International, 2001. ISBN 0139586539
- [22] Aracil Javier, *“Introducción a Matlab y Simulink”*; Trabajo Fin de Carrera. Universidad de Sevilla, 2006.
- [23] Núñez Yáñez, Pablo. *“MINI-MOTO CONTROLADA POR COMPUTADOR: ARQUITECTURA SOFTWARE”* Trabajo de fin de Carrera. Universidad Carlos III de Madrid. 2010.
- [24] FUNDAMENTOS INFORMATICOS, (2008). *INTELIGENCIA ARTIFICIAL*. [online] Available at: <https://fundamentosinformaticosjl.wordpress.com/2008/01/27/las-computadoras/> [Accessed 5 May. 2015].

# 9 Anexos

## 9.1 Planos





DETALLE A



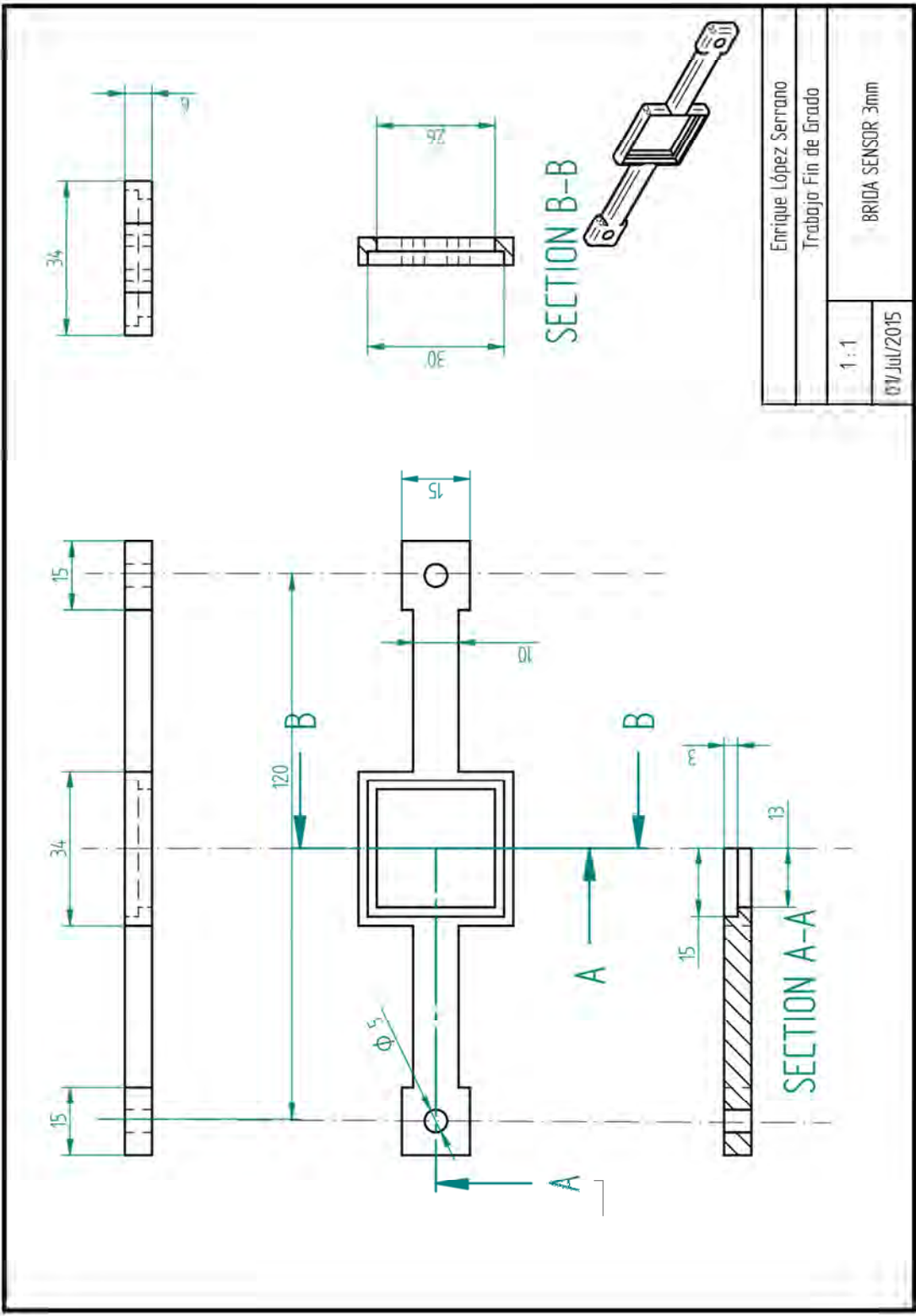
Enrique López Serrano

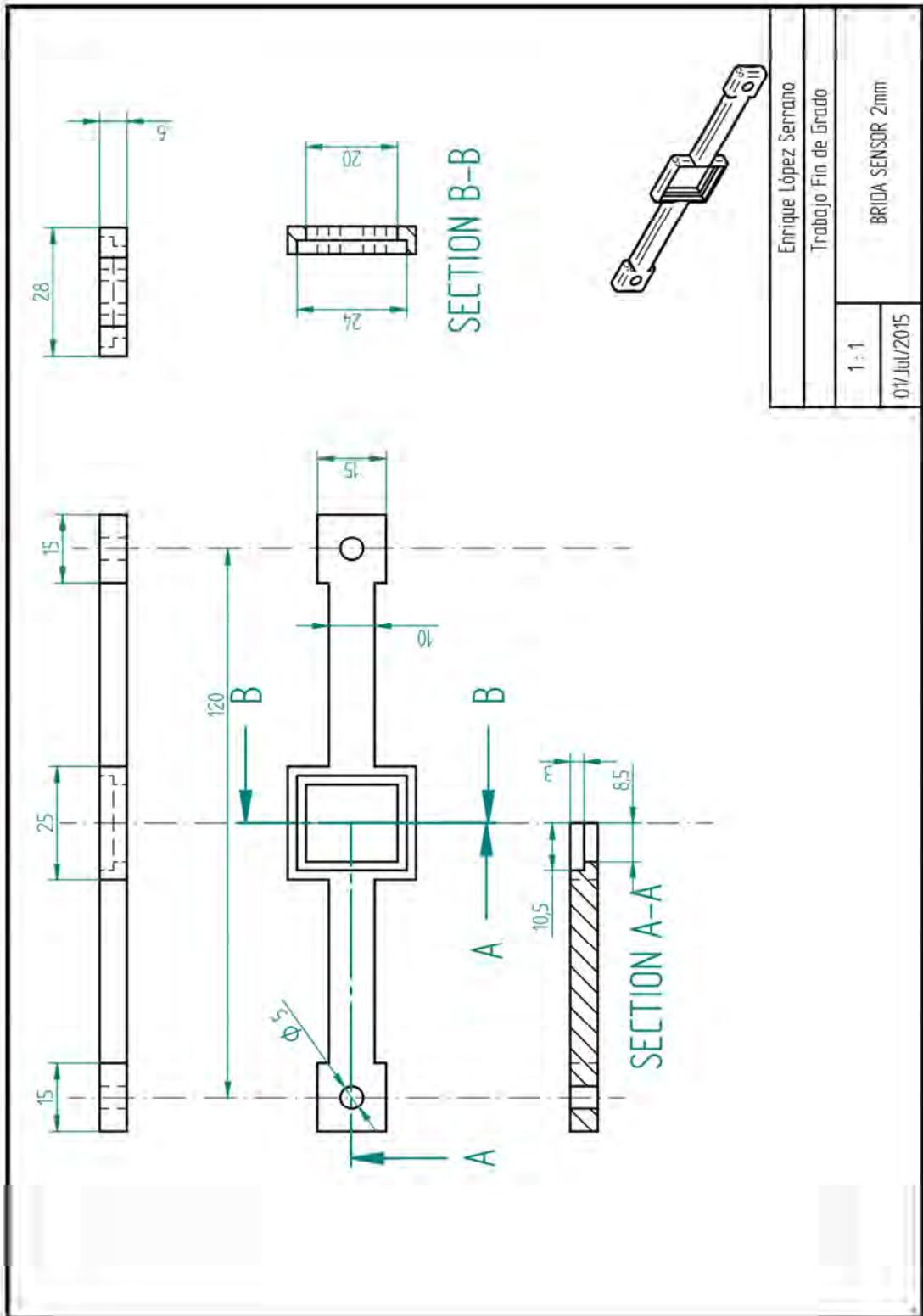
Trabajo Fin de Grado

1:2

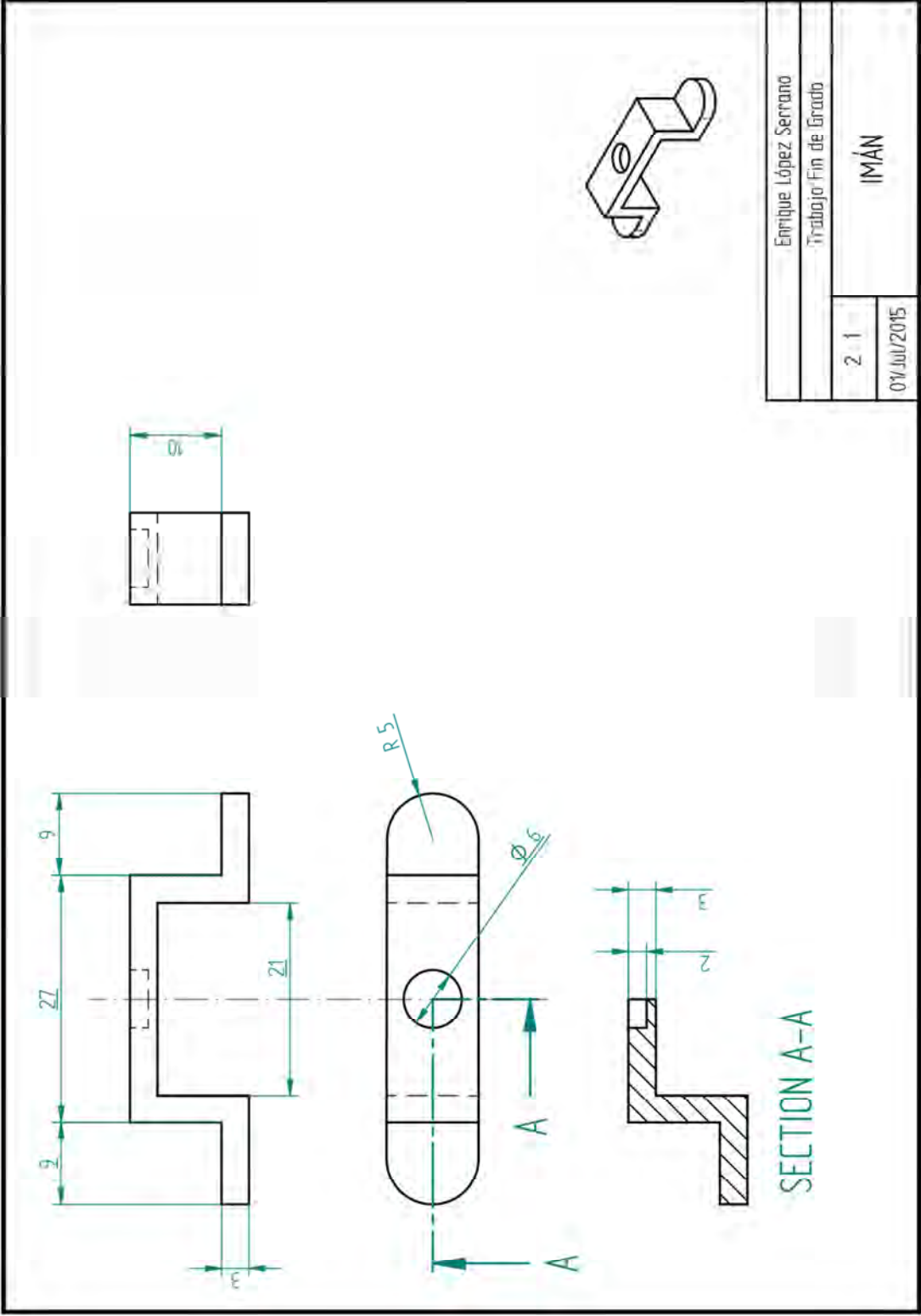
01/JUL/2015

MANGO





Enrique López Serrano	
Trabajo Fin de Grado	
1:1	BRIDA SENSOR 2mm
01/JUL/2015	



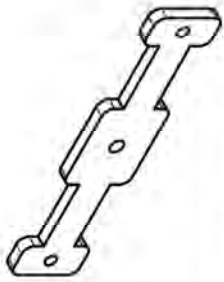
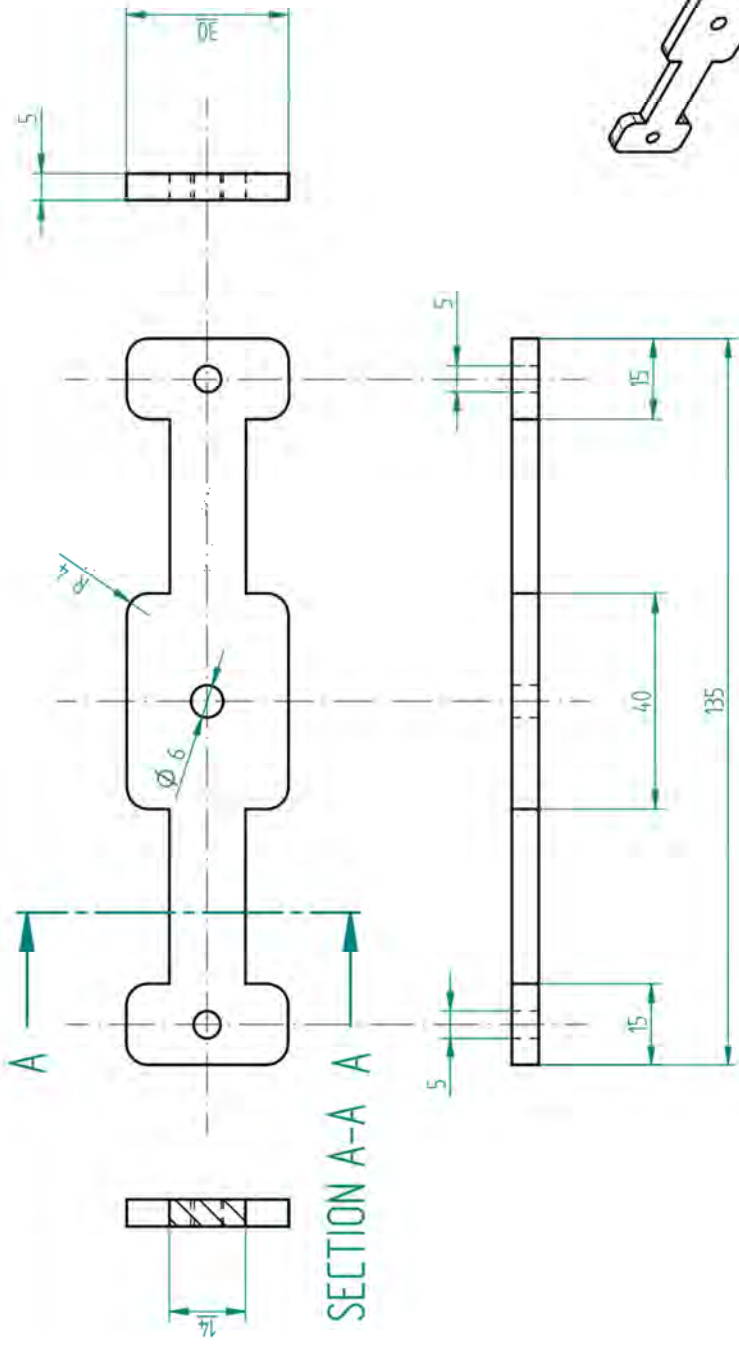
Enrique López Serrano

