



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA

DEPARTAMENTO DE INFORMÁTICA DE SISTEMAS Y COMPUTADORES  
ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA DEL DISEÑO  
UNIVERSIDAD POLITÉCNICA DE VALENCIA

**DISEÑO, CONSTRUCCIÓN Y CONTROL POR ORDENADOR DE  
UN BRAZO ROBÓTICO**

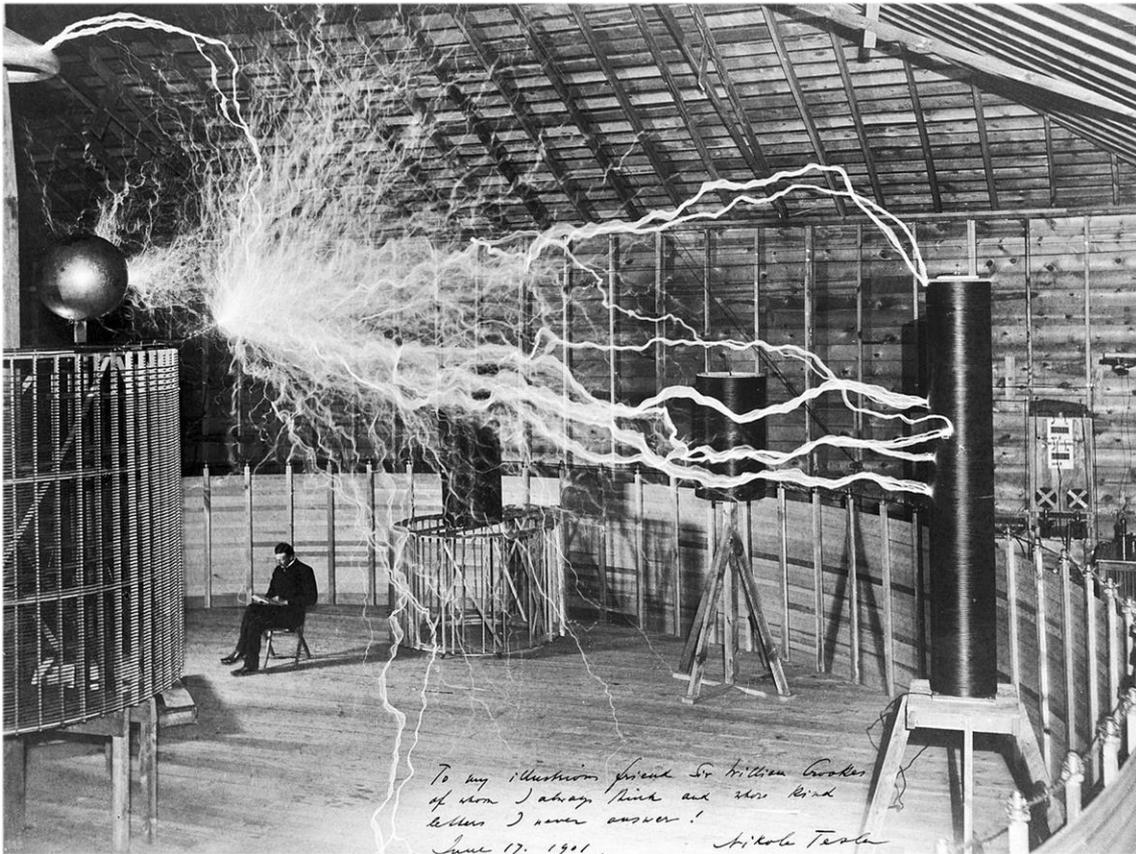
TRABAJO DE FIN DE GRADO  
GRADO EN INGENIERIA ELECTRONICA INDUSTRIAL Y AUTOMATICA

**Autor:** Marc García Abella

**Tutor:** Carlos Domínguez Montagud

Julio 2018





*"El desarrollo del hombre depende fundamentalmente de la invención. Es el producto más importante de su cerebro creativo"- Nicola Tesla*



## **Resumen**

El desarrollo y uso de los robots en el entorno industrial es cada vez mayor, existen diferentes tipos de robot de acuerdo con su función en la factoría. El proyecto consta del diseño, modelado, construcción y programación de un robot paletizador a escala, basándose en diseño de los ya existentes en el mercado.

El robot será destinado al uso docente por tanto será reproducible con impresión 3d y sus componentes son de bajo coste.

Posee cuatro ejes controlados por un servomotor y 3 motores paso a paso que serán manejados utilizando un Arduino UNO y que gestionará los sensores de posición. Posee una interfaz gráfica de usuario diseñada y programada en Labview que sirve para controlar y comunicarse con el robot.

Este proyecto supone la confluencia de los distintos conocimientos obtenidos a lo largo de la carrera y la experiencia.



## **Abstract**

The development and use of robots in the industrial environment is growing, there are different types of robots according to their function in the factory. The project consists of the design, modeling, construction and programming of a scale palletizing robot, based on the design of the existing ones in the market.

The robot will be destined to the teaching use therefore it will be reproducible with 3d printing and its components are of low cost.

It has four axes controlled by a servomotor and 3 stepper motors that will be managed using an Arduino UNO and that will manage the position sensors. It has a graphical user interface designed and programmed in Labview that serves to control and communicate with the robot.

This project supposes the confluence of the different knowledge obtained throughout the career and experience.



## **Resum**

El desenvolupament i ús dels robots en l'entorn industrial és cada vegada més gran, hi ha diferents tipus de robot d'acord amb la seua funció en la factoria. El projecte consta del disseny, modelatge, construcció i programació d'un robot paletitzador a escala, basant-se en disseny dels ja existents en el mercat.

El robot serà destinat a l'ús docent per tant serà reproducible amb impressió 3D i els seus components són de baix cost.

Posseeix quatre eixos controlats per un servomotor i 3 motors pas a pas que seran manejats per un Arduino UNO i que gestionarà els sensors de posició. Posseeix una interfície gràfica d'usuari dissenyada i programada en Labview que serveix per controlar i comunicar-se amb el robot.

Aquest projecte suposa la confluència dels diferents coneixements obtinguts al llarg de la carrera i l'experiència.



## ***Agradecimientos***

*Quiero agradecer el apoyo recibido por todas las personas que han estado a mi lado, tanto compañeros como familiares durante la carrera y la realización de este proyecto.*

*Gracias a mis padres por apoyarme sin poner límites a la realización de proyectos e inventos.*

*Agradecimiento en especial a mi yayo Pepe Abella y mi yaya Carmen Veintimilla que siempre creyeron en mí, que me inspiraron y supusieron mi constante motivación para crear inventos, iniciarme en mundo de la electrónica y cursar la carrera.*

*Siempre en el recuerdo.*



# Estructura del Proyecto

---

- I. MEMORIA DEL PROYECTO
- II. PLANOS DEL PROYECTO
- III. PRESUPUESTO Y ESTUDIO ECONÓMICO



# I. MEMORIA DEL PROYECTO

---



# Contenido

<b>Capítulo 1. Introducción</b> .....	23
1.1 Motivación del proyecto.....	23
1.2 Objetivo del proyecto.....	24
1.3 Especificaciones del diseño.....	24
<b>Capítulo 2. Introducción a la Robótica</b> .....	25
2.1 Historia de la robótica .....	25
2.2 Robots industriales .....	26
2.2.1 Características .....	26
2.2.2 Configuraciones .....	27
2.2.3 Clasificación según la configuración .....	28
2.2.4 Tipos de robot según su funcionalidad.....	29
2.2.5 Unidad de control.....	31
<b>Capítulo 3. Física del Robot</b> .....	31
3.1 Cinemática directa.....	31
3.2 Cinemática inversa .....	32
3.3 Cálculo trayectorias.....	33
3.4 Cálculo velocidad.....	34
3.5 Cálculo fuerza .....	34
<b>Capítulo 4. Diseño del robot</b> .....	35
4.1 Elección de los actuadores .....	35
4.2 Elección de Mecanismos.....	39
4.3 Elección de Sensores.....	40
4.4 Elección del Sistema de Control .....	43
4.5 Diseño de la estructura mecánica .....	50
4.5.1 Elección del modelo .....	50
4.5.2 Diseño 2D AutoCAD .....	50
4.5.3 Diseño 3D SolidWorks.....	53
4.5.3.2 Brazo, diseño detallado.....	59
4.5.3.3 Muñeca, diseño detallado .....	59
4.5.3.4 Engranajes, diseño detallado .....	61
4.5.3.5 Barras, diseño detallado .....	69

4.5.3.6 Base y laterales, diseño detallado.....	70
4.5.3.7 Piezas auxiliares, diseño detallado.....	72
4.5.3.8 Ensamblaje del modelo completo, diseño detallado.....	73
4.5.3.9 Cinemática del Robot diseñado.....	74
<b>Capítulo 5. Impresión 3d</b> .....	<b>78</b>
5.1 Proceso.....	78
5.2 Métodos.....	79
5.2.1 Estereolitografía.....	79
5.2.2 Modelado por deposición de fundente.....	81
5.3 Impresión de las piezas del robot.....	83
5.3.1 Impresoras elegidas.....	83
5.3.2 Material seleccionado.....	83
5.3.3 Exportación del modelo.....	84
5.3.4 Laminación de las piezas.....	85
<b>Capítulo 6. Software</b> .....	<b>90</b>
6.1 Funcionamiento LabVIEW.....	91
6.2 Programación del brazo robot.....	94
6.2.1 Bloque comunicación.....	94
6.2.2 Bloque control de los motores.....	96
6.2.3 Bloque configuración y estado de los motores.....	100
6.2.4 Bloque configuración inicial.....	102
6.2.5 Variables de sistema.....	104
6.2.6 Posición Home.....	104
<b>Capítulo 7. Pruebas y resultado final</b> .....	<b>105</b>
<b>Capítulo 8. Mejoras en el diseño</b> .....	<b>105</b>
<b>Capítulo 9. Conclusiones</b> .....	<b>106</b>
<b>Capítulo 9. Bibliografía</b> .....	<b>107</b>
<b>Capítulo 10. Anexos</b> .....	<b>108</b>