Trabajo Fin de Grado

2.1

01/JUL/2015

IMAN



Enrique López Serrano

Trabajo Fin de Grado

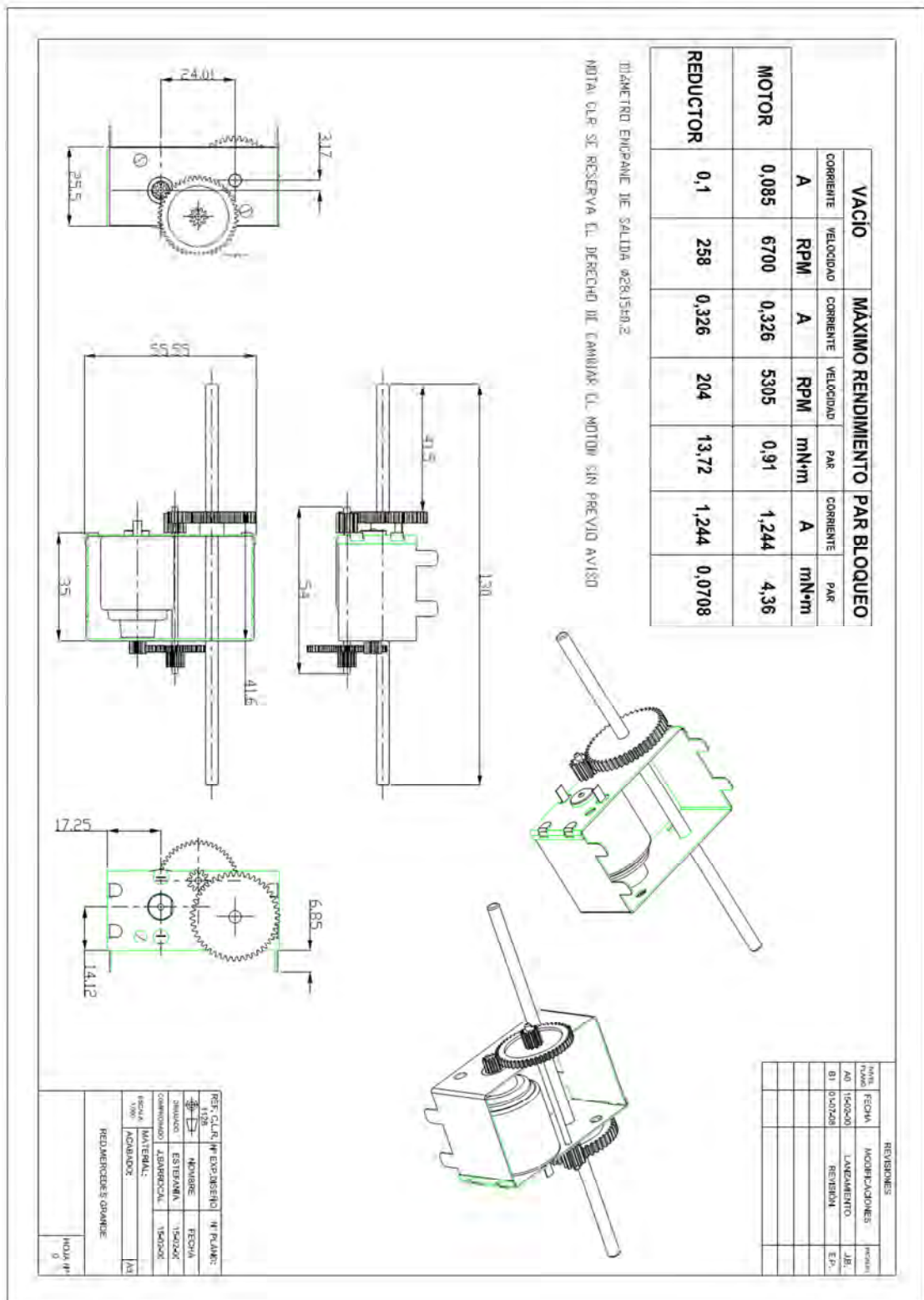
1 : 1

01/JUL/2015

BRIDA DE AJUSTE



## 9.2 Características Motores



## 9.3 STM32F4



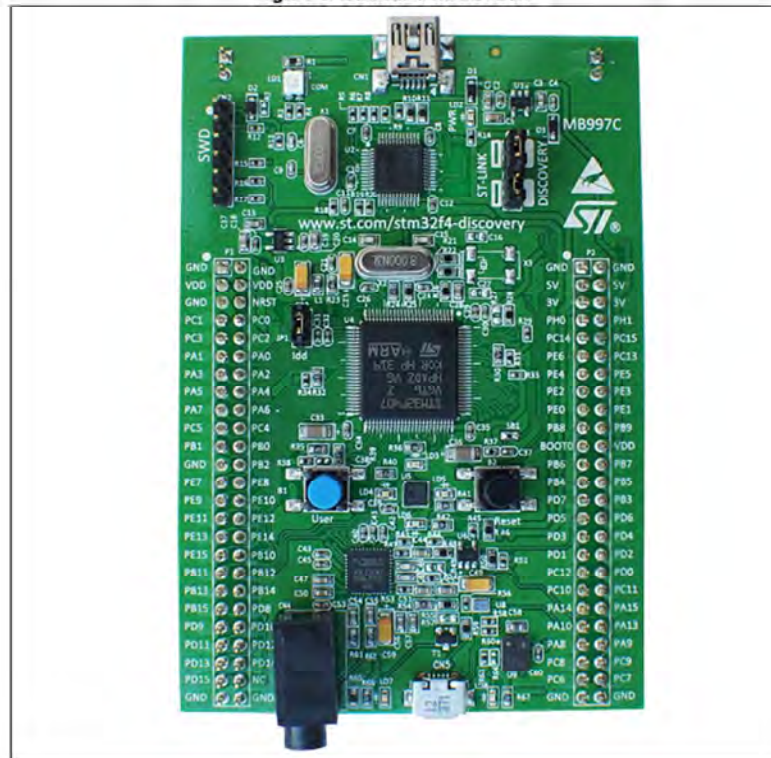
# UM1472 User manual

## Discovery kit for STM32F407/417 lines

### Introduction

The STM32F4DISCOVERY helps you to discover the STM32F407 & STM32F417 lines' high-performance features and to develop your applications. It is based on an STM32F407VGT6 and includes an ST-LINK/V2 embedded debug tool interface, ST MEMS digital accelerometer, ST MEMS digital microphone, audio DAC with integrated class D speaker driver, LEDs, pushbuttons and a USB OTG micro-AB connector.

Figure 1. STM32F4DISCOVERY



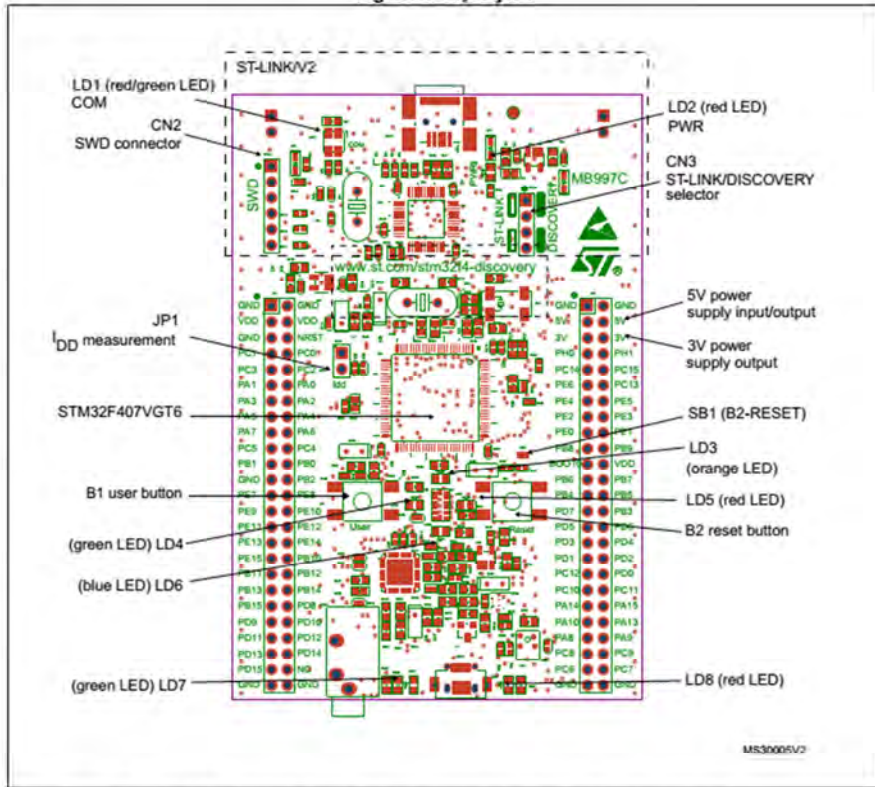
1. Picture not contractual

### 3 Features

The STM32F4DISCOVERY offers the following features:

- STM32F407VGT6 microcontroller featuring 1 MB of Flash memory, 192 KB of RAM in an LQFP100 package
- On-board ST-LINK/V2 with selection mode switch to use the kit as a standalone ST-LINK/V2 (with SWD connector for programming and debugging)
- Board power supply, through USB bus or from an external 5V supply voltage
- External application power supply: 3V and 5V
- LIS302DL or LIS3DSH, ST MEMS motion sensor, 3-axis digital output accelerometer
- MP45DT02, ST MEMS audio sensor, omnidirectional digital microphone
- CS43L22, audio DAC with integrated class D speaker driver
- Eight LEDs:
  - LD1 (red/green) for USB communication
  - LD2 (red) for 3.3V power on
  - Four user LEDs, LD3 (orange), LD4 (green), LD5 (red) and LD6 (blue)
  - 2 USB OTG LEDs LD7 (green) VBus and LD8 (red) over-current
- Two pushbuttons (user and reset)
- USB OTG with micro-AB connector
- Extension header for LQFP100 I/Os for quick connection to prototyping board and easy probing

Figure 3. Top layout

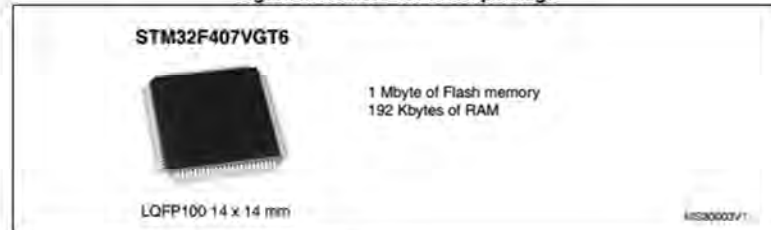


Note: Pin 1 of CN2, CN3, JP1, P1 and P2 connectors are identified by a square.

#### 4.1 STM32F407VGT6 microcontroller

This ARM Cortex-M4 32-bit MCU with FPU has 210 DMIPS, up to 1 MB Flash/192+4 KB RAM, USB OTG HS/FS, Ethernet, 17 TIMs, 3 ADCs, 15 comm. interfaces and a camera.

Figure 5. STM32F407VGT6 package

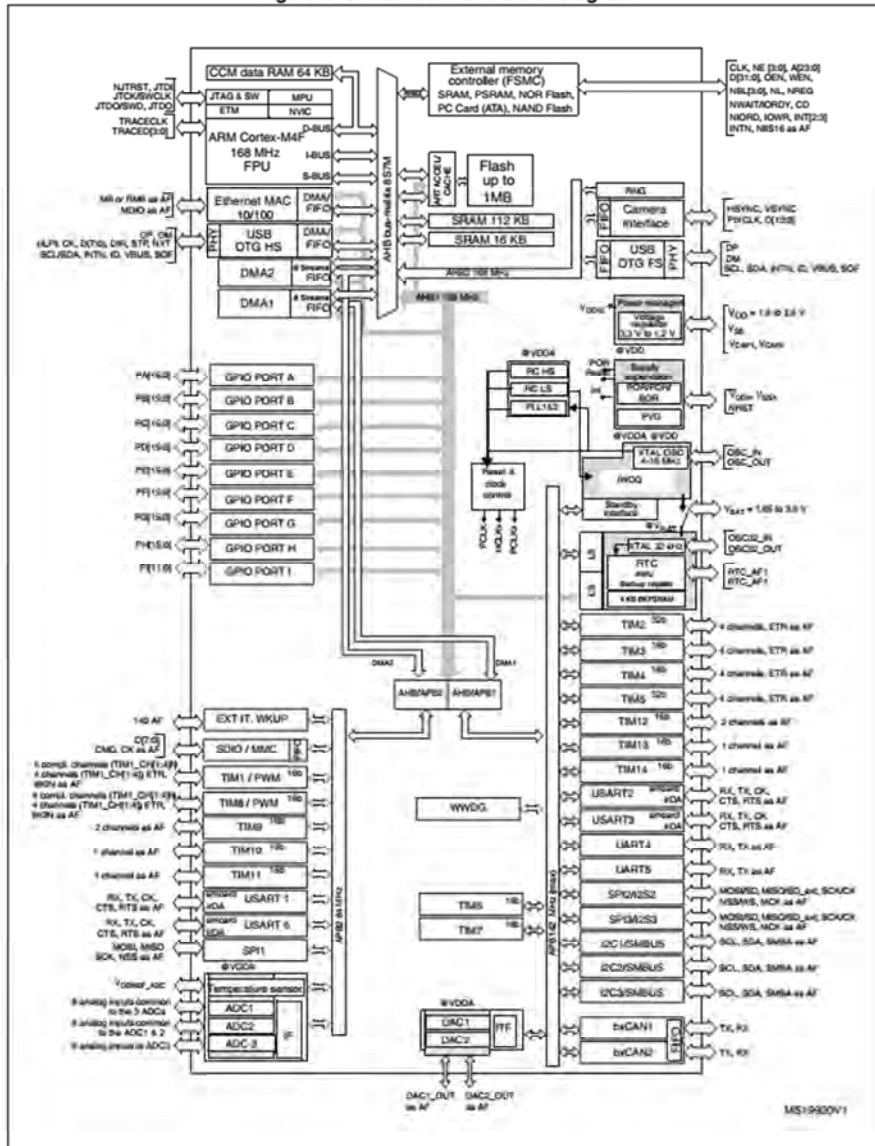


This device provides the following benefits.