## Tabla de ilustraciones

Ilustración 1 Línea de montaje .....	23
Ilustración 2 Robots industriales en línea de montaje .....	23
Ilustración 3 Robot médico .....	25
Ilustración 4 Robot DIY .....	25
Ilustración 5 Hoja de datos y espacio de trabajo de robot IRB 140 .....	26
Ilustración 6 Distintos tipos de articulaciones y esquema de ejes de robot.....	27
Ilustración 7 Esquema robot cartesiano, CNC e Impresora 3d .....	28
Ilustración 8 Esquema robot cilíndrico y ejemplo de robot real .....	28
Ilustración 9 Esquema de robot esférico y ejemplo de robot real .....	28
Ilustración 10 Esquema de robot antropomórfico, similitudes con brazo humano y robot antropomórfico real .....	29
Ilustración 11 Esquema de robot SCARA y robot real SCARA .....	29
Ilustración 12 Robot FlexPicker de ABB.....	29
Ilustración 13 Robot manejador KUKA.....	30
Ilustración 14 Robots de pintura .....	30
Ilustración 15 Robot paletizador Fanuc .....	30
Ilustración 16 Armario Kuka con TP, Mando Flexpendant ABB .....	31
Ilustración 17 Ejemplo de robot con el algoritmo aplicado, Matriz D-H de datos obtenidos .....	32
Ilustración 18 Métodos de cinemática Directos e Inversos .....	32
Ilustración 19 Trayectoria Trapezoidal, Trayectoria perfil en S .....	33
Ilustración 20 Trayectoria S parcial, Trayectoria Senoidal .....	33
Ilustración 21 PWM y comportamiento del servomotor .....	35
Ilustración 22 Tower Pro 9G.....	36
Ilustración 23 Cuatro fases de rotación de un motor PAP .....	36
Ilustración 24 Motor nema 17, A4988, DVR825.....	37
Ilustración 25 Señales de entrada de los pines Step y Dir.....	37
Ilustración 26 Configuración de los pines MS1 MS2 MS3 para el A4988 .....	38
Ilustración 27 Esquema de conexión básico del driver .....	38
Ilustración 28 Funcionamiento interno del Harmonic Drive y sus elementos .....	39
Ilustración 29 Tuerca autoblocante .....	40
Ilustración 30 Rodamiento rígido de bolas .....	40
Ilustración 31 Principio de funcionamiento de un encoder absoluto y relativo .....	41
Ilustración 32 Esquema electrónico de un resolver y diagrama de funcionamiento .....	41
Ilustración 33 Diagrama de funcionamiento con partes del potenciómetro y esquema electrónico .....	42
Ilustración 34 Potenciómetro común y potenciómetro multivuelta .....	42
Ilustración 35 Tarjeta de adquisición de datos .....	43
Ilustración 36 Fotografía de Rapsberry Pi modelo 2B y logo de la compañía que la desarrolla	43
Ilustración 37 Placa de desarrollo Arduino UNO .....	45
Ilustración 38 Esquema de un conversor A/D.....	45
Ilustración 39 Proceso de cuantización .....	46
Ilustración 40 Arriba: señal con un correcto muestreo, Abajo: señal con aliasing .....	47
Ilustración 41 Esquema de conexión de los potenciómetros.....	47
Ilustración 42 Esquema de conexiones de la Shield CNC .....	48

Ilustración 43 CNC shield localizando los pines SCL, SDA y Coolant además de los pines de 5V, GND, el botón de reset y el borne de conexión de la alimentación .....	49
Ilustración 44 Conexión de pines del motor, conexión de driver de potencia y pines MS1, MS2 y MS3 .....	49
Ilustración 45 Modelo 2D KR 40 PA de KUKA abierto en AutoCAD .....	50
Ilustración 46 Modelo 2D del IRB 460 de ABB abierto en AutoCAD.....	51
Ilustración 47 Modelos de los robots Kuka y ABB con mayor detalle .....	51
Ilustración 48 Dibujo esquemático de la idea general respetando la geometría en sus elementos .....	52
Ilustración 49 Dibujo esquema de la base y logotipo del robot .....	52
Ilustración 50 Elementos que componen un robot .....	53
Ilustración 51 Guardar documentos AutoCAD en formato DWG .....	54
Ilustración 52 Importación del modelo DWG 2D de AutoCAD en SolidWorks .....	55
Ilustración 53 Croquis del modelo DWG importado .....	55
Ilustración 54 Selección del plano (verde) y cara de la pieza (rojo) para la extrusión.....	56
Ilustración 55 Simulación de la pieza tras introducir la distancia de extrusión .....	56
Ilustración 56 Diseño de croquis para los agujeros .....	57
Ilustración 57 Herramienta de corte para vaciar el agujero.....	57
Ilustración 58 Detalle del resultado final del corte.....	58
Ilustración 59 Diseño final del antebrazo.....	58
Ilustración 60 Diseño en del brazo del robot con detalle del logotipo .....	59
Ilustración 61 Piezas laterales izquierda y derecha, la derecha con separador .....	59
Ilustración 62 Separador cilíndrico .....	60
Ilustración 63 Zócalo para servomotor.....	60
Ilustración 64 Ensamblaje de la muñeca, la segunda imagen muestra alojado el servomotor 9g60	
Ilustración 65 Herramienta Toolbox con la elección de engranajes disponibles .....	61
Ilustración 66 Diseño de pedestal para diseñar los engranajes.....	63
Ilustración 67 Diseño de los engranajes en Toolbox y comprobación .....	64
Ilustración 68 Corona con diferente orientación, detalle del sistema de fijación con el potenciómetro.....	65
Ilustración 69 Detalle del sistema de fijación del potenciómetro.....	65
Ilustración 70 Detalle de fijación del sensor de posición.....	66
Ilustración 71 Engranaje del eje 2 con agujeros, y engranaje del eje 3 con barra .....	66
Ilustración 72 Piñón con diferentes orientaciones, detalle de fijación .....	67
Ilustración 73 Detalle del sistema de fijación piñón-eje motor .....	67
Ilustración 74 Corona de la base con sistema de fijación al potenciómetro.....	68
Ilustración 75 Detalle de sistema de fijación del potenciómetro multivuelta a la corona .....	68
Ilustración 76 Piñón de engranaje de la base con sistema de fijación al eje del motor .....	68
Ilustración 77 Barra de unión del enlace triangular con la muñeca .....	69
Ilustración 78 Enlace triangular alojado en la unión entre el antebrazo y brazo .....	69
Ilustración 79 Barra de unión entre el enlace triangular y la base.....	69
Ilustración 80 Barra de unión entre el extremo del antebrazo y el engranaje con barra .....	69
Ilustración 81 Base del robot, detalle de diseño de la parte superior .....	70
Ilustración 82 Base del robot, detalle de diseño de la parte inferior .....	70
Ilustración 83 Soporte del brazo.....	71
Ilustración 84 Lateral del robot .....	71
Ilustración 85 Izquierda potenciómetro normal, derecha potenciómetro multivuelta.....	72
Ilustración 86 Diseño motor nema 17 .....	72

Ilustración 87 Izquierda rodamiento P19, derecha rodamiento 623Z .....	72
Ilustración 88 Ensamble de la base del robot junto con el sistema de engranajes .....	73
Ilustración 89 Ensamble completo del robot.....	73
Ilustración 90 Esquema del robot don la distribución de los paralelogramos. Paralelogramo 1 verde. Paralelogramo 2 azul. Paralelogramo 3 rojo. ....	74
Ilustración 91 Ejes de Denavit-Hartenberg y cotas del robot.....	75
Ilustración 92 Detalle de la muñeca con los ejes de Denavit-Hartenberg y cota .....	75
Ilustración 93 1: Esfera original; 2,3,4: Esfera convertida en STL con orden creciente en calidad, orden                    descendiente en tamaño de las facetas triangulares .....	78
Ilustración 94 Funcionamiento y partes de una impresora estereolitográfica .....	80
Ilustración 95 Impresión 3D estereolitográfica .....	80
Ilustración 96 Esquema de deposición del fundente con sistema de coordenadas .....	81
Ilustración 97 Impresión de una pieza con salientes .....	82
Ilustración 98 Impresora RepRap Prusa i3.....	82
Ilustración 99 Hephestos BQ.....	83
Ilustración 100 Filamentos PLA de BQ y Z-Ultrat de Zotrax.....	83
Ilustración 101 Ventana de exportación STL en SolidWorks y esqueleto conformado por las facetas triangulares.....	84
Ilustración 102 Software Cura.....	85
Ilustración 103 Piezas impresas con diferentes alturas de capa con sus correspondientes tiempos de impresión y calidad final .....	86
Ilustración 104 Diferentes grosores de pared .....	86
Ilustración 105 Tipo de relleno y porcentaje de relleno.....	87
Ilustración 106 Patrón de calibración.....	87
Ilustración 107 Generación de soporte.....	88
Ilustración 108 Izquierda: pieza de PLA impresa sin refrigeración, Derecha: pieza con refrigeración .....	88
Ilustración 109 Diferentes tipos de adherencia a la plataforma de impresión.....	89
Ilustración 110 Ventana de creación de proyectos.....	91
Ilustración 111 Menú de controles y panel frontal con distintos controles .....	91
Ilustración 112 Ventana de instalación del Toolkit para Arduino.....	93
Ilustración 113 Ventana IDE de Arduino con firmware de LabVIEW .....	93
Ilustración 114 Interfaz gráfica de usuario del programa de control del robot .....	94
Ilustración 115 Detalle del bloque de comunicación .....	95
Ilustración 116 Sección de código correspondiente al inicio de la comunicación y cierre de la aplicación .....	95
Ilustración 117 Sección de código para la configuración inicial de la placa.....	96
Ilustración 118 Bloque de control de los motores.....	96
Ilustración 119 Bloque control del movimiento de los motores.....	97
Ilustración 120 Código C, movimiento de los motores.....	98
Ilustración 121 Bloque Movimiento de los motores .....	99
Ilustración 122 Bloque “Stepper write” en el motor PAP y bloque “Stepper Go To” .....	99
Ilustración 123 Bloque de configuración y estado de los motores, interfaz gráfica.....	100
Ilustración 124 Bloque VI habilitación de motores.....	100
Ilustración 125 Diagrama de bloques del sub-VI de habilitación de motores con los dos estados posibles (true/false) .....	100
Ilustración 126 Arriba: sub-VI con diferentes entradas; Abajo: configuració interna de bloques del sub-VI.....	101

Ilustración 127 Control de velocidades y aceleraciones .....	101
Ilustración 128 Bloque de calibración inicial en el GUI .....	102
Ilustración 129 Caja de verificación, configuración de bloques .....	102
Ilustración 130 Bloque de “Reset” .....	102
Ilustración 131 Bucle de adquisición de señal por el puerto analógico .....	103
Ilustración 132 Calibración inicial, sección de código dentro del Script de movimiento de los motores .....	103
Ilustración 133 Izquierda: variables locales; Derecha: variables globales .....	104
Ilustración 134 Botón de home, Variable global y Bucle de pasos para revertir a la posición de Home .....	104

# Capítulo 1. Introducción

## 1.1 Motivación del proyecto

La tecnología ha supuesto un cambio radical en el mundo tal y como lo conocemos, en cuestión de años hemos sido capaces de realizar tareas que se consideraban imposibles o muy complicadas gracias al uso de herramientas tecnológicas. Desde electrodomésticos o smartphones hasta software de cálculo, diseño o gestión.

Actualmente y cada vez más, la tecnología está en aumento, por tanto, se requiere mayor fabricación de elementos tecnológicos y a su vez que éstos sean más económicos. Entra en juego entonces la cadena de montaje cuyas bases estableció Henry Ford.