- 168 MHz/210 DMIPS Cortex-M4 with single cycle DSP MAC and floating point unit providing:
  - Boosted execution of control algorithms
  - More features possible for your applications
  - Ease of use
  - Better code efficiency
  - Faster time to market
  - Elimination of scaling and saturation
  - Easier support for meta-language tools
- Designed for high performance and ultra fast data transfers; ART Accelerator, 32-bit, 7-layer AHB bus matrix with 7 masters and 8 slaves including 2 blocks of SRAM, Multi DMA controllers: 2 general purpose, 1 for USB HS, 1 for Ethernet, One SRAM block dedicated to the core, providing performance equivalent to 0-wait execution from Flash. Concurrent execution and data transfers and simplified resource allocation
- Outstanding power efficiency; Ultra-low dynamic power, RTC <1  $\mu$ A typical in VBAT mode, 3.6 V down to 1.7 V VDD, Voltage regulator with power scaling capability, providing extra flexibility to reduce power consumption for applications requiring both high processing and low power performance when running at low voltage or on a rechargeable battery
- Maximum integration: Up to 1 Mbyte of on-chip Flash memory, 192 Kbytes of SRAM, reset circuit, internal RCs, PLLs, WLCSP package available, providing more features in space constrained applications
- Superior and innovative peripherals providing new possibilities to connect and communicate high speed data and more precision due to high resolution
- Extensive tools and software solutions providing a wide choice within the STM32 ecosystem to develop your applications.



Figure 6. STM32F407VGT6 block diagram



## 4.12 Extension connectors

The male headers P1 and P2 can connect the STM32F4DISCOVERY to a standard prototyping/wrapping board. STM32F407VGT6 GPIOs are available on these connectors. P1 and P2 can also be probed by an oscilloscope, logical analyzer or voltmeter.

Table 5. MCU pin description versus board function

Main function	MCU pin	Board function															
		Alternate functions	LQFP100	CS43L22	MP45DT02	LIS302DL or LIS31ESH	Pushbutton	LED	SWD	USB	OSC	Free I/O	Power supply	CNS	CN2	P1	P2
BOOT0	VPP		94	-	-	-	-	-	-	-	-	-	-	-	-	-	21
NRST	-		14	-	-	-	RESET	-	NRST	-	-	-	-	-	5	6	-
PA0-WKUP	USART2_CTS/ USART4_TX/ ETH_MII_CRX/ TIM2_CH1_ETR/ TIM5_CH1/ TIM8_ETR/ ADC123_IN0/ WKUP		23	-	-	-	USER	-	-	-	-	-	-	-	-	12	-
PA1	USART2_RTS/ USART4_RX/ ETH_RMII_REF_CLK/ ETH_MII_RX_CLK/ TIM5_CH2/ TIMM2_CH2/ ADC123_IN1		24	-	-	-	-	-	-	-	-	-	-	-	-	11	-
PA2	USART2_TX/ TIM5_CH3/ TIM9_CH1/ TIM2_CH3/ ETH_MDIO/ ADC123_IN2		25	-	-	-	-	-	-	-	-	-	-	-	-	14	-

Table 5. MCU pin description versus board function (continued)

MCU pin		Board function															
Main function	Alternate functions	LQFP100	CS43L22	MP450T02	LIS302DL or LIS3DSH	Pushbutton	LED	SWD	USB	OSC	Free I/O	Power supply	CNS	CN2	P1	P2	
PA3	USART2_RX/ TIM5_CH4/ TIM9_CH2/ TIM2_CH4/ OTG_HS_ULPI_D0/ ETH_MII_COL/ ADC123_IN3	26	-	-	-	-	-	-	-	-	-	-	-	-	-	13	-
PA4	SPI1_NSS/ SPI3_NSS/ USART2_CK/ DCMI_HSYNC/ OTG_HS_SOF/ I2S3_WS/ ADC12_IN4/ DAC1_OUT	29	LRCQ/ANTx	-	-	-	-	-	-	-	-	-	-	-	-	16	-
PA5	SPI1_SCK/ OTG_HS_ULPI_CK/ TIM2_CH1_ETR/ TIM8_CHIN/ ADC12_IN5/ DAC2_OUT	30	-	-	SCU/SPC	-	-	-	-	-	-	-	-	-	-	15	-
PA6	SPI1_MISO/ TIM8_BKIN/ TIM13_CH1/ DCMI_PIXCLK/ TIM3_CH1/ TIM1_BKIN/ ADC12_IN6	31	-	-	SDO	-	-	-	-	-	-	-	-	-	-	18	-
PA7	SPI1_MOSI/ TIM8_CH1N/ TIM14_CH1/TIM3_CH2/ ETH_MII_RX_DV/ TIM1_CH1N/ RMII_CRS_DV/ ADC12_IN7	32	-	-	SDA/SDI/SDO	-	-	-	-	-	-	-	-	-	-	17	-



Table 5. MCU pin description versus board function (continued)

MCU pin		Board function															
Main function	Alternate functions	LQFP100	CS43L22	MP450T02	LIS302DL or LIS3DSH	Pushbutton	LED	SWD	USB	OSC	Free I/O	Power supply	CNS	CN2	P1	P2	
PA8	MCO1/ USART1_CK/ TIM1_CH1/ I2C3_SCL/ OTG_FS_SOF	67	-	-	-	-	-	-	-	-	-	-	-	-	-	-	43
PA9	USART1_TX/ TIM1_CH2/ I2C3_SMBA/ DCMI_D0/ OTG_FS_VBUS	68	-	-	-	-	GREEN	-	VBUS	-	-	-	1	-	-	-	44
PA10	USART1_RX/ TIM1_CH3/ OTG_FS_ID/ DCMI_D1	69	-	-	-	-	-	-	ID	-	-	-	4	-	-	-	41
PA11	USART1_CTS/ CAN1_RX/ TIM1_CH4/ OTG_FS_DM	70	-	-	-	-	-	-	DM	-	-	-	2	-	-	-	-
PA12	USART1_RTS/ CAN1_TX/ TIM1_ETR/ OTG_FS_DP	71	-	-	-	-	-	-	DP	-	-	-	3	-	-	-	-
PA13	JTMS-SWDIO	72	-	-	-	-	-	SWDIO	-	-	-	-	-	4	-	-	42
PA14	JTCK-SWCLK	76	-	-	-	-	-	SWCLK	-	-	-	-	-	2	-	-	39
PA15	JTDI/ SPI3_NSS/ I2S3_WS/ TIM2_CH1_ETR/ SPI1_NSS	77	-	-	-	-	-	-	-	-	-	-	-	-	-	-	40

Table 5. MCU pin description versus board function (continued)

MCU pin		Board function														
Main function	Alternate functions	LQFP100	CS43L22	MP450T02	LIS302DL or LIS3DSH	Pushbutton	LED	SWD	USB	OSC	Free I/O	Power supply	CNS	CN2	P1	P2
PB0	TIM3_CH3/ TIM8_CH2N/ OTG_HS_ULPI_D1/ ETH_MII_RXD2/ TIM1_CH2N/ ADC12_IN8	35	-	-	-	-	-	-	-	-	-	-	-	-	22	-
PB1	TIM3_CH4/ TIM8_CH3N/ OTG_HS_ULPI_D2/ ETH_MII_RXD3/ OTG_HS_INTN/ TIM1_CH3N/ ADC12_IN9	36	+	+	-	-	+	-	+	-	-	-	-	-	21	-
PB2	-	37	-	-	-	-	-	-	-	-	-	-	-	-	24	-
PB3	JTDO/ TRACESWO/ SPI3_SCK/ I2S3_CK/ TIM2_CH2/ SPI1_SCK	89	-	-	-	-	-	SWO	-	-	-	-	-	6	-	28
PB4	NJTRST/ SPI3_MISO/ TIM3_CH1/ SPI1_MISO/ I2S3ext_SD	90	-	-	-	-	-	-	-	-	-	-	-	-	-	25
PB5	I2C1_SMBA/ CAN2_RX/ OTG_HS_ULPI_D7/ ETH_PPS_OUT/ TIM3_CH2/ SPI1_MOSI/ SPI3_MOSI/ DCMI_D10/ I2S3_SD	91	-	-	-	-	-	-	+	-	-	-	-	-	-	26

Table 5. MCU pin description versus board function (continued)

MCU pin		Board function															
Main function	Alternate functions	LQFP100	CS43L22	MP450T02	LIS302DL or LIS3DSH	Pushbutton	LED	SWD	USB	OSC	Free I/O	Power supply	CNS	CN2	P1	P2	
PB6	I2C1_SCL/ TIM4_CH1/ CAN2_TX/ OTG_FS_INTN/ DCMI_D5/ USART1_TX	92	SCL	-	-	-	-	-	-	-	-	-	-	-	-	-	23
PB7	I2C1_SDA/ FSMC_NU/ DCMI_VSYNC/ USART1_RX/ TIM4_CH2	93	-	-	-	-	-	-	-	-	-	-	-	-	-	-	24
PB8	TIM4_CH3/ SDIO_D4/ TIM10_CH1/ DCMI_D6/ OTG_FS_SCL/ ETH_MII_TXD3/ I2C1_SCL/ CAN1_RX	95	-	-	-	-	-	-	-	-	-	-	-	-	-	-	19
PB9	SPI2_NSS/ I2S2_WS/ TIM4_CH4/ TIM11_CH1/ OTG_FS_SDA/ SDIO_D5/ DCMI_D7/ I2C1_SDA/ CAN1_TX	96	SDA	-	-	-	-	-	-	-	-	-	-	-	-	-	20
PB10	SPI2_SCK/ I2S2_CK/ I2C2_SCL/ USART3_TX/ OTG_HS_ULPI_D3/ ETH_MII_RX_ER/ OTG_HS_SCL/ TIM2_CH3	47	-	CLK	-	-	-	-	-	-	-	-	-	-	-	-	34

Table 5. MCU pin description versus board function (continued)

MCU pin		Board function														
Main function	Alternate functions	LQFP100	CS43L22	MP450702	LIS302DL or LIS3DSH	Pushbutton	LED	SWD	USB	OSC	Free I/O	Power supply	CNS	CN2	P1	P2
PB11	I2C2_SDA/ USART3_RX/ OTG_HS_ULPI_DA/ ETH_RMII_TX_EN/ ETH_MII_TX_EN/ OTG_HS_SDA/ TIM2_CH4	48	-	-	-	-	-	-	-	-	-	-	-	-	35	-
PB12	SPI2_NSS/ I2S2_WS/ I2C2_SMBA/ USART3_CK/ TIM1_BKIN/ CAN2_RX/ OTG_HS_ULPI_D5/ ETH_RMII_TXD0/ ETH_MII_TXD0/ OTG_HS_ID	51	-	-	-	-	-	-	-	-	-	-	-	-	36	-
PB13	SPI2_SCK/ I2S2_CK/ USART3_CTS/ TIM1_CH1N/ CAN2_TX/ OTG_HS_ULPI_D6/ ETH_RMII_TXD1/ ETH_MII_TXD1/ OTG_HS_VBUS	52	-	-	-	-	-	-	-	-	-	-	-	-	37	-
PB14	SPI2_MISO/ TIM1_CH2N/ TIM12_CH1/ OTG_HS_DMUSART3 _RTS/ TIM8_CH2N/ I2S2ext_SD	53	-	-	-	-	-	-	-	-	-	-	-	-	38	-

Table 5. MCU pin description versus board function (continued)

MCU pin		Board function														
Main function	Alternate functions	LQFP100	CS43L22	MP450T02	LIS302DL or LIS3DSH	Pushbutton	LED	SWD	USB	OSC	Free I/O	Power supply	CNS	CN2	P1	P2
PB15	SPI2_MOSI/ I2S2_SD/ TIM1_CH3N/ TIM8_CH3N/ TIM12_CH2/ OTG_HS_DP	54	-	-	-	-	-	-	-	-	-	-	-	-	39	-
PC0	OTG_HS_ULPI_STP/ ADC123_IN10	15	-	-	-	-	-	-	PowerOn	-	-	-	-	-	8	-
PC1	ETH_MDC/ ADC123_IN11	16	-	-	-	-	-	-	-	-	-	-	-	-	7	-
PC2	SPI2_MISO/ OTG_HS_ULPI_DIR/ TH_MII_TXD2/ I2S2ext_SD/ ADC123_IN12	17	-	-	-	-	-	-	-	-	-	-	-	-	10	-
PC3	SPI2_MOSI/ I2S2_SD/ OTG_HS_ULPI_NXT/ ETH_MII_TX_CLK/ ADC123_IN13	18	-	DOU/AIN4x	-	-	-	-	-	-	-	-	-	-	9	-
PC4	ETH_RMII_RX_D0/ ETH_MII_RX_D0/ ADC12_IN14	33	-	-	-	-	-	-	-	-	-	-	-	-	20	-
PC5	ETH_RMII_RX_D1/ ETH_MII_RX_D1/ ADC12_IN15	34	-	-	-	-	-	-	-	-	-	-	-	-	19	-
PC6	I2S2_MCK/ TIM8_CH1/ SDIO_D6/ USART6_TX/ DCMI_D0/ TIM3_CH1	63	-	-	-	-	-	-	-	-	-	-	-	-	-	47