Hace alrededor de 100 años, Henry Ford consiguió reducir los costes de la producción en su fábrica de vehículos a las afueras de Detroit usando partes estandarizadas y un montaje más



Ilustración 1 Línea de montaje

eficiente basado en la especialización del operario en la realización de una tarea repetitiva en la construcción del coche. Este método le permitió multiplicar por diez la fabricación de automóviles. Con la introducción de los robots industriales y los PLCs se produjo una revolución industrial que permitió la automatización de las líneas de producción conocida como industria 3.0.

Los robots industriales son capaces de realizar las tareas repetitivas con mayor precisión, rapidez y fuerza que un operario convencional lo que permite abaratar costes y mejorar la calidad del producto final. Estas máquinas se asemejan al comportamiento de un brazo humano cuyas funciones dependen del diseño de estos o de la herramienta que lleven acoplada. Estos robots se mueven gracias a la acción de unos motores que proporcionan a las articulaciones el movimiento y la fuerza requeridas para ejecutar las órdenes.

La motivación principal de este proyecto surgió tras cursar la asignatura de 4º curso del grado denominada “Sistemas Robotizados”, en ella se aprendieron las bases del funcionamiento de los robots, las cinemáticas de éstos, el comportamiento físico, el trazado de trayectorias, el control de fuerza y control de articulaciones. En las practicas pude notar la falta de un modelo físico al que aplicar todos los controles diseñados por lo que es la principal base que sustenta la motivación del proyecto. Este proyecto plantea la accesibilidad de un modelo económico para que cada grupo de prácticas en el aula de robotizados tenga un robot industrial que pueda mover y programar, realizar experimentos y pruebas. Durante el periodo de prácticas en Ford Almussafes pude ver la magnitud de las instalaciones con robots y sus aplicaciones, además de manejar robots industriales Kuka KRC4 por lo que me sirvió de experiencia a la hora de enfocar este proyecto.



Ilustración 2 Robots industriales en línea de montaje

## 1.2 Objetivo del proyecto

El objetivo del presente trabajo de fin de grado es crear un brazo robot industrial a escala y compacto, que permita al alumnado desarrollar y aplicar las técnicas y fundamentos aprendidos en la robótica con un modelo real. El modelo final consta de un brazo robótico con actuadores y sensores y una aplicación diseñada en Labview como interfaz de control del robot.

Entre los objetivos principales destacamos:

- 1- Investigar el mercado y el mundo de la robótica industrial, extraer información para elegir el modelo que mejor se adapte.
- 2- Diseñar un modelo que se ajuste a las exigencias del diseño teniendo en cuenta todos los elementos que van a actuar.
- 3- Fabricar las piezas diseñadas y ensamblar el modelo con todos los actuadores, sensores y unidad de control
- 4- Programar una interfaz en un software que sea capaz de comunicarse con el robot leyendo sensores y gobernando sobre los actuadores indicando el estado geometría y situación del robot.

## 1.3 Especificaciones del diseño

El dispositivo no debe superar 50 cm de largo con una altura máxima de 40 cm, debe poder colocarse en una mesa de trabajo convencional.

El robot tendrá un máximo de cuatro ejes y su diseño se asemejará a un robot paletizador.

El diseño de las piezas se realizará teniendo en cuenta que van a ser impresas por impresoras 3d, utilizando materiales biodegradables.

Los materiales de tornillería y rodamientos a utilizar serán de bajo costo fáciles de obtener.

La electrónica que se va a utilizar no debe suponer un desarrollo muy complejo, deben realizarse conexiones sencillas. El sistema que controle el robot debe de poder conectarse al ordenador y presentar una interfaz sencilla, intuitiva y de fácil manejo.

Los motores del robot controlarán precisamente la posición del sistema, sin discontinuidades ni saltos bruscos. Se requiere un control de velocidad y aceleración.

El robot no debe suponer un peligro ni para las personas ni para el entorno por tanto los motores que se utilicen se podrán detener fácilmente con la mano careciendo de fuerza para dañar a un ser humano.

La programación se realizará con un software que permita el manejo del sistema con botoneras sencillas, dispondrá de información acerca del posicionamiento del robot en el espacio. El programa dispondrá de variables para controlar el comportamiento cinemático del robot. La programación debe ser sencilla e intuitiva.

## Capítulo 2. Introducción a la Robótica

### 2.1 Historia de la robótica

Durante muchos años el ser humano ha tratado de crear máquinas para facilitar el trabajo tedioso, muchas de estas máquinas tratan de imitar el comportamiento del cuerpo humano o sus extremidades.

El escritor checo *Karel Čapek* acuñó el término "robot" en 1923, a partir de la palabra de su lengua nativa *robot*, que significa servidumbre o trabajo forzado. Posteriormente *Isaac Asimov* a través de sus novelas asentó el término de robot, definiendo la ciencia que estudia a los robots y creando las tres leyes fundamentales de la robótica.

La robótica combina diversas disciplinas, entre las más fundamentales: la física, la mecánica, la electrónica, la ingeniería de control, la informática y la inteligencia artificial.

El mercado de la robótica da diferentes soluciones dependiendo del destino final de los robots, podemos crear una clasificación de acuerdo con su funcionamiento:

- **Robots industriales:** Conjunto de robots manipuladores automáticos, reprogramables y multifuncionales. Tienen una unidad de control que gestiona los movimientos y calcula trayectorias además de un control de entradas y salidas externas. Poseen al menos 3 ejes. Son utilizados en industrias de todo tipo ya que existe un amplio abanico de modelos dependiendo de la funcionalidad buscada. Son robustos y pueden llevar acoplados infinidad de herramientas, se sitúan de manera fija en un pedestal o acoplados sobre raíles mecánicos.
- **Robots médicos:** Podemos encontrar una gran variedad de este tipo de robots en los hospitales. Se utilizan para la realización de pruebas médicas y cirugía. Estos robots permiten técnicas menos invasivas y ejecuciones más precisas minimizando riesgos y tiempos de recuperación tras operaciones.
- **Robots domésticos o de servicio:** Engloba todos los dispositivos que desarrollan actividades del hogar en entornos no industriales, no son tan robustos como los industriales ni requieren de unidades de computación avanzadas. Suelen desempeñar tareas de limpieza del hogar.
- **Juguetes robóticos:** Existen de muchos tipos, enfocados al ocio y jóvenes que se inician en el mundo de la robótica. Está en auge el mundo de la robótica casera con la filosofía DIY que acerca a los usuarios módulos de programación y control a precios económicos.



Ilustración 3 Robot médico



Ilustración 4 Robot DIY

## 2.2 Robots industriales

### 2.2.1 Características

A la hora de diseñar o elegir un robot industrial se ha de tener en cuenta las diferentes características que permitan llevar a cabo la tarea que se le quiere asignar. Las dos principales características a tener en cuenta son:

- **Grados de libertad:** Es el número mínimo de variables necesarias para definir la posición del robot en el espacio y su efector final. Se conoce como efector final al extremo del robot, es la parte del robot que nos interesa ya que adquirirá la posición y orientación deseada. Las variables del espacio ( $X, Y, Z, \alpha, \beta, \gamma$ ) 6 en total permiten alcanzar cualquier posición (restringida en el espacio) y orientación. No obstante, pueden existir robots con grados de libertad adicionales que son redundantes o robots con menos grados de libertad donde la orientación no es un requisito.
- **Volumen de trabajo:** La estructura del robot fija la distancia en el espacio, el volumen de trabajo determina las posiciones en el espacio 3d las cuales un robot puede alcanzar. Según la función a la que van a ser destinados se buscan robots que cumplan los requisitos.

Las dos características anteriores y otras más como el peso, carga, velocidad, precisión etc. se reflejan en la hoja de datos del robot.

#### IRB 140

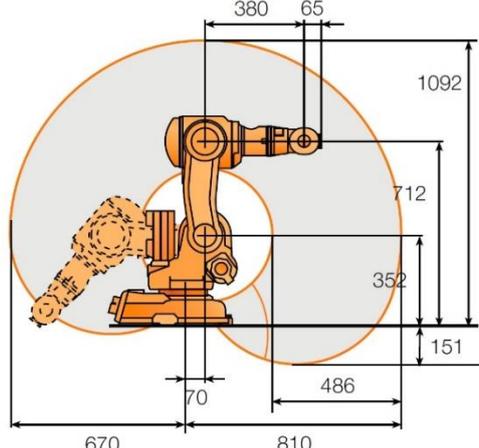
Specification				Electrical Connections		Working range	
Robot versions	Handling capacity	Reach of 5th axis	Remarks	Supply voltage	Rated power		
IRB 140/IRB 140T	6 kg	810 mm		200-600 V, 50/60 Hz	Transformer rating: 4.5 kVA Power consumption typically: 0.4 kW		
IRB 140F/IRB 140TF	6 kg	810 mm	Foundry Plus 2 Protection				
IRB 140CF/IRB 140TCR	6 kg	810 mm	Clean Room				
IRB 140W/IRB 140TW	6 kg	810 mm	SteamWash Protection				
Supplementary load (on upper arm alt. wrist)							
on upper arm				1 kg			
on wrist				0.5 kg			
Number of axes							
Robot manipulator				6			
External devices				6			
Integrated signal supply				12 signals on upper arm			
Integrated air supply				Max. 8 bar on upper arm			
IFCS Controller variants:				Single cabinet, Dual cabinet, Compact, Panel mounted			
Performance							
Position repeatability				0.03 mm (average result from ISO test)			
Axis movement				Axis		Working range	
				1		360°	
				2		200°	
				3		280°	
				4		Unlimited (400° default)	
				5		230°	
				6		Unlimited (800° default)	
Max. TCP velocity				2.5 m/s			
Max. TCP acceleration				20 m/s <sup>2</sup>			
Acceleration time 0-1 m/s				0.15 sec			
Velocity							
Axis no.				IRB 140		IRB 140T	
1				200°/s		250°/s	
2				200°/s		250°/s	
3				260°/s		260°/s	
4				360°/s		360°/s	
5				360°/s		360°/s	
6				450°/s		450°/s	
Environment							
Ambient temperature for				Robot manipulator: 5 - 45°C			
Relative humidity				Max. 95%			
Degree of protection,				Manipulator: IP67			
Options				Foundry Plus 2 SteamWash (High pressure steam washable) Clean Room, class 6 (certified by IPA)			
Noise level				Max. 70 dB (A)			
Safety				Double circuits with supervision, emergency stops and safety functions, 3-position enable device			
Emission				EMC/EMI-shielded			
Data and dimensions may be changed without notice							

Ilustración 5 Hoja de datos y espacio de trabajo de robot IRB 140

## 2.2.2 Configuraciones

Para lograr movimientos, los robots poseen distintos tipos de articulaciones, los ejes del robot se clasifican en primarios y secundarios, los primarios coinciden con los 3 primeros.

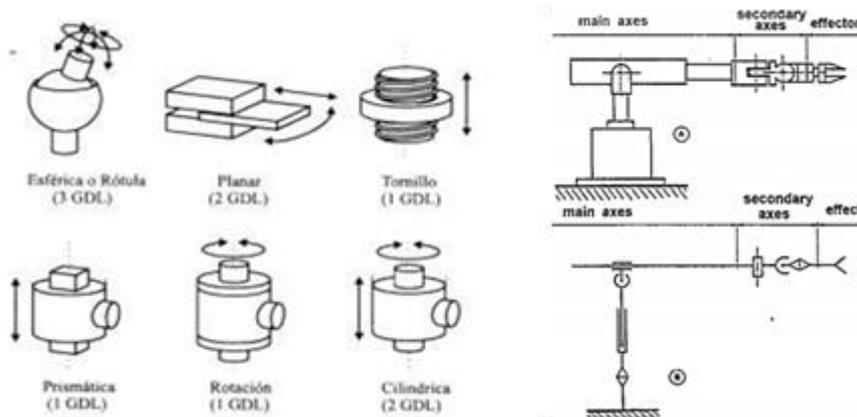


Ilustración 6 Distintos tipos de articulaciones y esquema de ejes de robot

De entre todas las articulaciones, las más utilizadas en robótica son la prismática y la de revolución, en la siguiente tabla se propone una comparativa:

PRISMÁTICA	REVOLUCIÓN
No tiene cinemática redundante	Cinemática redundante
Cinemática inversa trivial	Cinemática inversa no trivial
Menos deformación mecánica	Más deformación mecánica
Más pesada	Más ligera
Puede tener elementos de mayor longitud	Los elementos son de menor longitud
Mayor precisión	Menor precisión
No pierde resolución con longitud	Pierde resolución cuando aumenta la longitud
Dinámica simple: <ul style="list-style-type: none"> <li>▪ Inercia constante</li> <li>▪ Gravedad constante</li> <li>▪ No existen fuerzas centrífugas ni de Coriolis</li> </ul>	Dinámica compleja: <ul style="list-style-type: none"> <li>▪ Inercia variable dependiente del momento de inercia</li> <li>▪ Gravedad variable</li> <li>▪ Existen fuerzas centrífugas y de Coriolis</li> </ul>
Mayor longitud de los elementos para conseguir la misma área de trabajo $A_p = L_1 \cdot L_2$	Menor longitud de los elementos para conseguir la misma área de trabajo $A_r = \pi \cdot (L_1 + L_2)^2 - \pi \cdot (L_1 - L_2)^2 \rightarrow \text{Si } L_1 = L_2$ $A_r = \pi \cdot (L_1 + L_2)^2 = \pi \cdot L_1^2 + \pi \cdot L_2^2 + 2 \cdot \pi \cdot L_1 \cdot L_2$
Más cara	Más barata
3 GDL como máximo	6 GDL como máximo

### 2.2.3 Clasificación según la configuración

Para efectuar los movimientos, los robots industriales presentan distintas configuraciones que permiten desplazar cada uno de sus ejes, se pueden clasificar:

- **Robot cartesiano:** Es aquel robot que se mueve linealmente entre los 3 ejes de posición, dibuja líneas rectas en sus desplazamientos por lo tanto sus ejes (X, Y, Z) se encuentran dispuestos ortogonalmente. Normalmente el efector final se encuentra con una orientación fija salvo que se añada un eje adicional. Su función principal es dibujar planos, son rápidos y precisos, y fácil montaje. Sus mayores aplicaciones son el fresado en máquinas de corte numérico "CNC" e impresoras 3d.



Ilustración 7 Esquema robot cartesiano, CNC e Impresora 3d

- **Robot cilíndrico:** Poseen una base rotacional y dos articulaciones prismáticas perpendiculares entre sí, ha caído en desuso por tanto ya no se fabrican.

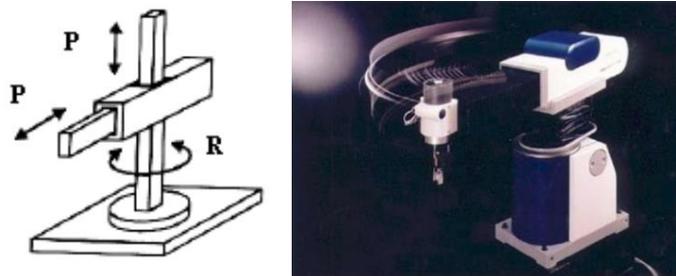


Ilustración 8 Esquema robot cilíndrico y ejemplo de robot real

- **Robot esférico:** Compuesto por 3 articulaciones, una articulación en la base rotacional, una articulación adicional de revolución en el codo, y una articulación prismática. De igual forma que el cilíndrico ya no se fabrican.

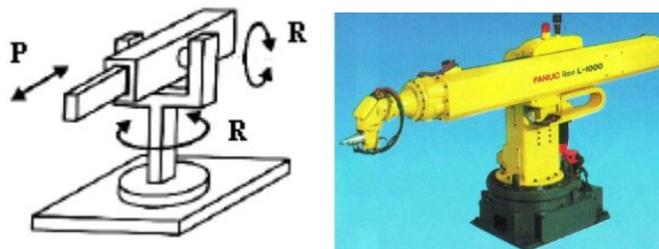


Ilustración 9 Esquema de robot esférico y ejemplo de robot real

- Robot articulado o antropomórfico: Simulan la articulación de un brazo humano. Las articulaciones de este robot son de revolución. Son multidisciplinarios y fáciles de adaptar a cualquier entorno industrial.

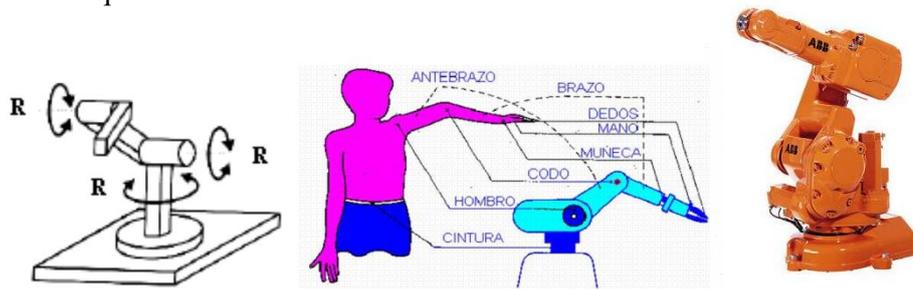


Ilustración 10 Esquema de robot antropomórfico, similitudes con brazo humano y robot antropomórfico real

- Robot SCARA: acrónimo de Selective Compliant Assembly Robot Arm, son robots con cuatro grados de libertad, diseñados para trabajar en un plano horizontal, compuestos por 3 articulaciones de revolución y una prismática. Son conocidos por su gran rapidez, precisión y repetitividad.

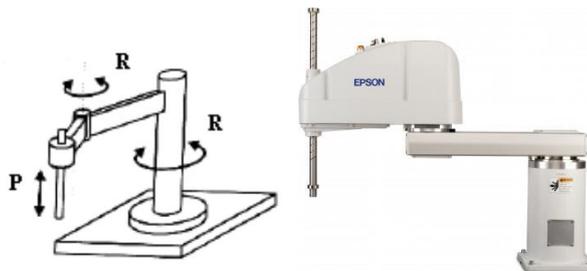


Ilustración 11 Esquema de robot SCARA y robot real SCARA

## 2.2.4 Tipos de robot según su funcionalidad

- Robots araña o clasificadores: robots rápidos para clasificar o empaquetar elementos pequeños. Empleados en la industria médica y alimenticia.



Ilustración 12 Robot FlexPicker de ABB

- Robots manejadores: robots multidisciplinarios adaptables a muchos entornos, permiten levantar cargas, soldar, atornillar, cortar, aplicar masilla... la función depende de la herramienta acoplada en su extremo. Son los robots industriales más utilizados en el mundo por su versatilidad.



*Ilustración 13 Robot manejador KUKA*

- Robots de pintura: orientados principalmente a la aplicación de pintura sobre superficies y piezas.



*Ilustración 14 Robots de pintura*

- Robots paletizadores: especializados en empaquetar y transportar cargas de gran peso. Poseen cuatro grados de libertad que le permiten posicionarse en el espacio y orientarse sobre el eje de la base. La configuración física del robot permite que el efector final siempre se encuentre paralelo al suelo. Nos centraremos en este modelo pues es el tipo de robot que trata de replicar el actual proyecto, descrito en las especificaciones técnicas.



*Ilustración 15 Robot paletizador Fanuc*

## 2.2.5 Unidad de control

El armario o unidad de control se encarga de gestionar la posición velocidad y aceleración del robot, permite trazar trayectorias, realizar programas, gestionar las seguridades y obtener el estado de entradas y salidas. El mando del robot es un reflejo del armario, permite al operario acceder de forma sencilla a la visualización y modificación de parámetros del robot. La programación actual de un robot se efectúa en la mayoría de los casos con el mando, donde el operario mueve a través del Joystick el robot, y posteriormente guarda cada uno de los puntos geométricos en que se encuentra el robot. Cuando se ejecuta la rutina del robot, este recorre cada uno de los puntos anteriormente señalados trazando trayectorias previamente especificadas.



Ilustración 16 Armario Kuka con TP, Mando Flexpendant ABB

# Capítulo 3. Física del Robot

## 3.1 Cinemática directa

La cinemática del robot permite el estudio de los movimientos de éste. Las variables a considerar son la posición, velocidad y aceleración de los elementos del robot sin tener en cuenta las fuerzas que actúan en él. La cinemática directa permite a través del uso de ecuaciones matemáticas obtener la posición y orientación del extremo del robot (también llamado actuador final) o efector final a partir de las posiciones individuales de cada una de las articulaciones. Se ha de conocer siempre los parámetros geométricos de los elementos del robot, los valores de las articulaciones y siempre debe estar asociado a un sistema de coordenadas de referencia.

Para resolver el problema cinemático directo se pueden utilizar dos métodos:

- **Método geométrico:** permite obtener la posición y orientación del extremo del robot utilizando relaciones geométricas, es un método no sistemático, y se complica cuando existen muchos grados de libertad.