Table 5. MCU pin description versus board function (continued)

MCU pin		Board function															
Main function	Alternate functions	LQFP100	CS43L22	MP450T02	LIS302DL or LIS3DSH	Pushbutton	LED	SWD	USB	OSC	Free I/O	Power supply	CNS	CN2	P1	P2	
PC7	I2S3_MCK/ TIM8_CH2/ SDIO_D7/ USART6_RX/ DCMI_D1/ TIM3_CH2	64	MCLK	-	-	-	-	-	-	-	-	-	-	-	-	-	48
PC8	TIM8_CH3/ SDIO_D0/ TIM3_CH3/ USART6_CK/ DCMI_D2	65	-	-	-	-	-	-	-	-	-	-	-	-	-	-	45
PC9	I2S_CKIN/ MCO2/ TIM8_CH4/ SDIO_D1/ I2C3_SDA/ DCMI_D3/ TIM3_CH4	66	-	-	-	-	-	-	-	-	-	-	-	-	-	-	46
PC10	SPI3_SCK/ I2S3_CK/ UART4_TX/ SDIO_D2/ DCMI_D8/ USART3_TX	78	SCLK	-	-	-	-	-	-	-	-	-	-	-	-	-	37
PC11	UART4_RX/ SPI3_MISO/ SDIO_D3/ DCMI_D4/ USART3_RX/ I2S3ext_SD	79	-	-	-	-	-	-	-	-	-	-	-	-	-	-	38
PC12	UART5_TX/ SDIO_CK/ DCMI_D9/ SPI3_MOSI/ I2S3_SD/ USART3_CK	80	SDIN	-	-	-	-	-	-	-	-	-	-	-	-	-	35



Table 5. MCU pin description versus board function (continued)

MCU pin		Board function															
Main function	Alternate functions	LQFP100	CS43L22	MP450T02	LIS302DL or LIS3DSH	Pushbutton	LED	SWD	USB	OSC	Free I/O	Power supply	CNS	CN2	P1	P2	
PC13	RTC_AF1	7	-	-	-	-	-	-	-	-	-	-	-	-	-	-	12
PC14	OSC32_IN	8	-	-	-	-	-	-	-	OSC32_IN	-	-	-	-	-	-	9
PC15	OSC32_OUT	9	-	-	-	-	-	-	-	OSC32_OUT	-	-	-	-	-	-	10
PD0	FSMC_D2/ CAN1_RX	81	-	-	-	-	-	-	-	-	-	-	-	-	-	-	36
PD1	FSMC_D3/ CAN1_TX	82	-	-	-	-	-	-	-	-	-	-	-	-	-	-	33
PD2	TIM3_ETR/ UART5_RXSDIO_CMD / DCMI_D11	83	-	-	-	-	-	-	-	-	-	-	-	-	-	-	34
PD3	FSMC_CLK/ USART2_CTS	84	-	-	-	-	-	-	-	-	-	-	-	-	-	-	31
PD4	FSMC_NOE/ USART2_RTS	85	RESET	-	-	-	-	-	-	-	-	-	-	-	-	-	32
PD5	FSMC_NWE/ USART2_TX	86	-	-	-	-	RED	-	OverCurrent	-	-	-	-	-	-	-	29
PD6	FSMC_NWAIT/ USART2_RX	87	-	-	-	-	-	-	-	-	-	-	-	-	-	-	30
PD7	USART2_CK/ FSMC_NE1/ FSMC_NCE2	88	-	-	-	-	-	-	-	-	-	-	-	-	-	-	27

Table 5. MCU pin description versus board function (continued)

MCU pin		Board function														
Main function	Alternate functions	LQFP100	CS43L22	MP450T02	LIS302DL or LIS3DSH	Pushbutton	LED	SWD	USB	OSC	Free I/O	Power supply	CN5	CN2	P1	P2
PD8	FSMC_D13/ USART3_TX	55	-	-	-	-	-	-	-	-	-	-	-	-	40	-
PD9	FSMC_D14/ USART3_RX	56	-	-	-	-	-	-	-	-	-	-	-	-	41	-
PD10	FSMC_D15/ USART3_CK	57	-	-	-	-	-	-	-	-	-	-	-	-	42	-
PD11	FSMC_A16/ USART3_CTS	58	-	-	-	-	-	-	-	-	-	-	-	-	43	-
PD12	FSMC_A17/ TIM4_CH1/ USART3_RTS	59	-	-	-	-	GREEN	-	-	-	-	-	-	-	44	-
PD13	FSMC_A18/ TIM4_CH2	60	-	-	-	-	ORANGE	-	-	-	-	-	-	-	45	-
PD14	FSMC_D0/ TIM4_CH3	61	-	-	-	-	RED	-	-	-	-	-	-	-	46	-
PD15	FSMC_D1/ TIM4_CH4	62	-	-	-	-	BLUE	-	-	-	-	-	-	-	47	-
PE0	TIM4_ETR/ FSMC_NBL0/ DCMI_D2	97	-	-	INT1	-	-	-	-	-	-	-	-	-	-	17
PE1	FSMC_NBL1/ DCMI_D3	98	-	-	INT2	-	-	-	-	-	-	-	-	-	-	18
PE2	TRACECLK/ FSMC_A23/ ETH_MII_TXD3	1	-	-	-	-	-	-	-	-	-	-	-	-	-	15
PE3	TRACED0/ FSMC_A19	2	-	-	CS_12C/SPI	-	-	-	-	-	-	-	-	-	-	16



Table 5. MCU pin description versus board function (continued)

MCU pin		Board function															
Main function	Alternate functions	LQFP100	CS43L22	MP450T02	LIS302DL or LIS3DSH	Pushbutton	LED	SWD	USB	OSC	Free I/O	Power supply	CNS	CN2	P1	P2	
PE4	TRACED1/ FSMC_A20/ DCMI_D4	3	-	-	-	-	-	-	-	-	-	-	-	-	-	-	13
PE5	TRACED2/ FSMC_A21/ TIM9_CH1/ DCMI_D6	4	+	+	-	-	-	-	+	-	-	-	-	+	+	+	14
PE6	TRACED3/ FSMC_A22/ TIM9_CH2/ DCMI_D7	5	-	-	-	-	-	-	-	-	-	-	-	-	-	-	11
PE7	FSMC_D4/ TIM1_ETR	38	+	+	-	-	+	+	+	+	-	-	-	-	-	-	25
PE8	FSMC_D5/ TIM1_CH1N	39	-	-	-	-	-	-	-	-	-	-	-	-	-	-	26
PE9	FSMC_D6/ TIM1_CH1	40	-	-	-	-	-	-	-	-	-	-	-	-	-	-	27
PE10	FSMC_D7/ TIM1_CH2N	41	-	-	-	-	-	+	+	-	-	-	-	-	-	-	28
PE11	FSMC_D8/ TIM1_CH2	42	-	-	-	-	-	-	-	-	-	-	-	-	-	-	29
PE12	FSMC_D9/ TIM1_CH3N	43	-	-	-	-	-	-	-	-	-	-	-	-	-	-	30
PE13	FSMC_D10/ TIM1_CH3	44	-	-	-	-	-	-	+	-	-	-	-	-	-	-	31
PE14	FSMC_D11/ TIM1_CH4	45	-	-	-	-	-	-	-	+	-	-	-	-	-	-	32
PE15	FSMC_D12/ TIM1_BKIN	46	-	-	-	-	-	-	-	-	-	-	-	-	-	-	33
PH0	OSC_IN	12	-	-	-	-	-	-	-	OSC_IN	-	-	-	-	-	-	7

Table 5. MCU pin description versus board function (continued)

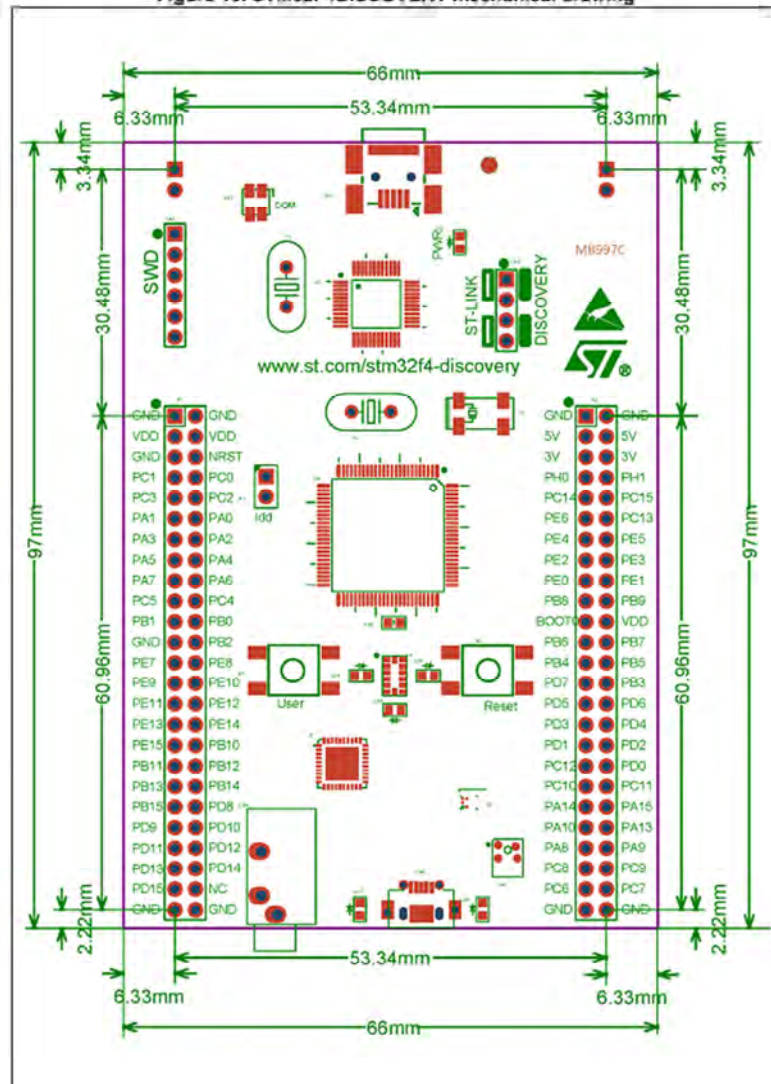
MCU pin		Board function														
Main function	Alternate functions	LQFP100	CS43L22	MP450T02	LIS302DL or LIS3DSH	Pushbutton	LED	SWD	USB	OSC	Free I/O	Power supply	CN5	CN2	P1	P2
PH1	OSC_OUT	13	-	-	-	-	-	-	-	OSC_OUT	-	-	-	-	-	8
-	-	-	-	-	-	-	-	-	-	-	-	5V	-	-	-	3
-	-	-	-	-	-	-	-	-	-	-	-	5V	-	-	-	4
-	-	-	-	-	-	-	-	-	-	-	-	3V	-	-	-	5
-	-	-	-	-	-	-	-	-	-	-	-	3V	-	-	-	6
-	-	-	-	-	-	-	-	-	-	-	-	VDD	-	-	3	-
-	-	-	-	-	-	-	-	-	-	-	-	VDD	-	-	4	-
-	-	-	-	-	-	-	-	-	-	-	-	VDD	-	-	-	22
-	-	-	-	-	-	-	-	-	GND	GND	-	GND	5	3	1	-
-	-	-	-	-	-	-	-	-	-	-	-	GND	-	-	2	-
-	-	-	-	-	-	-	-	-	-	-	-	GND	-	-	5	-
-	-	-	-	-	-	-	-	-	-	-	-	GND	-	-	23	-
-	-	-	-	-	-	-	-	-	-	-	-	GND	-	-	49	-
-	-	-	-	-	-	-	-	-	-	-	-	GND	-	-	50	-
-	-	-	-	-	-	-	-	-	-	-	-	GND	-	-	-	1
-	-	-	-	-	-	-	-	-	-	-	-	GND	-	-	-	2

Table 5. MCU pin description versus board function (continued)

MCU pin		Board function																
Main function	Alternate functions	P2	P1	CN2	CN5	Power supply	Free I/O	OSC	USB	SWD	LED	Pushbutton	LIS302DL or LIS3DSH	MP450T02	C943L22	LQFP100	49	50

5 Mechanical drawing

Figure 10. STM32F4DISCOVERY mechanical drawing



## 9.4 AS5045B



### AS5045B

#### 12-Bit Programmable Magnetic Position Sensor

### 1 General Description

The AS5045B is a contact less magnetic position sensor for accurate angular measurement over a full turn of 360 degrees.

It is a system-on-chip, combining integrated Hall elements, analog front end and digital signal processing in a single device.

To measure the angle, only a simple two-pole magnet, rotating over the center of the chip, is required. The magnet can be placed above or below the IC.

The absolute angle measurement provides instant indication of the magnet's angular position with a resolution of  $0.0879^\circ = 4096$  positions per revolution. This digital data is available as a serial bit stream and as a PWM signal.

An internal voltage regulator allows the AS5045B to operate at either 3.3V or 5V supplies.

- Two digital 12-bit absolute outputs:
  - Serial interface
  - Pulse width modulated (PWM) output
- Quadrature A/B/I output 12-bit
- User programmable zero position
- Failure detection mode for magnet placement, monitoring, and loss of power supply
- Red-Yellow-Green indicators display placement of magnet in Z-axis
- Serial read-out of multiple interconnected AS5045B devices using Daisy Chain mode
- Tolerant to magnet misalignment and gap variations
- Wide temperature range:  $-40^\circ\text{C}$  to  $+125^\circ\text{C}$
- Small Pb-free package: SSOP 16 (5.3mm x 6.2mm)

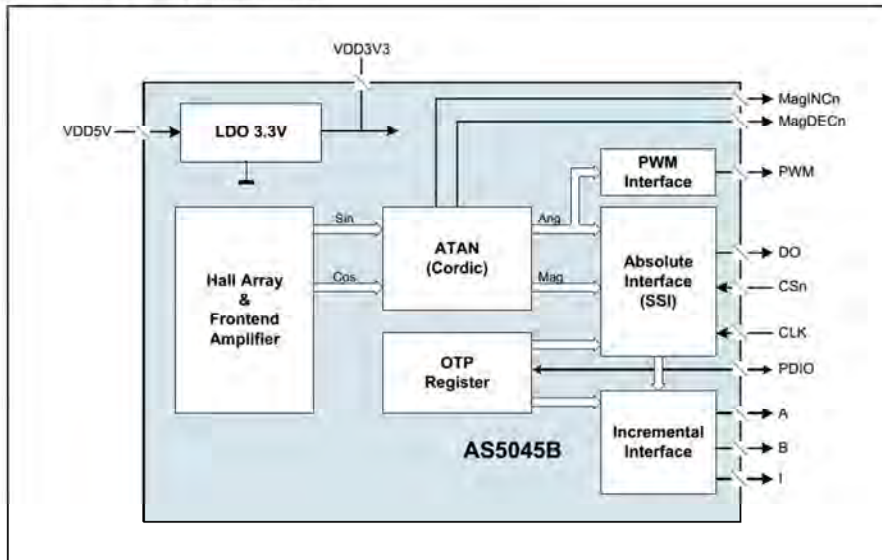
### 2 Key Features

- Contact less high resolution rotary position sensor over a full turn of 360 degrees

### 3 Applications

The device is ideal for industrial applications like automatic or elevator doors, robotics, motor control and optical encoder replacement.

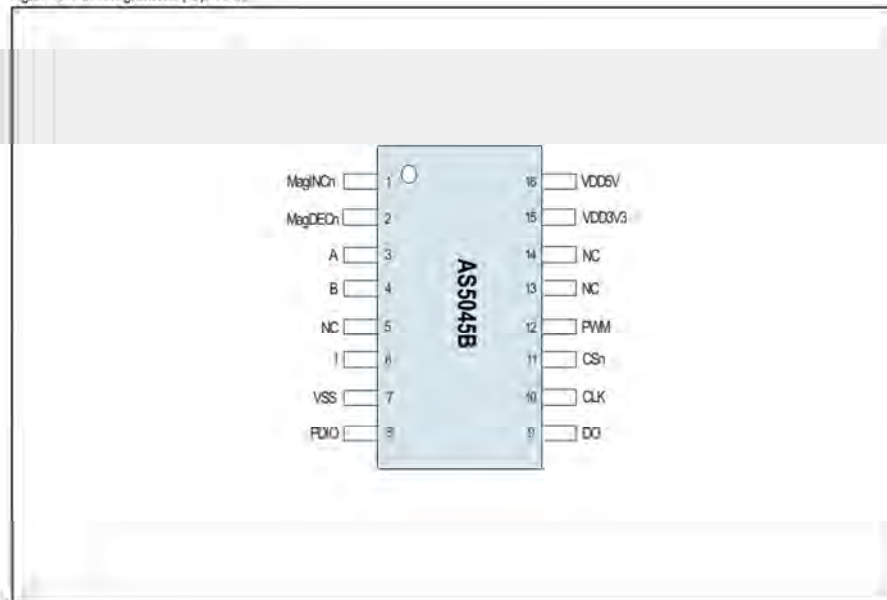
Figure 1. Blockdiagram Rotary Position Sensor IC.





## 4 Pin Assignments

Figure 2. Pin Assignments (Top View)



### 4.1 Pin Descriptions

The following SSOP16 shows the description of each pin of the standard SSOP16 package (Shrink Small Outline Package, 16 leads, body size: 5.3mm x 6.2mm); (see Figure 2).

Table 1. Pin Descriptions

Pin Name	Pin Number	Pin Type	Description
MagINCn	1	Digital output open drain	<b>Magnet Field Magnitude Increase.</b> Active low. Indicates a distance reduction between the magnet and the device surface. (see Table 8)
MagDECn	2		<b>Magnet Field Magnitude Decrease.</b> Active low. Indicates a distance increase between the device and the magnet. (see Table 8)
A	3	Digital output	Quadrature output A (1024 Pulses)
B	4		Quadrature output B (1024 Pulses)
NC	5		Must be left unconnected
I	6	Digital output	Index signal for the quadrature output.
VSS	7	Supply pin	Negative Supply Voltage (GND)
PDIO	8	Digital input pull-down	<b>OTP Programming Input and Data Input for Daisy Chain mode.</b> Pin has an internal pull-down resistor (74kΩ). Connect this pin to VSS if programming is not required.
DO	9	Digital output/ tri-state	<b>Data Output</b> of Synchronous Serial Interface
CLK	10	Digital input, Schmitt-Trigger input	<b>Clock Input</b> of Synchronous Serial Interface; Schmitt-Trigger input





Table 1. Pin Descriptions

Pin Name	Pin Number	Pin Type	Description
CS <sub>n</sub>	11	Digital input pull-down, Schmitt-Trigger input	<b>Chip Select.</b> Active low, Schmitt-Trigger input, internal pull-up resistor (50kΩ)
PWM	12	Digital output	<b>Pulse Width Modulation</b>
NC	13	~	Must be left unconnected
NC	14	~	Must be left unconnected
VDD3V3	15	Supply pin	3V-Regulator Output, internally regulated from VDD5V. Connect to VDD5V for 3V supply voltage. Do not load externally.
VDD5V	16	Supply pin	Positive Supply Voltage, 3.0V to 5.5V

Pin 1 and 2 are the magnetic field change indicators, MagINC<sub>n</sub> and MagDEC<sub>n</sub> (magnetic field strength increase or decrease through variation of the distance between the magnet and the device). These outputs can be used to detect the valid magnetic field range. Furthermore those indicators can also be used for contact-less push-button functionality.

Pin 3 and 4 are used for incremental angle information in 12-bit quadrature signal format.

Pin 6 Index output used for incremental angle information. (Zero position reference).

Pins 7, 15, and 16 are supply pins, pins 5, 13, and 14 are for internal use and must not be connected.

Pin 8 (PDIO) is used to program the zero-position into the OTP(see page 17). This pin is also used as digital input to shift serial data through the device in Daisy Chain configuration, (see page 13).

Pin 11 Chip Select (CS<sub>n</sub>; active low) selects a device within a network of AS5045Bs and initiates serial data transfer. A logic high at CS<sub>n</sub> puts the data output pin (DO) to tri-state and terminates serial data transfer. This pin is also used for alignment mode (see Alignment Mode on page 21) and programming mode (see Programming the AS5045B on page 17).

Pin 12 allows a single wire output of the 12-bit absolute position value. The value is encoded into a pulse width modulated signal with 1μs pulse width per step (1μs to 4096μs over a full turn). By using an external low pass filter, the digital PWM signal is converted into an analog voltage, e.g. for making a direct replacement of potentiometers possible.



## 5 Absolute Maximum Ratings

Stresses beyond those listed in Table 2 may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated in Section 6 Electrical Characteristics on page 6 is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Table 2. Absolute Maximum Ratings

Parameter	Min	Max	Units	Comments
<b>Electrical Parameters</b>				
DC supply voltage at pin VDD5V	-0.3	7	V	
DC supply voltage at pin VDD3V3		5	V	
Input pin voltage	-0.3	VDD5V +0.3	V	Except VDD3V3
Input current (latchup immunity)	-100	100	mA	Norm: EIA/JESD78 Class II Level A
<b>Electrostatic Discharge</b>				
Electrostatic discharge		± 2	kV	Norm: JESD22-A114E
<b>Temperature Ranges and Storage Conditions</b>				
Storage temperature	-55	150	°C	Min -67°F; Max +302°F
Package Body temperature		260	°C	The reflow peak soldering temperature (body temperature) specified is in accordance with IPC/JEDEC J-STD-020 "Moisture/Reflow Sensitivity Classification for Non-Hermetic Solid State Surface Mount Devices". The lead finish for Pb-free leaded packages is matte tin (100% Sn).
Humidity non-condensing	5	85	%	
Moisture Sensitive Level (MSL)		3		Represents a maximum floor time of 168h





## 7 Timing Characteristics

T<sub>AMB</sub> = -40 to +125 °C, V<sub>DD5V</sub> = 3.0 to 3.6V (3V operation) V<sub>DD5V</sub> = 4.5 to 5.5V (5V operation), unless otherwise noted. Also valid for version I.

Table 6. Timing Characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Units
<b>Synchronous Serial Interface (SSI)</b>						
t <sub>DOActive</sub>	Data output activated (logic high)	Time between falling edge of CS <sub>n</sub> and data output activated			100	ns
t <sub>CLKFE</sub>	First data shifted to output register	Time between falling edge of CS <sub>n</sub> and first falling edge of CLK	500			ns
T <sub>CLKZ</sub>	Start of data output	Rising edge of CLK shifts out one bit at a time	500			ns
t <sub>DOValid</sub>	Data output valid	Time between rising edge of CLK and data output valid			413	ns
t <sub>DOInstate</sub>	Data output tri-state	After the last bit DO changes back to "tri-state"			100	ns
t <sub>CSn</sub>	Pulse width of CS <sub>n</sub>	CS <sub>n</sub> =high; To initiate read-out of next angular position	500			ns
f <sub>CLK</sub>	Read-out frequency	Clock frequency to read out serial data	>0		1	MHz
<b>Pulse Width Modulation Output</b>						
f <sub>PWM</sub>	PWM frequency	Signal period = 4098µs ±10% at T <sub>AMB</sub> = -40 to +125°C	220	244	268	Hz
PW <sub>MIN</sub>	Minimum pulse width	Position 0d; angle 0 degree	0.90	1	1.10	µs
PW <sub>MAX</sub>	Maximum pulse width	Position 4098d; angle 359.91 degrees	3686	4096	4506	µs
<b>Programming Conditions</b>						
t <sub>PROG</sub>	Programming time per bit	Time to prog. a single fuse bit	10		20	µs
t <sub>CHARGE</sub>	Refresh time per bit	Time to charge the cap after t <sub>PROG</sub>	1			µs
f <sub>LOAD</sub>	LOAD frequency	Data can be loaded at n x 2µs			500	kHz
f <sub>READ</sub>	READ frequency	Read the data from the latch			2.5	MHz
f <sub>WRITE</sub>	WRITE frequency	Write the data to the latch			2.5	MHz



## 8 Detailed Description

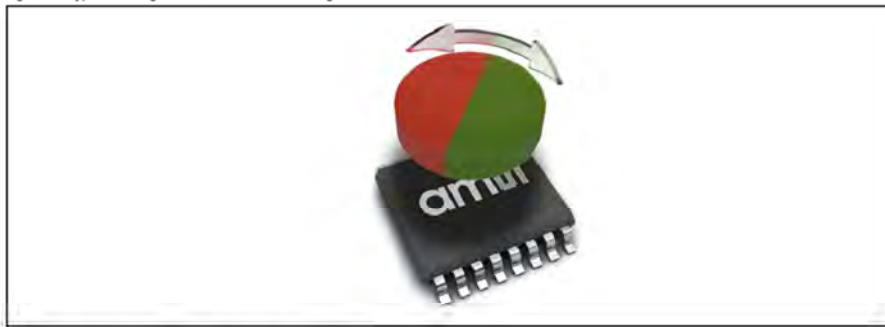
The AS5045B is manufactured in a CMOS standard process and uses a spinning current Hall technology for sensing the magnetic field distribution across the surface of the chip. The integrated Hall elements are placed around the center of the device and deliver a voltage representation of the magnetic field at the surface of the IC.

Through Sigma-Delta Analog / Digital Conversion and Digital Signal-Processing (DSP) algorithms, the AS5045B provides accurate high-resolution absolute angular position information. For this purpose a Coordinate Rotation Digital Computer (CORDIC) calculates the angle and the magnitude of the Hall array signals.

The DSP is also used to provide digital information at the outputs MagINCn and MagDECn that indicate movements of the used magnet towards or away from the device's surface. A small low cost diametrically magnetized (two-pole) standard magnet provides the angular position information (see Figure 16).

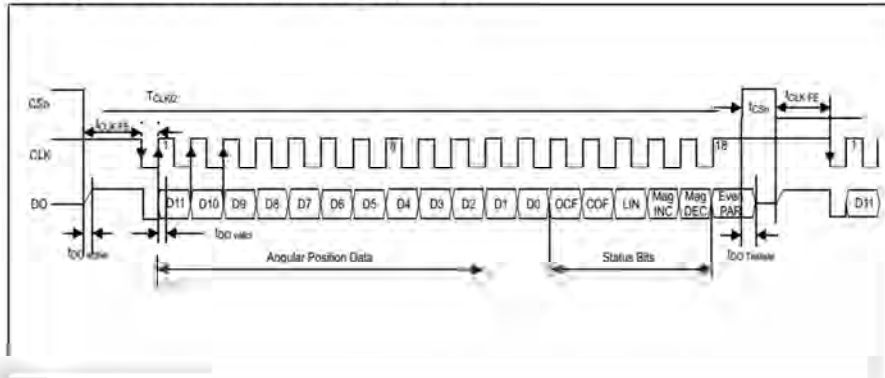
The AS5045B senses the orientation of the magnetic field and calculates a 12-bit binary code. This code can be accessed via a Synchronous Serial Interface (SSI). In addition, an absolute angular representation is given by a Pulse Width Modulated signal at pin 12 (PWM). This PWM signal output also allows the generation of a direct proportional analog voltage, by using an external Low-Pass-Filter. The AS5045B is tolerant to magnet misalignment and magnetic stray fields due to differential measurement technique and Hall sensor conditioning circuitry.

Figure 4. Typical Arrangement of AS5045B and Magnet



### 8.1 Synchronous Serial Interface (SSI)

Figure 5. Synchronous Serial Interface with Absolute Angular Position Data





If CSn changes to logic low, Data Out (DO) will change from high impedance (tri-state) to logic high and the read-out will be initiated.

- After a minimum time  $t_{CLK,FE}$ , data is latched into the output shift register with the first falling edge of CLK.
- Each subsequent rising CLK edge shifts out one bit of data.
- The serial word contains 18 bits, the first 12 bits are the angular information D[11:0], the subsequent 6 bits contain system information, about the validity of data such as OCF, COF, LIN, Parity and Magnetic Field status (increase/decrease).
- A subsequent measurement is initiated by a "high" pulse at CSn with a minimum duration of  $t_{CSn}$ .

#### Data Content

**D11:D0** absolute angular position data (MSB is clocked out first)

**OCF** (Offset Compensation Finished), logic high indicates the finished Offset Compensation Algorithm

**COF** (Cordic Overflow), logic high indicates an out of range error in the CORDIC part. When this bit is set, the data at D11:D0 is invalid. The absolute output maintains the last valid angular value.

This alarm can be resolved by bringing the magnet within the X-Y-Z tolerance limits.