- Método Denavit-Hartenberg:** Se utiliza para describir la relación que existe entre dos sistemas de referencia asociados a eslabones. Denavit y Hartenberg propusieron en 1955 un método matricial que permite obtener un sistema de coordenadas  $\{S_i\}$  asociado a cada eslabón  $i$  de una cadena articulada de manera sistemática. Además, esta representación permite pasar de un sistema de coordenadas a otro mediante 4 transformaciones básicas que dependen exclusivamente de las características geométricas del eslabón. Las transformaciones básicas que relacionan el sistema de referencia del elemento  $i$  con el elemento  $i-1$  son las siguientes:

- 1-Rotación  $\theta_i$  alrededor del eje  $Z_{i-1}$
- 2-Traslación  $d_i$  a lo largo del eje  $Z_{i-1}$
- 3-Traslación  $a_i$  a lo largo del eje  $X_i$
- 4-Rotación  $\alpha_i$  alrededor del eje  $X_i$

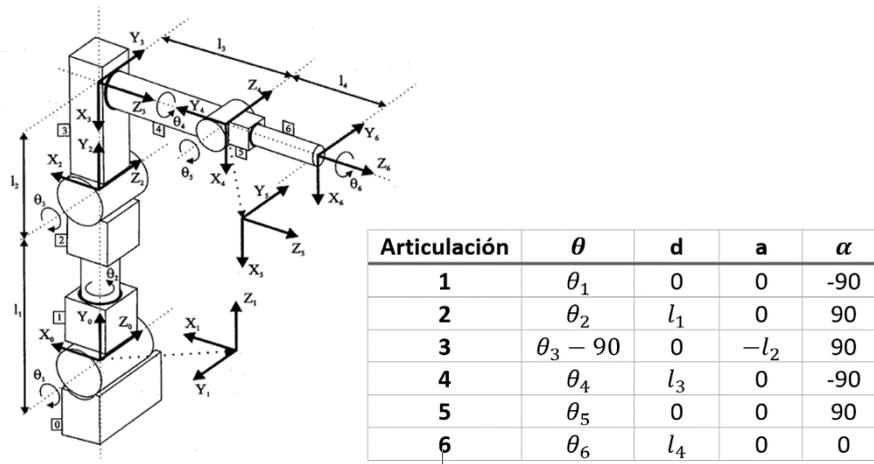


Ilustración 17 Ejemplo de robot con el algoritmo aplicado, Matriz D-H de datos obtenidos

### 3.2 Cinemática inversa

Permite encontrar las coordenadas de cada articulación y su posición a partir de la posición y orientación del efector final. El método es redundante, para una posición final existen diferentes soluciones. Los métodos genéricos suelen tener problemas de cálculo en tiempo real por lo que en la práctica se pueden utilizar distintas formas de solución para diferentes robots. Como método para facilitar cálculo se suelen desacoplar los tres primeros ejes de los tres últimos, ajustando la posición con los tres primeros, y la orientación con los tres últimos.

Ambos dos métodos son complementarios:

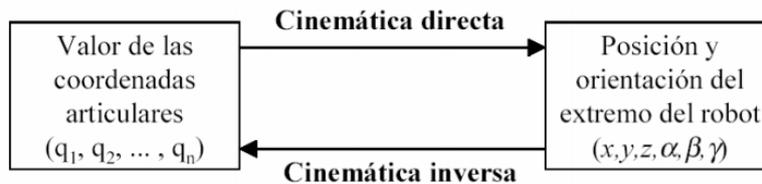


Ilustración 18 Métodos de cinemática Directos e Inversos

### 3.3 Cálculo trayectorias

La función principal es obtener trayectorias físicamente realizables para ejecutar los comandos o posiciones requeridas. Permite calcular valores ideales de posición y velocidad para cada valor de muestreo.

- Interpolación lineal: modelo más sencillo de conseguir alcanzar posiciones, el problema que presenta es que es un sistema brusco porque no presenta continuidad de movimiento. Alcanza cada uno de los puntos programados.
- Interpolación cúbica: permite redondear las esquinas presentes en la interpolación lineal, es un sistema suave, no ocasiona cambios bruscos en movimientos, no alcanza todos los puntos programados, pero se acerca a ellos.

En el trazado de trayectorias se deben evitar discontinuidades pues estas suponen derivadas infinitas lo que provocan tirones y los elementos mecánicos y actuadores sufren más, por tanto, comporta un desgaste mayor.

A la hora de escoger un modelo de trazado de trayectoria se debe tener en cuenta la potencia de cálculo computacional pues modelos con comportamientos menos abruptos requieren de un mayor cálculo y por tanto mayor coste computacional. Entre los modelos existentes destacamos los más utilizados:

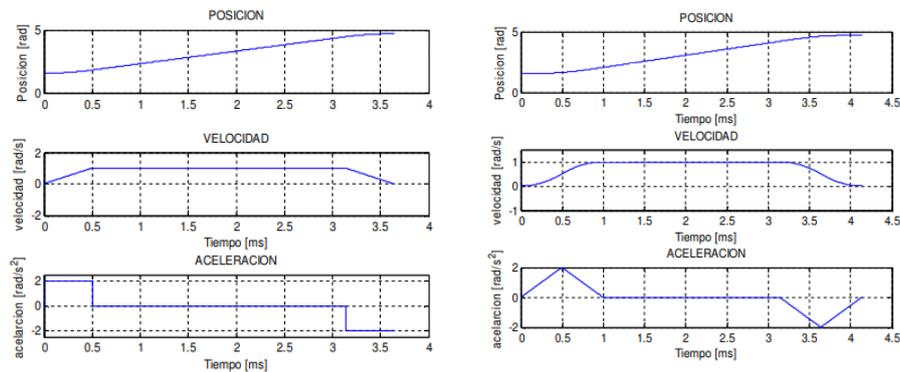


Ilustración 19 Trayectoria Trapezoidal, Trayectoria perfil en S

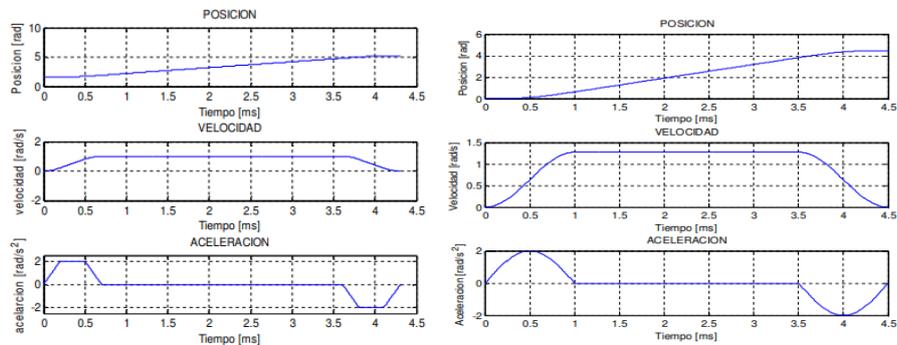


Ilustración 20 Trayectoria S parcial, Trayectoria Senoidal

### 3.4 Cálculo velocidad

La cinemática de velocidades busca la relación existente entre las velocidades de las articulaciones y la velocidad del efector final del robot. La velocidad del extremo se descompone en velocidad lineal y angular. Para que se cumpla la velocidad deseada en el extremo del robot, la cinemática de velocidades calcula la velocidad de cada una de las articulaciones.

La velocidad angular será:

– En articulación prismática:  $w_i^{i-1} = 0$

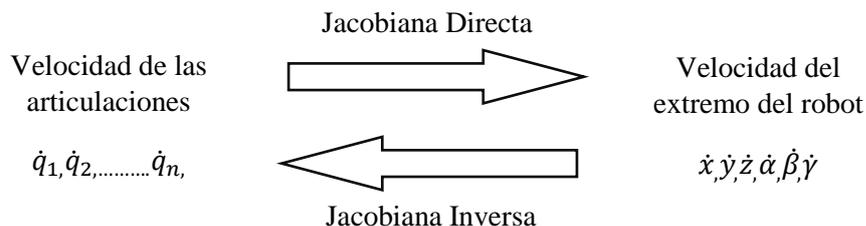
– En articulación de revolución:  $w_i^{i-1} = \dot{q}_i \times \vec{z}_i$

La velocidad lineal será:

– En articulación prismática:  $v_i^{i-1} = \dot{q}_i$

– En articulación de revolución:  $v_i^{i-1} = (\dot{q}_i \times \vec{z}_i) \times \vec{l}_i$

A partir del modelo conocido se puede derivar para obtener la relación existente entre las velocidades de las articulaciones y el extremo del robot y viceversa. Se recurre al uso de la matriz Jacobiana.



### 3.5 Cálculo fuerza

El robot entra en contacto con el entorno por tanto puede hacer daño al propio entorno y a si mismo por lo que se debe diseñar un sistema que controle la fuerza del robot:

- Control explícito: se adquiere información directamente de los sensores de fuerza de las articulaciones, los reguladores más utilizados son los PID, PID con filtro, PIV y PIV con prealimentación. Existen problemas en la utilización de este método como el ruido en el sensor de fuerza. Se ha de tener en cuenta que los robots están diseñados para un control por posición.
- Control implícito: no controla directamente la fuerza sino el comportamiento dinámico del sistema por tanto evita problemas de medida en sensores de fuerza y se aprovecha el control por posición diseñado para el robot.

## Capítulo 4. Diseño del robot

### 4.1 Elección de los actuadores

El modelo requiere de actuadores que permitan mover las articulaciones. Se buscarán alternativas para modelos a pequeña escala relacionados con aeromodelismo o DIY.

A continuación, se exponen diferentes alternativas y la solución tomada:

- **Motores sin escobillas (Brushless):** es un tipo de motor eléctrico que no emplea escobillas para el cambio de polarización, la ausencia de estas escobillas permite un aumento de rendimiento y disminuye el calor y ruido generados. Se controlan de manera trifásica. Ampliamente usados en drones. Se descarta esta opción puesto que el consumo eléctrico es elevado, posee un sistema electrónico complejo y funciona a mayores revoluciones de las requeridas.
- **Servomotores:** son dispositivos formados por un motor de corriente continua y una unidad de control. Los servos son capaces de ubicarse en una posición concreta dentro del rango establecido de operación y pueden mantenerse estables en dicha posición gracias a un bucle de realimentación que se encarga de verificar que el motor se encuentre en la posición deseada. El control de los servomotores de aeromodelismo o hobby se realiza mediante PWM. El PWM o modulación por ancho de pulso es la técnica que modifica el ciclo de trabajo de una señal periódica. Según el ciclo de trabajo de la señal, normalmente cuadrada porque es generada por microcontroladores, el servo adquiere una posición concreta.

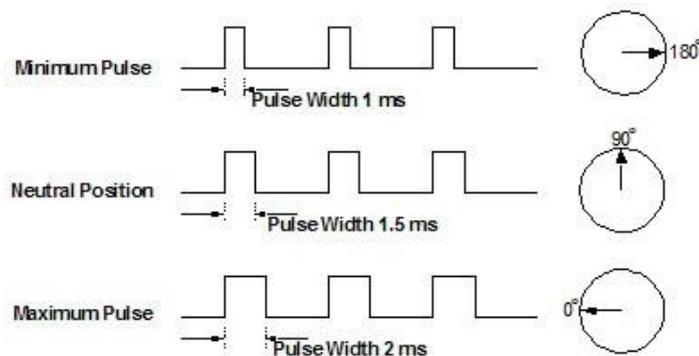


Ilustración 21 PWM y comportamiento del servomotor

Existen tres problemáticas con el uso de servomotores. La primera, al ser un bucle cerrado no se puede controlar ni medir la fuerza, no se puede crear un control externo porque ya tiene uno implícito. La segunda, el control de posición limita la velocidad, el servomotor trabaja a una velocidad fija y si se pretende trazar trayectorias el sistema presenta discontinuidades en los incrementos de posición. El último motivo es la precisión, en el

sistema de servomotores existe el parámetro de banda muerta, este determina el cambio mínimo del ciclo de trabajo para que este se vea reflejado en el movimiento del motor, lo que ocasiona que cambios mínimos de posición requeridos no se reflejen en el actuador. Usaremos un único servomotor para mover el eje 4 debido principalmente al peso y tamaño reducido.