**LIN** (Linearity Alarm), logic high indicates that the input field generates a critical output linearity.

When this bit is set, the data at D11:D0 can still be used, but can contain invalid data. This warning can be resolved by bringing the magnet within the X-Y-Z tolerance limits.

**Even Parity** bit for transmission error detection of bits 1...17 (D11...D0, OCF, COF, LIN, MagINC, MagDEC)

Placing the magnet above the chip, angular values increase in clockwise direction by default.

Data D11:D0 is valid, when the status bits have the following configurations:

Table 7. Status Bit Outputs

OCF	COF	LIN	Mag INC	Mag DEC	Parity
0	0	0	0	0	Even checksum of bits 1:15
0	0	0	0	1	
0	0	0	1	0	
0	0	0	1	1	

**Note:** MagInc=MagDec=1 is only recommended in YELLOW mode (see Table 8)

**Z-axis Range Indication (Push Button Feature, Red/Yellow/Green Indicator).** The AS5045B provides several options of detecting movement and distance of the magnet in the Z-direction. Signal indicators MagINCn and MagDECn are available both as hardware pins (pins #1 and 2) and as status bits in the serial data stream (see Figure 5).

In the default state, the status bits MagINC, MagDec and pins MagINCn, MagDECn have the following function:

Table 8. Magnetic Field Strength Red-Yellow-Green Indicator

Status Bits			Hardware Pins		OTP: Mag CompEn = 1 (Red-Yellow-Green)
Mag INC	Mag DEC	LIN	Mag INCn	Mag DECn	Description
0	0	0	Off	Off	No distance change Magnetic input field OK (GREEN range, ~45...75mT)
1	1	0	On	Off	YELLOW range: magnetic field is ~25...45mT or ~75...135mT. The AS5045B can still be operated in this range, but with slightly reduced accuracy.
1	1	1	On	On	RED range: magnetic field is <-25mT or >~135mT. It is still possible to operate the AS5045B in the red range, but not recommended.
All other combinations			n/a	n/a	Not available

**Note:** Pin 1 (MagINCn) and pin 2 (MagDECn) are active low via open drain output and require an external pull-up resistor. If the magnetic field is in range, both outputs are turned off.



The two pins can also be combined with a single pull-up resistor. In this case, the signal is high when the magnetic field is in range. It is low in all other cases (see Table 8).

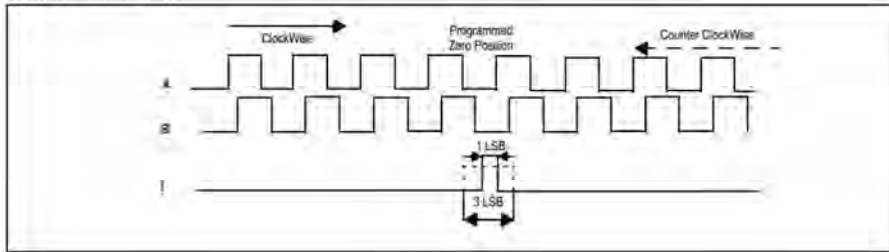
## 8.2 Incremental Mode

The AS5045B has an internal interpolator block. This function is used if the input magnetic field is too fast and a code position is missing. In this case an interpolation is done.

**Incremental Power-up Lock Option.** After power-up, the incremental outputs can optionally be locked or unlocked, depending on the status of the CSn pin:

- CSn = low at power-up: CSn has an internal pull-up resistor and must be externally pulled low ( $R_{ext} \leq 5k\Omega$ ). If CSn is low at power-up, the incremental outputs (A, B, Index) will be high until the internal offset compensation is finished. This unique state (A=B=Index = high) can be used as an indicator for the external controller to shorten the waiting time at power-up. Instead of waiting for the specified maximum power-up-time (0), the controller can start requesting data from the AS5045B as soon as the state (A=B=Index = high) is cleared.
- CSn = high or open at power-up: In this mode, the incremental outputs (A, B, Index) will remain at logic high state, until CSn goes low or a low pulse is applied at CSn. This mode allows intentional disabling of the incremental outputs until, for example the system microcontroller is ready to receive data.

Figure 6. Incremental Output

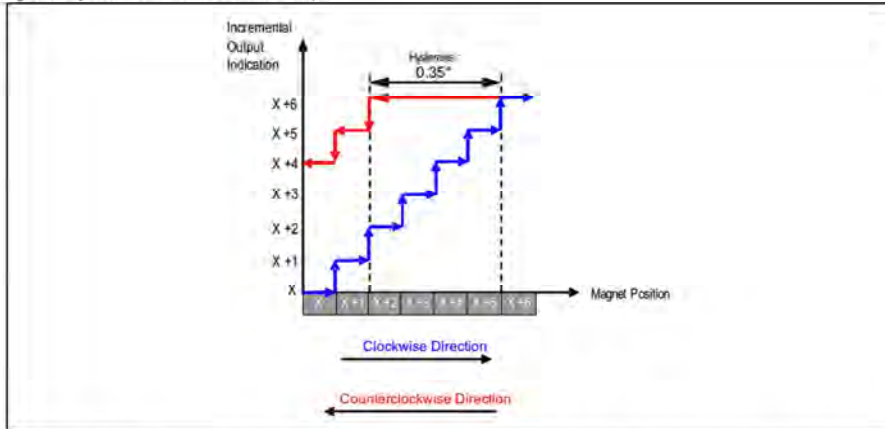


The hysteresis trimming is done at the final test (factory trimming) and set to 4 LSB, related to a 12-bit number.

**Incremental Output Hysteresis.** To avoid flickering incremental outputs at a stationary magnet position, a hysteresis is introduced. In case of a rotational direction change, the incremental outputs have a hysteresis of 4 LSB. Regardless of the programmed incremental resolution, the hysteresis of 4 LSB always corresponds to the highest resolution of 12-bit. In absolute terms, the hysteresis is set to 0.35 degrees for all resolutions. For constant rotational directions, every magnet position change is indicated at the incremental outputs (see Figure 7). For example, if the magnet turns clockwise from position "x+3" to "x+4", the incremental output would also indicate this position accordingly. A change of the magnet's rotational direction back to position "x+3" means that the incremental output still remains unchanged for the duration of 4 LSB, until position "x+2" is reached. Following this direction, the incremental outputs will again be updated with every change of the magnet position.



Figure 7. Hysteresis Window for Incremental Outputs



**Incremental Output Validity.** During power on the incremental output is kept stable high until the offset compensation is finished and the CSn is low (internal Pull Up) the first time. In quadrature mode  $A = B = \text{Index} = \text{high}$  indicates an invalid output. If the interpolator recognizes a difference larger than 128 steps between two samples it holds the last valid state. The interpolator synchronizes up again with the next valid difference. This avoids undefined output burst, e.g. if no magnet is present.

### 8.3 Daisy Chain Mode

The Daisy Chain mode allows connection of several AS5045Bs in series, while still keeping just one digital input for data transfer (see "Data IN" in Figure 8). This mode is accomplished by connecting the data output (DO; pin 9) to the data input (PDIO; pin 8) of the subsequent device. The serial data of all connected devices is read from the DO pin of the first device in the chain. The length of the serial bit stream increases with every connected device, it is  $n * (18+1)$  bits;  $n$  = number of devices, e.g. 38 bit for two devices, 57 bit for three devices, etc.

The last data bit of the first device (Parity) is followed by a dummy bit and the first data bit of the second device (D11), etc. (see Figure 9).

Figure 8. Daisy Chain Hardware Configuration

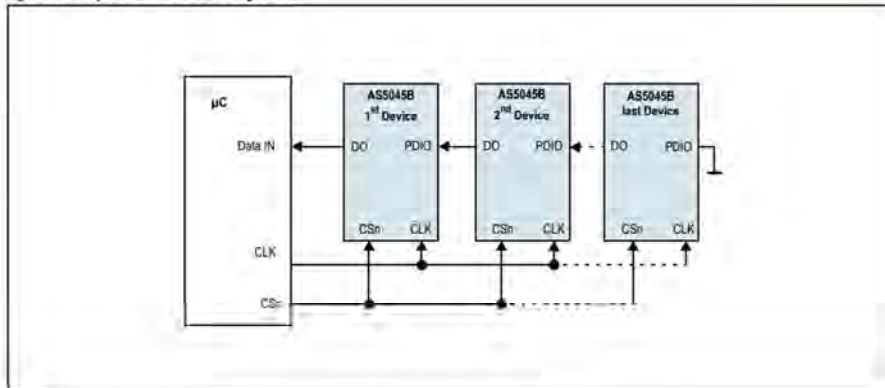
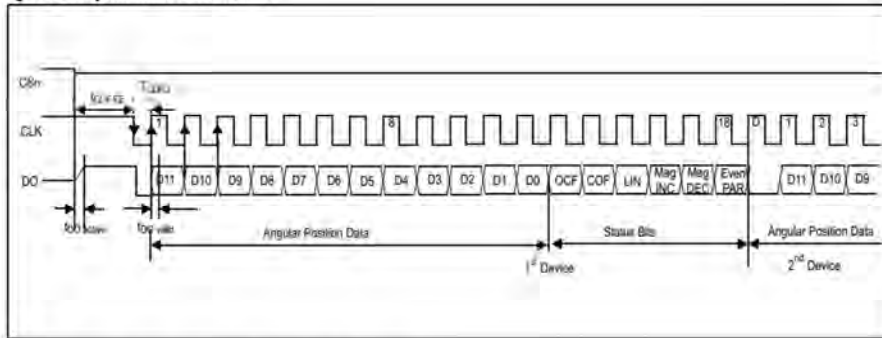






Figure 9. Daisy Chain Mode Data Transfer



### 8.4 Pulse Width Modulation (PWM) Output

The AS5045B provides a pulse width modulated output (PWM), whose duty cycle is proportional to the measured angle. For angle position 0 to 4094

$$Position = \frac{t_{on} - 4098}{(t_{on} + t_{off})} - 1 \tag{EQ 1}$$

**Examples:**

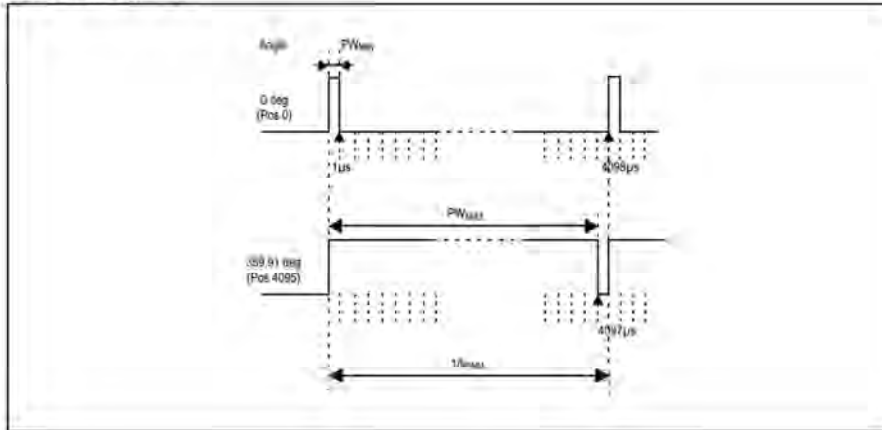
1. An angle position of 180° will generate a pulse width  $t_{on} = 2049\mu s$  and a pause toff of 2049  $\mu s$  resulting in Position = 2048 after the calculation:  $2049 * 4098 / (2049 + 2049) - 1 = 2048$
2. An angle position of 359.8° will generate a pulse width  $t_{on} = 4095\mu s$  and a pause toff of 3  $\mu s$  resulting in Position = 4094 after the calculation:  $4095 * 4098 / (4095 + 3) - 1 = 4094$

**Exception:**

1. An angle position of 359.9° will generate a pulse width  $t_{on} = 4097\mu s$  and a pause toff of 1  $\mu s$  resulting in Position = 4096 after the calculation:  $4097 * 4098 / (4097 + 1) - 1 = 4096$

The PWM frequency is internally trimmed to an accuracy of ±5% (±10% over full temperature range). This tolerance can be cancelled by measuring the complete duty cycle as shown above.

Figure 10. PWM Output Signal





### 8.4.1 Changing the PWM Frequency

The PWM frequency of the AS5045B can be divided by two by setting a bit (PWMhalfEN) in the OTP register (see Programming the AS5045B on page 17). With PWMhalfEN = 0 the PWM timing is as shown in Table 9:

Table 9. PWM Signal Parameters (Default mode)

Symbol	Parameter	Typ	Unit	Note
$f_{PWM}$	PWM frequency	244	Hz	Signal period: 4097 $\mu$ s
$PW_{MIN}$	MIN pulse width	1	$\mu$ s	- Position 0d - Angle 0 deg
$PW_{MAX}$	MAX pulse width	4097	$\mu$ s	- Position 4095d - Angle 359.91 deg

When PWMhalfEN = 1, the PWM timing is as shown in Table 10:

Table 10. PWM Signal Parameters with Half Frequency (OTP option)

Symbol	Parameter	Typ	Unit	Note
$f_{PWM}$	PWM frequency	122	Hz	Signal period: 8194 $\mu$ s
$PW_{MIN}$	MIN pulse width	2	$\mu$ s	- Position 0d - Angle 0 deg
$PW_{MAX}$	MAX pulse width	8194	$\mu$ s	- Position 4095d - Angle 359.91 deg

## 8.5 Analog Output

An analog output can be generated by averaging the PWM signal, using an external active or passive low pass filter. The analog output voltage is proportional to the angle. 0° = 0V, 360° =  $V_{DD5V}$ .

Using this method, the AS5045B can be used as direct replacement of potentiometers

Figure 11. Simple 2nd Order Passive RC Low Pass Filter

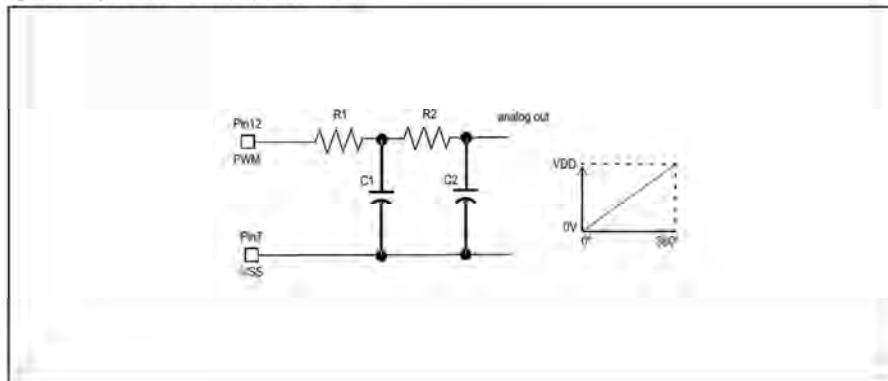


Figure 10 shows an example of a simple passive low pass filter to generate the analog output.

$$R1, R2 \geq 10k\Omega \quad C1, C2 \geq 2.2\mu F / 6V \quad (EQ 2)$$

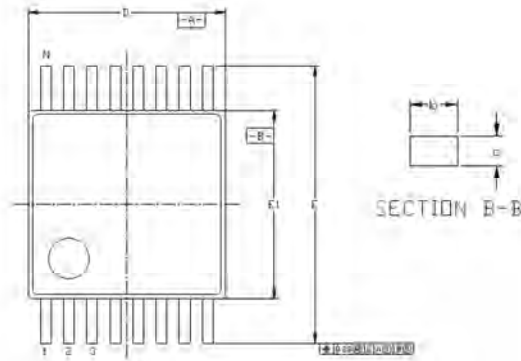
R1 should be greater than or equal to 4k $\Omega$  to avoid loading of the PWM output. Larger values of  $R_x$  and  $C_x$  will provide better filtering and less ripple, but will also slow down the response time.



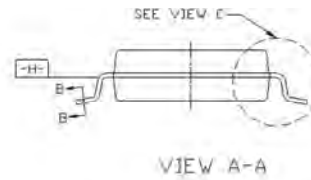
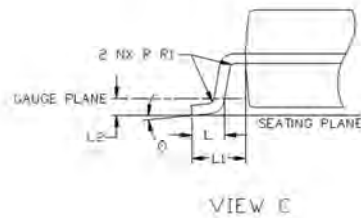
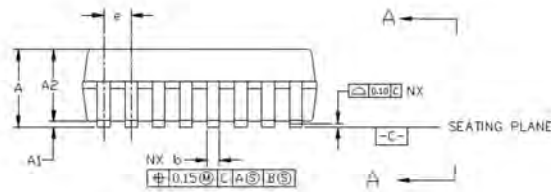
### 10 Package Drawings and Markings

The device is available in SSOP 16 (5.3mm x 6.2mm).

Figure 20. Package Drawings and Dimensions



Symbol	Min	Nom	Max
A	1.73	1.86	1.99
A1	0.05	0.13	0.21
A2	1.68	1.73	1.78
b	0.22	0.315	0.38
e	0.09	0.17	0.25
D	5.90	6.20	6.50
E	7.40	7.80	8.20
E1	5.00	5.30	5.60
e	-	0.65 BSC	-
L	0.55	0.75	0.95
L1	-	1.25 REF	-
L2	-	0.25 BSC	-
R	0.09	-	-
Θ	0°	4°	8°
N	16		



**Notes:**

1. Dimensions and tolerancing conform to ASME Y14.5M-1994.
2. All dimensions are in millimeters. Angles are in degrees.

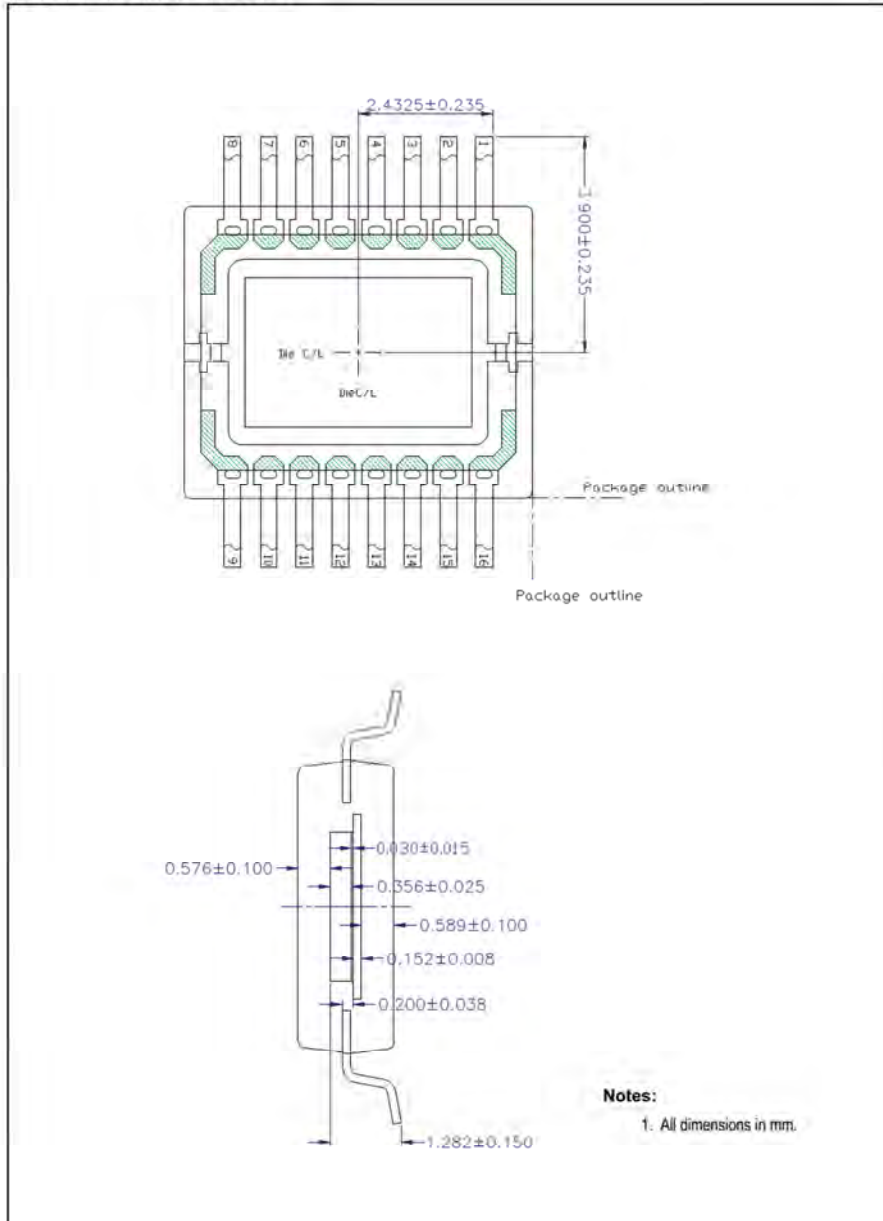
**Marking: YYWWMZZ.**

YY	WW	M	ZZ
Last two digits of the manufacturing year	Manufacturing week	Plant identifier	Assembly traceability code





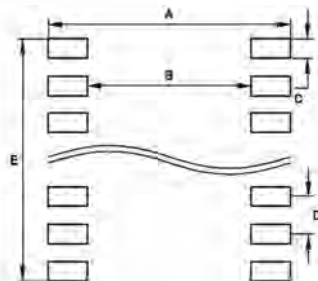
Figure 21. Vertical Cross Section of SSOP-16





## 10.1 Recommended PCB Footprint

Figure 22. PCB Footprint



Recommended Footprint Data	
Symbol	mm
A	9.02
B	6.16
C	0.46
D	0.85
E	5.01

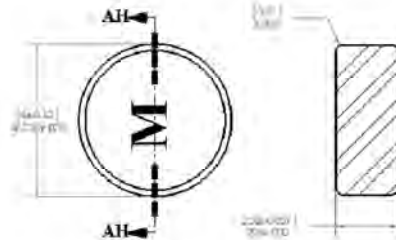


# D6.0x2.5-N35

ordering number: 2910041-1

### Physical Data

Diameter: 6.0 +/- .1mm  
 Thickness: 2.5 +/- .05mm  
 Density: 7.6 gm/cc  
 Mass: ~.53gram  
 Temperature coeff. of max. Linear expansion: 5.0 to 8.0 x 10<sup>-6</sup> °C<sup>-1</sup>



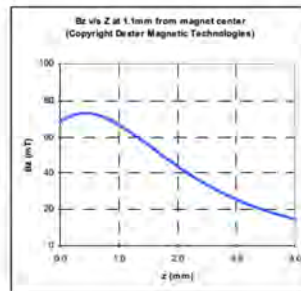
### Magnetic Characteristics

Material: NdFeB  
 Energy grade: 35 MGOe  
 Temperature coeff. of magnetic field strength: -0.15% °C<sup>-1</sup>  
 Max. Operating temp: 120°C  
 Coating: Nickel plating



### Magnetic Performance Data

Magnetic flux density plot for Use with AMS On-axis sensors: (max Bz on line at 1.1mm from center of magnet)



For additional data and options, visit our web site [www.dextermag.com](http://www.dextermag.com)  
 Contact us at: Tel. 1+(800) 317-2537 email, [info@dextermag.com](mailto:info@dextermag.com)

Document D6.0x2.5-N35 Rev-1 2011-01-18

## 9.5 L298



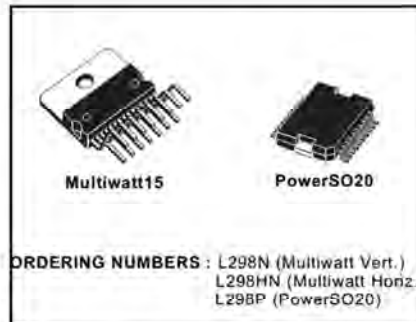
**L298**

### DUAL FULL-BRIDGE DRIVER

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

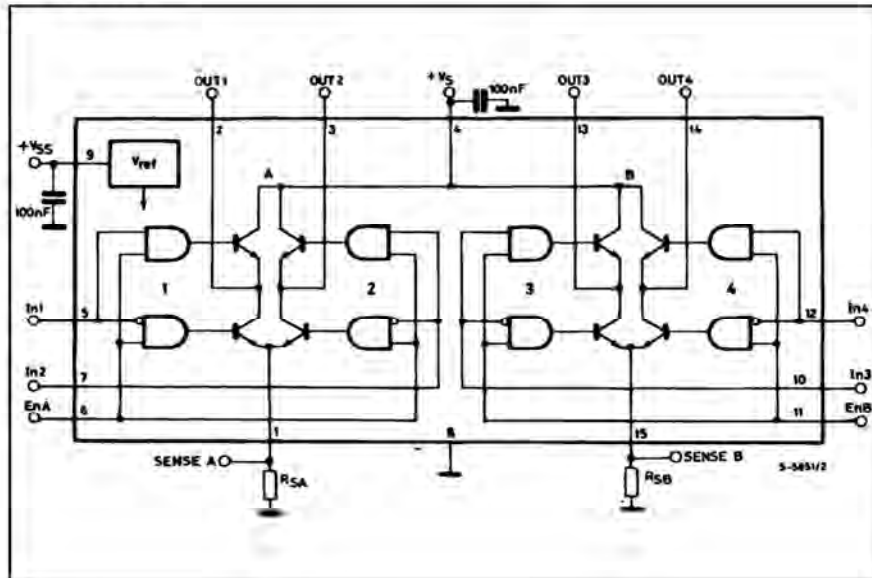
#### DESCRIPTION

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



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

#### BLOCK DIAGRAM



January 2000

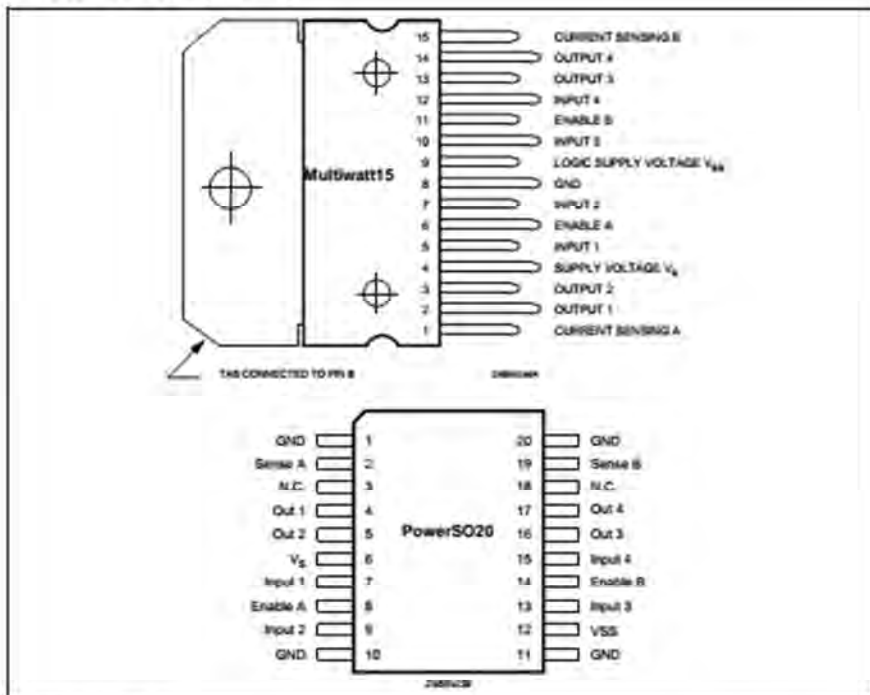
1/13

## L298

### ABSOLUTE MAXIMUM RATINGS

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

### PIN CONNECTIONS (top view)



### THERMAL DATA

Symbol	Parameter	PowerSO20	Multiwatt15	Unit
$R_{\theta(j-c)}$	Thermal Resistance Junction-case	Max. -	3	$^\circ C/W$
$R_{\theta(j-a)}$	Thermal Resistance Junction-ambient	Max. 13 (*)	35	$^\circ C/W$

(\*) Mounted on aluminum substrate

## PIN FUNCTIONS (refer to the block diagram)

MW.15	PowerSO	Name	Function
1;15	2;19	Sense A; Sense B	Between this pin and ground is connected the sense resistor to control the current of the load.
2;3	4;5	Out 1; Out 2	Outputs of the Bridge A; the current that flows through the load connected between these two pins is monitored at pin 1.
4	6	V <sub>S</sub>	Supply Voltage for the Power Output Stages. A non-inductive 100nF capacitor must be connected between this pin and ground.
5;7	7;9	Input 1; Input 2	TTL Compatible Inputs of the Bridge A.
6;11	8;14	Enable A; Enable B	TTL Compatible Enable Input: the L state disables the bridge A (enable A) and/or the bridge B (enable B).
8	1,10,11,20	GND	Ground.
9	12	V <sub>SS</sub>	Supply Voltage for the Logic Blocks. A 100nF capacitor must be connected between this pin and ground.
10; 12	13;15	Input 3; Input 4	TTL Compatible Inputs of the Bridge B.
13; 14	16;17	Out 3; Out 4	Outputs of the Bridge B. The current that flows through the load connected between these two pins is monitored at pin 15.
—	3;18	N.C.	Not Connected