El servomotor que utilizaremos será el “Tower Pro 9g” se trata de un servomotor de aeromodelismo de tamaño compacto y 9 gramos de peso.



Ilustración 22 Tower Pro 9G

- Motores PAP (paso a paso): es un motor que se controla mediante una serie de impulsos eléctricos, estos impulsos provocan desplazamientos angulares discretos, por lo tanto, el motor gira un número de grados asociados a la cantidad de desplazamientos o pasos. Este motor presenta las ventajas de precisión y repetitividad en cuanto al posicionamiento.
  - Motores unipolares: Simples de controlar, compuestos por 5 o 6 cables uno de los cuales es común a la fuente de alimentación y el resto se conectan al dispositivo que controla el giro.
  - Motores bipolares: Mayor complejidad de control, 4 cables que controlan dos bobinas, estos cuatro cables son conectados a la unidad de control y el flujo de corriente a través de las bobinas las magnetiza y orienta los imanes del rotor, si se controla el flujo de corriente en una secuencia concreta se genera el movimiento.

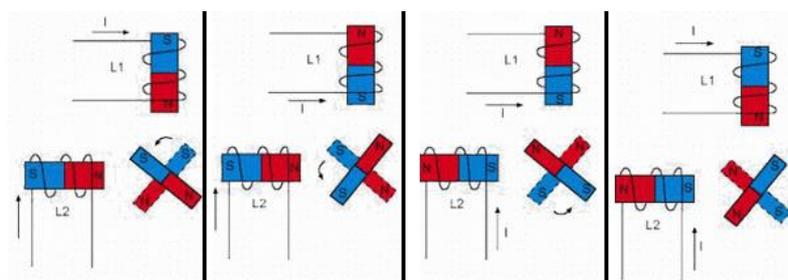


Ilustración 23 Cuatro fases de rotación de un motor PAP

Estos motores tienen la característica de poder fijar la posición, es decir si se suministra una corriente de mantenimiento el motor genera fuerza para evitar que este se desplace y cambie de posición.

Se presenta como la solución adoptada ya que cumple las especificaciones del diseño de movimiento del robot, es preciso, permite crear un sistema de control, aplicar cambios de velocidad y puede mantenerse estático.

Entre las posibilidades que oferta el mercado se ha optado por el motor nema 17 que cumple los requisitos anteriormente descritos. Posee una resolución de  $1.8^\circ$  por paso lo que se transforma en 400 pasos para obtener una vuelta completa.

- Driver de motores:

El motor se controlará mediante un driver de motores, se trata de un puente en H que genera la secuencia correcta para el movimiento del motor en el sentido y posición deseados además de la corriente necesaria para moverse. Existen drivers especializados para el control de motores paso a paso bipolares como el A4988 o el DVR8825



Ilustración 24 Motor nema 17, A4988, DVR825

Los drivers A4988 o el DVR8825 de la marca Pololu se controlan principalmente mediante dos pines, el pin “Step” recibe la cantidad de pulsos que debe mover el motor y “Dir” controla la dirección de este. Esta configuración simplifica la conexión con un microcontrolador.

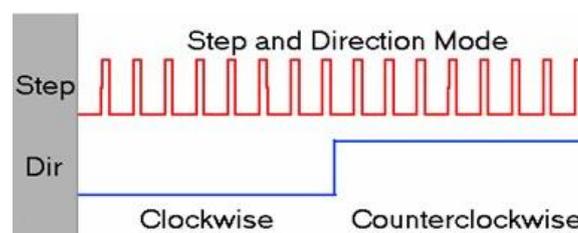


Ilustración 25 Señales de entrada de los pines Step y Dir

Posee además un pin de “Enable” que bloquea el motor o permite el libre movimiento de este, este pin se encuentra normalmente en estado alto. Los pines “MS1, MS2, MS3” configuran los pasos del motor. La configuración de estos pines permite aumentar la resolución del movimiento del motor permitiendo moverlo medios pasos, cuartos de paso, octavos y dieciseisavos de paso.

Para activar las configuraciones de micro-pasos se deben puentear las señales “MS1, MS2 y MS3”, normalmente se encuentran en estado bajo o “low” por lo que siguiendo la tabla siguiente de configuraciones se obtiene el modo excitación deseado:

MS1	MS2	MS3	Microstep Resolution	Excitation Mode
L	L	L	Full Step	2 Phase
H	L	L	Half Step	1-2 Phase
L	H	L	Quarter Step	W1-2 Phase
H	H	L	Eighth Step	2W1-2 Phase
H	H	H	Sixteenth Step	4W1-2 Phase

Ilustración 26 Configuración de los pines MS1 MS2 MS3 para el A4988

L = low (estado bajo) y H = high (estado alto) cada uno de los pines se debe conectar a tierra en caso de estado bajo o a Vdd (Voltaje de alimentación del driver) si se quiere en estado alto. Los drivers “Pololu” llevan integrado un potenciómetro cuya misión es la de regular la cantidad de corriente suministrada al motor, esta corriente refleja la fuerza que el motor es capaz de mover. Según el fabricante “Pololu” para la serie nema 17 los valores óptimos de trabajo se encuentran ente 200 y 600 mA de corriente. Debido a la potencia que estos drivers manejan se calientan por tanto se debe efectuar una correcta disipación del calor generado añadiendo disipadores pasivos de aluminio. La disipación ayuda al correcto funcionamiento del chip ya que si la temperatura es extrema este puede reiniciarse, perder la configuración, movimiento y en un caso extremo destruirse.

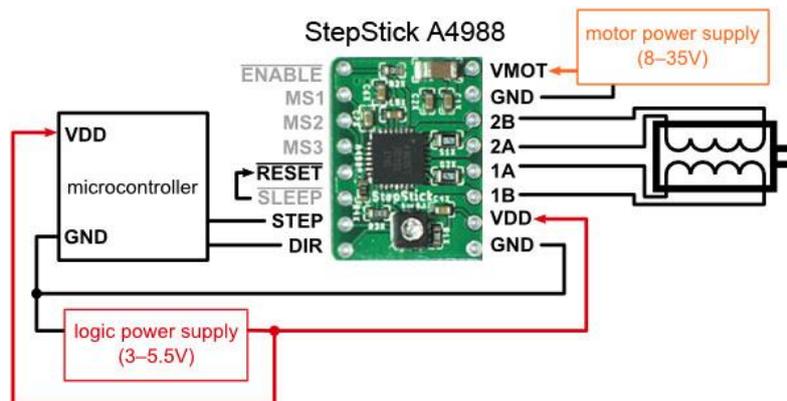


Ilustración 27 Esquema de conexión básico del driver

La principal diferencia que presentan los dos modelos de drivers es que el DVR8825 es capaz de suministrar mayor corriente por tanto tiene mayor potencia que el A4988. El primero concretamente es capaz de suministrar 2.5 A frente a los 1.5 A del A4988. A su vez el DVR8825 es capaz de trabajar en un rango de excitación mayor, pues sus configuraciones de micro-pasos llegan hasta el trigésimo segundo de paso. Se elegirá el DVR8825 por sus prestaciones de potencia y precisión.

## 4.2 Elección de Mecanismos

Se deben insertar sistemas mecánicos que permitan transmitir el movimiento, aumentar la fuerza y reducir la velocidad del giro de los motores. Entre los sistemas de mecanismos más utilizados en la robótica antropomórfica destacamos los siguientes:

Correas o cadenas: Sistemas que sirven para transmitir movimientos de fuerza entre ruedas dentadas. Estos sistemas son rápidos, respetan el sentido de giro. El problema que presentan es la elasticidad, posibles desplazamientos en el caso de las correas y tienen un menor par frente a otros sistemas. Se descarta esta opción porque hablamos de un modelo pequeño y el mercado no ofrece correas ni cadenas para modelos reducidos.

Engranajes: Sistemas mecánicos que se utilizan para transmitir potencia mecánica de un componente a otro. Formados por dos ruedas dentadas denominadas piñón (la menor) y corona (la mayor). El movimiento se transmite circularmente mediante el contacto con las ruedas dentadas. El sentido del movimiento se invierte. Este sistema permite reducir velocidad a la par que se aumenta la fuerza y viceversa. Son sistemas poco elásticos con un gran par de fuerza. Será la opción elegida pues se adapta a las necesidades del diseño. Estos elementos pueden ser impresos con una impresora 3d.

- Diseño de los engranajes: se eligen de entre todos los tipos de engranajes, los engranajes cilíndricos de dientes rectos pues la transmisión se realiza entre ejes paralelos. La velocidad máxima del motor nema 17 es de 1800 RPM y su par de fuerza es de 15 mNm. Con estos datos podemos determinar que no se precisa una velocidad tan elevada por tanto se va a diseñar un sistema de engranajes de 4:1 que reduzca la velocidad cuatro veces. A su vez aumentará cuatro veces su par de fuerza. Más adelante en el apartado de diseño 3D se especifica detalladamente el diseño de este sistema.

Harmonic drive: Se requiere una mención especial de este sistema que es ampliamente utilizado en los sistemas robóticos, se trata de un mecanismo que combina el funcionamiento de engranaje con la teoría de la onda de deformación. Se basa en la dinámica elástica de uno de sus elementos utilizando la flexibilidad del metal, junto con un generador de onda y un engranaje circular. Esto permite un sistema de transmisión de potencia capaz de conseguir grandes relaciones de transmisión en una configuración que supone un peso y volumen bajo.

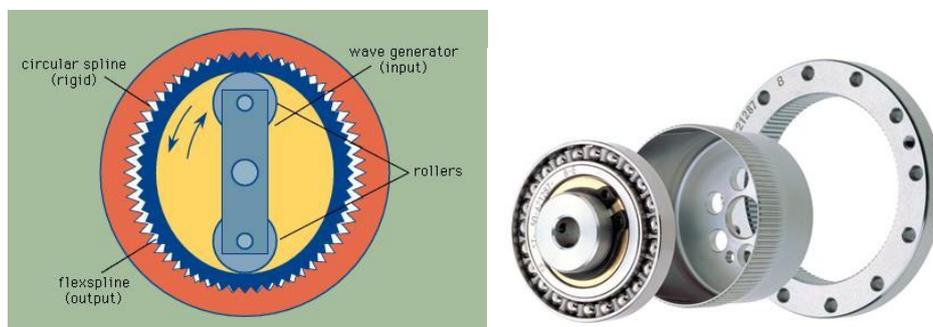


Ilustración 28 Funcionamiento interno del Harmonic Drive y sus elementos

**Tornillos:** Son elementos mecánicos utilizados para la fijación temporal de piezas. Están dotados de un elemento circular con rosca triangular que mediante fuerza de torsión se introduce en agujeros roscados, traviesa piezas o se acopla a tuercas. De entre las opciones que ofrece el mercado se va a utilizar tornillos de métrica 3 o M3, de diferentes longitudes. La cabeza del tornillo es indiferente se elegirá la opción más económica.