ELECTRICAL CHARACTERISTICS (V<sub>S</sub> = 42V; V<sub>SS</sub> = 5V, T<sub>J</sub> = 25°C; unless otherwise specified)

Symbol	Parameter	Test Conditions	Min.	Typ.	Max.	Unit
V <sub>S</sub>	Supply Voltage (pin 4)	Operative Condition	V <sub>in</sub> +2.5		46	V
V <sub>SS</sub>	Logic Supply Voltage (pin 9)		4.5	5	7	V
I <sub>S</sub>	Quiescent Supply Current (pin 4)	V <sub>en</sub> = H; I <sub>L</sub> = 0 V <sub>i</sub> = L V <sub>i</sub> = H		13 50	22 70	mA mA
I <sub>SS</sub>	Quiescent Current from V <sub>SS</sub> (pin 9)	V <sub>en</sub> = L V <sub>i</sub> = X V <sub>en</sub> = H; I <sub>L</sub> = 0 V <sub>i</sub> = L V <sub>i</sub> = H V <sub>en</sub> = L V <sub>i</sub> = X		24 7	36 12 6	mA mA mA
V <sub>L</sub>	Input Low Voltage (pins 5, 7, 10, 12)		-0.3		1.5	V
V <sub>H</sub>	Input High Voltage (pins 5, 7, 10, 12)		2.3		V <sub>SS</sub>	V
I <sub>L</sub>	Low Voltage Input Current (pins 5, 7, 10, 12)	V <sub>i</sub> = L			-10	μA
I <sub>H</sub>	High Voltage Input Current (pins 5, 7, 10, 12)	V <sub>i</sub> = H ≤ V <sub>SS</sub> - 0.6V		30	100	μA
V <sub>en</sub> = L	Enable Low Voltage (pins 6, 11)		-0.3		1.5	V
V <sub>en</sub> = H	Enable High Voltage (pins 6, 11)		2.3		V <sub>SS</sub>	V
I <sub>en</sub> = L	Low Voltage Enable Current (pins 6, 11)	V <sub>en</sub> = L			-10	μA
I <sub>en</sub> = H	High Voltage Enable Current (pins 6, 11)	V <sub>en</sub> = H ≤ V <sub>SS</sub> - 0.6V		30	100	μA
V <sub>CEsat(H)</sub>	Source Saturation Voltage	I <sub>L</sub> = 1A I <sub>L</sub> = 2A	0.95	1.35 2	1.7 2.7	V V
V <sub>CEsat(L)</sub>	Sink Saturation Voltage	I <sub>L</sub> = 1A (5) I <sub>L</sub> = 2A (5)	0.85	1.2 1.7	1.6 2.3	V V
V <sub>CEsat</sub>	Total Drop	I <sub>L</sub> = 1A (5) I <sub>L</sub> = 2A (5)	1.80		3.2 4.9	V V
V <sub>sens</sub>	Sensing Voltage (pins 1, 15)		-1 (1)		2	V

ELECTRICAL CHARACTERISTICS (continued)

Symbol	Parameter	Test Conditions	Min.	Typ.	Max.	Unit
T <sub>1</sub> (V <sub>I</sub> )	Source Current Turn-off Delay	0.5 V <sub>I</sub> to 0.9 I <sub>L</sub> (2); (4)		1.5		μs
T <sub>2</sub> (V <sub>I</sub> )	Source Current Fall Time	0.9 I <sub>L</sub> to 0.1 I <sub>L</sub> (2); (4)		0.2		μs
T <sub>3</sub> (V <sub>I</sub> )	Source Current Turn-on Delay	0.5 V <sub>I</sub> to 0.1 I <sub>L</sub> (2); (4)		2		μs
T <sub>4</sub> (V <sub>I</sub> )	Source Current Rise Time	0.1 I <sub>L</sub> to 0.9 I <sub>L</sub> (2); (4)		0.7		μs
T <sub>5</sub> (V <sub>I</sub> )	Sink Current Turn-off Delay	0.5 V <sub>I</sub> to 0.9 I <sub>L</sub> (3); (4)		0.7		μs
T <sub>6</sub> (V <sub>I</sub> )	Sink Current Fall Time	0.9 I <sub>L</sub> to 0.1 I <sub>L</sub> (3); (4)		0.25		μs
T <sub>7</sub> (V <sub>I</sub> )	Sink Current Turn-on Delay	0.5 V <sub>I</sub> to 0.9 I <sub>L</sub> (3); (4)		1.6		μs
T <sub>8</sub> (V <sub>I</sub> )	Sink Current Rise Time	0.1 I <sub>L</sub> to 0.9 I <sub>L</sub> (3); (4)		0.2		μs
f <sub>c</sub> (V <sub>I</sub> )	Commutation Frequency	I <sub>L</sub> = 2A		25	40	KHz
T <sub>1</sub> (V <sub>en</sub> )	Source Current Turn-off Delay	0.5 V <sub>en</sub> to 0.9 I <sub>L</sub> (2); (4)		3		μs
T <sub>2</sub> (V <sub>en</sub> )	Source Current Fall Time	0.9 I <sub>L</sub> to 0.1 I <sub>L</sub> (2); (4)		1		μs
T <sub>3</sub> (V <sub>en</sub> )	Source Current Turn-on Delay	0.5 V <sub>en</sub> to 0.1 I <sub>L</sub> (2); (4)		0.3		μs
T <sub>4</sub> (V <sub>en</sub> )	Source Current Rise Time	0.1 I <sub>L</sub> to 0.9 I <sub>L</sub> (2); (4)		0.4		μs
T <sub>5</sub> (V <sub>en</sub> )	Sink Current Turn-off Delay	0.5 V <sub>en</sub> to 0.9 I <sub>L</sub> (3); (4)		2.2		μs
T <sub>6</sub> (V <sub>en</sub> )	Sink Current Fall Time	0.9 I <sub>L</sub> to 0.1 I <sub>L</sub> (3); (4)		0.35		μs
T <sub>7</sub> (V <sub>en</sub> )	Sink Current Turn-on Delay	0.5 V <sub>en</sub> to 0.9 I <sub>L</sub> (3); (4)		0.25		μs
T <sub>8</sub> (V <sub>en</sub> )	Sink Current Rise Time	0.1 I <sub>L</sub> to 0.9 I <sub>L</sub> (3); (4)		0.1		μs

- 1) Sensing voltage can be -1 V for I<sub>L</sub> ≤ 50 μsec; in steady state V<sub>sen</sub> min ≥ 0.5 V.
- 2) See fig. 2.
- 3) See fig. 4.
- 4) The load must be a pure resistor.

Figure 1 : Typical Saturation Voltage vs. Output Current.

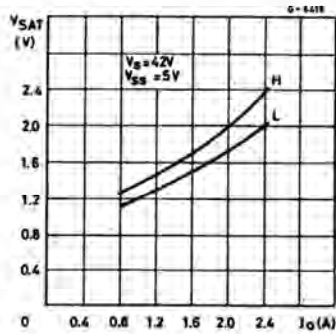
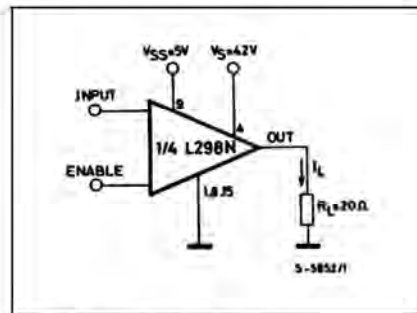


Figure 2 : Switching Times Test Circuits.



Note : For INPUT Switching, set EN = H  
For ENABLE Switching, set IN = H

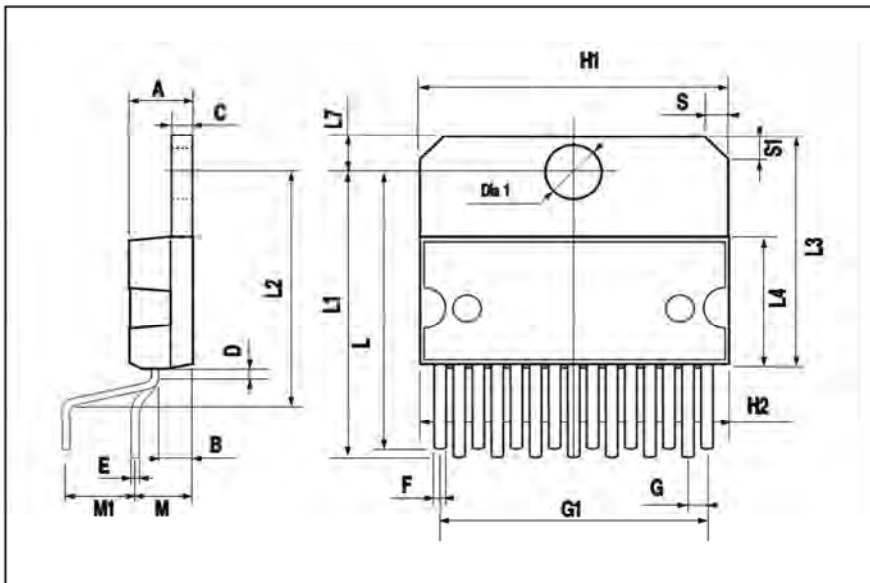
**L298**

DIM.	mm			inch		
	MIN.	TYP.	MAX.	MIN.	TYP.	MAX.
A			5			0.197
B			2.65			0.104
C			1.6			0.063
D		1			0.039	
E	0.49		0.55	0.019		0.022
F	0.66		0.75	0.026		0.030
G	1.02	1.27	1.52	0.040	0.050	0.060
G1	17.53	17.78	18.03	0.690	0.700	0.710
H1	19.6			0.772		
H2			20.2			0.795
L	21.9	22.2	22.5	0.862	0.874	0.886
L1	21.7	22.1	22.5	0.854	0.870	0.886
L2	17.65		18.1	0.695		0.713
L3	17.25	17.5	17.75	0.679	0.689	0.699
L4	10.3	10.7	10.9	0.406	0.421	0.429
L7	2.65		2.9	0.104		0.114
M	4.25	4.55	4.85	0.167	0.179	0.191
M1	4.63	5.08	5.53	0.182	0.200	0.218
S	1.9		2.6	0.075		0.102
S1	1.9		2.6	0.075		0.102
Dia1	3.65		3.85	0.144		0.152

**OUTLINE AND MECHANICAL DATA**



**Multiwatt15 V**



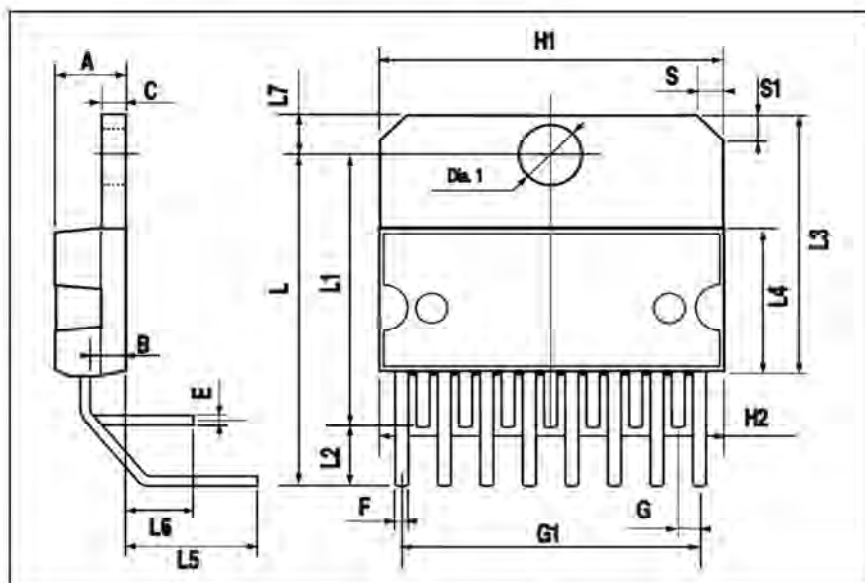


DIM.	mm			inch		
	MIN.	TYP.	MAX.	MIN.	TYP.	MAX.
A			5			0.197
B			2.65			0.104
C			1.6			0.063
E	0.49		0.55	0.019		0.022
F	0.66		0.75	0.026		0.030
G	1.14	1.27	1.4	0.045	0.050	0.055
G1	17.57	17.78	17.91	0.692	0.700	0.705
H1	19.6			0.772		
H2			20.2			0.795
L		20.57			0.810	
L1		18.03			0.710	
L2		2.54			0.100	
L3	17.25	17.5	17.75	0.679	0.689	0.699
L4	10.3	10.7	10.9	0.406	0.421	0.429
L5		5.28			0.208	
L6		2.38			0.094	
L7	2.65		2.9	0.104		0.114
S	1.9		2.6	0.075		0.102
S1	1.9		2.6	0.075		0.102
Di1	3.65		3.85	0.144		0.152

### OUTLINE AND MECHANICAL DATA



**Multiwatt15 H**



DIM.	mm			inch		
	MIN.	TYP.	MAX.	MIN.	TYP.	MAX.
A			3.6			0.142
a1	0.1		0.3	0.004		0.012
a2			3.3			0.130
a3	0		0.1	0.000		0.004
b	0.4		0.53	0.016		0.021
c	0.23		0.32	0.009		0.013
D (1)	15.8		16	0.622		0.630
D1	9.4		9.8	0.370		0.386
E	13.9		14.5	0.547		0.570
e		1.27			0.050	
e3		11.43			0.450	
E1 (1)	10.9		11.1	0.429		0.437
E2			2.9			0.114
E3	5.8		6.2	0.228		0.244
G	0		0.1	0.000		0.004
H	15.5		15.9	0.610		0.626
h			1.1			0.043
L	0.8		1.1	0.031		0.043
N	10' (max.)					
S	8' (max.)					
T		10			0.394	

(1) D and E' do not include mold flash or protrusions.  
 - Mold flash or protrusions shall not exceed 0.15 mm (0.006").  
 - Critical dimensions: "E", "G" and "a3"

**OUTLINE AND MECHANICAL DATA**



JEDEC MO-166

**PowerSO20**