**Tuercas:** pieza mecánica con un orificio central el cual presenta rosca, se utiliza para enroscar a un tornillo, y permite fijar piezas de elementos desmontables. Se utilizarán tuercas normales y autoblocantes, las ultimas permiten fijar piezas sujetas a vibraciones o movimientos gracias un freno de nailon insertado que hace que no se muevan ni se aflojen por lo que resultan muy útiles como topes. La serie a utilizar será la misma que la de los tornillos, métrica 3.

**Rodamientos:** este elemento mecánico tiene la función de reducir la fricción entre los ejes y las piezas asociadas a estos ejes por medio de rodaduras. Los rodamientos sirven de apoyo y facilitan los desplazamientos. Se utilizarán rodamientos 623Z que casan con la métrica 3 de los tornillos y rodamientos P19 para la base del robot.



Ilustración 29 Tuerca autoblocante



Ilustración 30 Rodamiento rígido de bolas

### 4.3 Elección de Sensores

Los sensores permiten extraer información del entorno, obtener una magnitud del medio que varía y convertirla en otra magnitud física con tal de cuantificar la variación dicha magnitud. En el proyecto se precisan de sensores de desplazamiento, en este caso desplazamiento angular para poder conocer la posición real de cada uno de los elementos del robot. A continuación, estudiaremos las diferentes alternativas:

**Encoders o codificador rotatorio:** se trata de un dispositivo electromecánico capaz de convertir la posición angular de un eje en un código digital. Los hay de dos tipos absoluto e incremental (relativo).

- Encoder absoluto: genera un código digital que es único para cada ángulo del eje. Se basa en un patrón complejo dispuesto en una hoja de metal, asociado al eje encontramos un disco aislador con contactos. Según la posición del eje, el disco aislador recorre el disco de metal y según la disposición de los contactos sobre el disco de metal genera una señal de salida informando de la posición angular.

Existe también una versión óptica, esta tiene el mismo principio de funcionamiento solo que el disco no es de metal es de un material transparente y el patrón se encuentra dibujado en negro, a su vez tiene una serie de transmisores y fotorreceptores enfrentados a través del disco con el patrón, según la posición del eje, el disco adquiere una configuración dejando pasar la luz en ciertas áreas, los fotorreceptores tienen una respuesta codificada indicando el estado del Angulo.

- El encoder rotatorio relativo o incremental: se asemeja al encoder absoluto solo que, con un funcionamiento mucho más simple, puede ser de metal u óptico. Este dispositivo informa solo de la dirección y sentido de giro, pero no de una posición absoluta. Para obtener una posición absoluta del sistema con este dispositivo se requiere conocer el posicionamiento inicial del eje. Se utiliza cuando el sistema absoluto es muy grande e incómodo debido al tamaño del sensor. Al ser un sistema mucho más simple su tamaño es muy reducido.

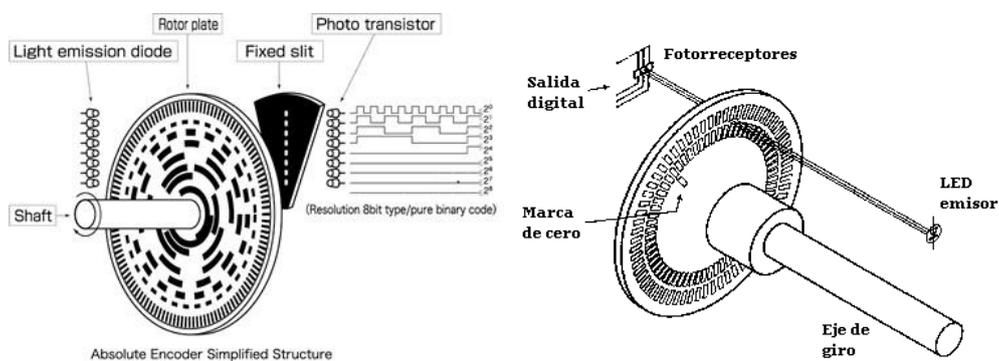


Ilustración 31 Principio de funcionamiento de un encoder absoluto y relativo

Aunque son sensores muy precisos e idóneos para aplicaciones robóticas se descarta su uso debido al coste económico que suponen.

Resolver: es transformador eléctrico rotativo que permite medir los grados de precisión de acuerdo con las formas de onda que se generan en sus bobinas. El interior de un resolver se asemeja al de un motor sin escobillas, consta de una bobina de excitación asociada al eje de giro y dos devanados de dos fases. El giro en el eje provoca una inducción de corriente desde la bobina de excitación a las bobinas de cada una de las fases, estas fases generan una señal de acuerdo al giro, dotando de información acerca de la dirección y sentido de giro y así obteniendo la posición del eje. Se descarta esta opción debido a la complejidad del circuito conversor y el precio del sensor.

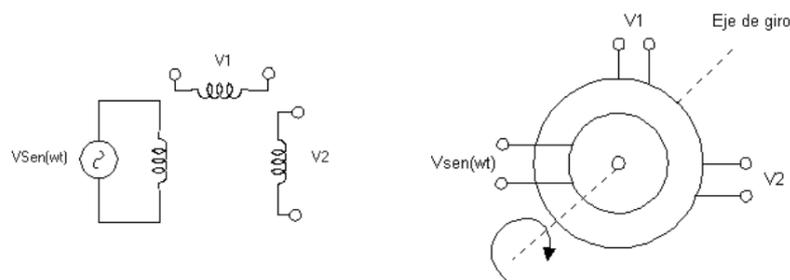


Ilustración 32 Esquema electrónico de un resolver y diagrama de funcionamiento

**Potenciómetros:** son resistencias variables. El dispositivo lo conforman tres terminales, al manipular el potenciómetro se modifica el cursor de éste, recorriendo una superficie resistiva, se comporta formando un divisor resistivo variable. El valor nominal del potenciómetro indica la resistencia en serie equivalente. El dispositivo puramente se utiliza en electrónica para obtener un divisor de tensión, pero entre otras aplicaciones se puede obtener un sensado de la posición, midiendo la tensión equivalente en el cursor a partir de la variación de su eje, cuando sus terminales extremos se encuentran conectados a una diferencia potencial conocida. Existen dos tipos de potenciómetros, lineales o angulares. A su vez según el comportamiento de la unidad resistiva existen lineales, logarítmicos (usados en audio) o anti-logarítmicos.

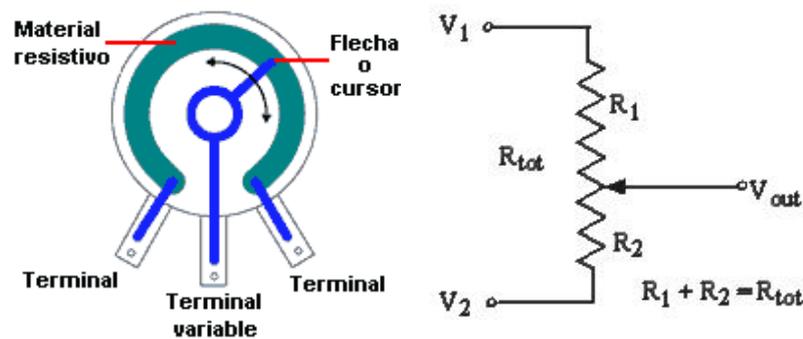


Ilustración 33 Diagrama de funcionamiento con partes del potenciómetro y esquema electrónico

El potenciómetro tiene un recorrido que no completa una vuelta completa, aproximadamente el cursor recorre 300° a lo largo de su recorrido entre ambos terminales extremos. Existe un tipo de potenciómetro llamado multivuelta, este es capaz de girar varias veces.

Optaremos por un potenciómetro angular para el sensado de la posición. Son elementos económicos y sencillos y pese a que la estabilidad térmica, el ruido eléctrico, y la velocidad limitada son inconvenientes a tener en cuenta, cumple los requisitos para un proyecto a pequeña escala como es el caso del presente brazo robótico. Como los brazos del robot no giran completamente con dos potenciómetros comunes será suficiente, en cambio la base requiere de giros completos por tanto se utilizará un potenciómetro multivuelta. El valor nominal de los tres potenciómetros será de 10K Ohmios.



Ilustración 34 Potenciómetro común y potenciómetro multivuelta

## 4.4 Elección del Sistema de Control

El sistema debe tener una unidad de control que gestione el movimiento de los motores, adquiera información de los sensores y que le permita comunicarse con un ordenador. La unidad de control junto con el ordenador se asemejará al armario y el mando del robot. Destacamos una serie de opciones a continuación:

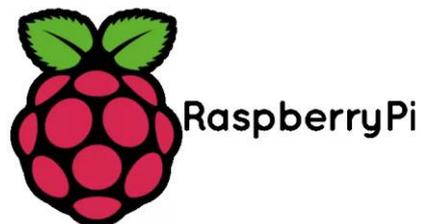
Tarjeta adquisición de datos: es un dispositivo que se comunica directamente con el ordenador, no tiene un sistema de memoria en sí, únicamente se trata de una interfaz de comunicación con el exterior. Existen programas de muchos tipos que se comunican con dichas interfaces y son capaces de enviar información al exterior en forma de voltaje o extraer información de sensores mediante sus unidades analógico/digitales. Se descarta esta opción pues supone un alto coste económico y se requiere de una unidad que pueda comportarse automáticamente con independencia de un ordenador para futuras aplicaciones o mejoras.



*Ilustración 35 Tarjeta de adquisición de datos*

Raspberry Pi: es un ordenador dispuesto en una placa reducida de bajo coste desarrollado en Reino Unido para estimular y desarrollar la enseñanza. No se indica expresamente que se trate de hardware libre, pero se permite el uso libre tanto educativa como particularmente. El software en cambio sí es libre (open source) y permite la instalación de diferentes sistemas operativos.

Aunque es una solución válida, no se requiere de una unidad que permita tanto poder computacional por lo que se elegirá una alternativa más económica.



*Ilustración 36 Fotografía de Raspberry Pi modelo 2B y logo de la compañía que la desarrolla*

Arduino: se trata de una compañía con filosofía open source y hardware, engloba un gran proyecto y una comunidad internacional. La compañía diseña y manufactura placas de desarrollo de hardware. Facilita el uso de la electrónica y la acerca a los usuarios menos expertos. Los productos que crea y ofrece la compañía son de Hardware y Software Libre, lo que permite el uso, la réplica y la distribución por cualquier individuo. Las placas que vende la compañía están disponibles en forma de placas ensambladas o en forma de DIY, (sus siglas en inglés de *Do It Yourself*). Existe un amplio mercado con placas de expansión y periféricos. Las placas de expansión o shields disponen de una configuración con pines exactamente idénticos a los de Arduino lo que facilita su conexión y compatibilidad.

Las placas Arduino usan diversos microcontroladores y microprocesadores. Por regla general los microchips usados son los Atmel AVR. La placa a su vez dispone de todo lo necesario para que el microcontrolador funcione y pueda comunicarse con el exterior como: USB, pines hembra conexionado, reloj de cuarzo, leds indicativos y circuitería adicional para su correcto funcionamiento.

Existen diferentes tipos de placas de acuerdo con su tamaño y potencia, nos centramos en Arduino UNO:

- Arduino UNO: es una placa de desarrollo con un microcontrolador ATmega328P como cerebro. Tiene un total de 14 pines digitales que se pueden utilizar como entradas y salidas, 6 de los cuales se pueden usar como salidas PWM. Posee además 6 entradas analógicas, un cristal de cuarzo de 16 MHz, un conector USB tipo B, un conector de alimentación DC, y un botón de reinicio. Dispone de todo lo necesario para soportar el microcontrolador y correr programas. Se puede alimentar mediante USB o con el conector DC.

Algunos datos técnicos esenciales de la placa:

Microcontrolador	ATmega328P
Tensión de funcionamiento	5V
Tensión de entrada recomendada	7-12V
Tensión de entrada limite	6-20V
Pines digitales E/S	14
Pines PWM	6
Pines de entrada analógica	6
Máxima corriente de salida pines E/S	20 mA
Corriente para el pin de 3.3V	50 mA
Memoria Flash	32 KB
SRAM	2 KB
EEPROM	1 KB
Velocidad del reloj	16 MHz
Led de estado	13
Largo	68.6 mm
Ancho	53.4 mm
Peso	25 g

La programación de la placa se realiza mediante la IDE de Arduino, más adelante en el apartado de Programación se centra una explicación detallada.



Ilustración 37 Placa de desarrollo Arduino UNO

Se elige esta opción como sistema válido gracias a su gran soporte, su tamaño y coste reducido. El sistema es capaz de gestionar entradas y salidas para sensores y actuadores. Posee memoria interna. Cumple todas las especificaciones de diseño planteadas.

- Convertor Analógico-Digital o ADC: es la unidad del microcontrolador capaz de leer y gestionar señales distintas de los valores de referencia para los que se asocian 0's y 1's. El microcontrolador en sus GPIO o pines de entrada y salida gestiona utilizando una lógica binaria, en el caso de Arduino, un 0 lógico equivale a 0V y un 1 lógico a 5V. Cuando se requiere de la lectura de sensores se presenta un problema y es que estos manejan voltajes variables. Para poder leer dichas tensiones se requiere el uso de un ADC que convierte valores intermedios de voltaje en información que interpreta el microcontrolador.

El proceso de un convertor A/D tiene tres partes:

- Muestreo: se encarga de obtener muestras de señal y asignarles el orden de entrada al sistema. El sistema de muestreo empleado es *Sample and Hold*, cuando llega un valor de señal, el sistema lo mantiene hasta que llega otro, así deconstruye la señal para que el dispositivo sea capaz de interpretarla y manejarla.
- Cuantificador: basándose en el orden y respetándolo asigna valores de tensión según la respuesta de señal obtenida. La precisión está limitada por error de cuantificación.
- Codificador: transforma los diferentes valores de tensión a un código binario que es capaz de identificar el chip.

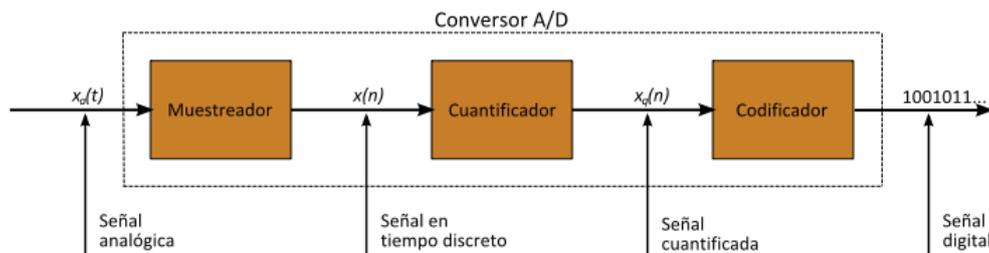


Ilustración 38 Esquema de un convertor A/D

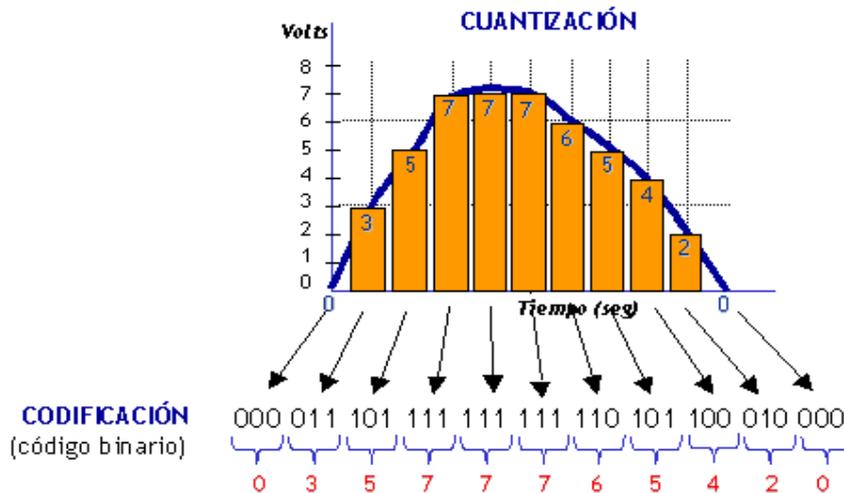


Ilustración 39 Proceso de cuantización

A la hora de realizar un muestreo de una señal analógica se deben tener en cuenta dos factores fundamentales para una captación correcta de la señal:

Tasa o periodo de muestreo: se denomina a la velocidad que el sistema conversor A/D toma muestras de la señal a medir. Las señales analógicas son continuas en el tiempo y para obtener información de éstas se deben extraer muestras representativas que sirvan como referencia para reconstruir la señal digitalmente. El tiempo que tarda el ADC entre la toma de una muestra y la siguiente es el denominado periodo de muestreo. También se puede hacer referencia en términos de frecuencia a frecuencia de muestreo cuando se trata de señales en las que la componente frecuencial es una variable importante a tener en cuenta. La relación se puede obtener de la siguiente forma:

$$T_{muestreo} = \frac{1}{f_{muestreo}}$$

Aliasing: cuando las muestras a medir son variables en el tiempo, es decir, disponen de una componente frecuencial a tener en cuenta, se deben atender con especial cuidado para no errar en su muestreo. Para que se produzca una fiel reproducción digital de la señal analógica que se está midiendo se debe respetar el teorema de muestreo de Shannon-Nyquist. El teorema dicta que, para una señal con frecuencia conocida, la frecuencia de muestreo debe ser al menos dos veces la frecuencia de la señal a muestrear:

$$f_{muestreo} = 2 \times f_{maxima\ señal}$$

Como se observa en la imagen inferior, el primer caso corresponde a un correcto muestreo de la señal, pues los puntos (muestreo) son capaces de reconstruir digitalmente la señal original.

Por el contrario, en el segundo caso, los puntos (muestreo) se han tomado con una frecuencia inadecuada sin respetar el teorema lo que genera una reproducción de la señal incorrecta:

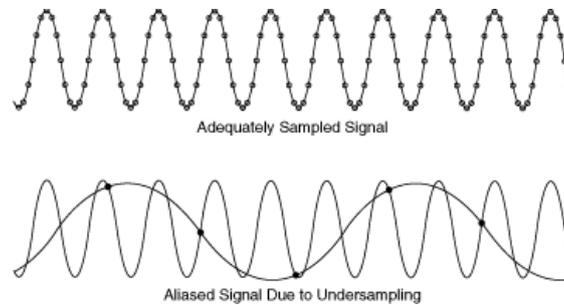


Ilustración 40 Arriba: señal con un correcto muestreo, Abajo: señal con aliasing

Sobremuestreo: cuando la señal se muestrea a una frecuencia muy superior a la frecuencia de Nyquist se produce un sobremuestreo, lo que ocasiona la aparición de ruido blanco en las medidas. El sobremuestreo ocasiona a su vez un consumo de recursos importantes del ADC.

Para muestrear una señal se ha de tener en cuenta el teorema de Nyquist con tal de evitar el aliasing sin superar enormemente esta frecuencia de muestreo para no ocasionar sobremuestreo. Una frecuencia adecuada de muestreo permitirá reconstruir digitalmente la señal original sin ruido ni un consumo excesivo de recursos.

En el caso de Arduino, dispone de un conversor analógico-digital con 6 canales. El conversor tiene una resolución de 10 bits, lo que significa que devuelve valores enteros de entre 0 y 1023. El fondo de escala en el ADC de Arduino es de 5V, por lo tanto:

$$1LSB = \frac{5V}{2^{10}} = 4.88mV$$

Esto quiere decir que la resolución del ADC de Arduino es de 4.88 mV, es la mínima variación de voltaje para que se vea reflejado digitalmente. La conexión de los potenciómetros en la placa quedará de la siguiente forma:

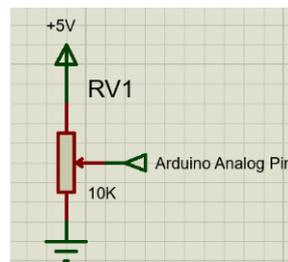


Ilustración 41 Esquema de conexión de los potenciómetros

Todos los elementos que se comunican con Arduino deben conectarse con la placa, bien sea directamente, con una PCB diseñada o una shield mencionada anteriormente:

- **Shield CNC:** se trata de una placa de hardware libre destinada para el control de máquinas de control numérico por computador o CNC. La pequeña placa está destinada para la creación de proyectos caseros con el fin de controlar motores a través de un software en el ordenador. La placa permite controlar hasta 4 motores paso a paso (tres de ellos independientes y el cuarto esclavo de uno de los anteriores) fácilmente con un Arduino gracias a su formato shield permitiendo compatibilidad con Arduino UNO o Arduino Leonardo. La placa soporta 4 drivers o controladores de potencia Pololu A4988 o Pololu DRV8825.

Entre otros dispone de entradas para 6 finales de carrera, salidas relé, entradas de sensores y control de ventiladores. Está diseñada principalmente para ser compatible con GRBL un firmware diseñado para el control de máquinas CNC con instrucciones G-CODE. Entre otras características destacamos las más importantes:

- Soporte para 4 ejes (X, Y, Z, A)
- Conexiones para finales de carrera para cada eje (6 en total)
- Salida "Spindle enable" y "direction"
- Salida "Coolant enable"
- Compatible con Pololu A4988 y DRV8825
- Jumpers para control de micro-stepping
- Diseño compacto
- Los motores pueden ser conectados con bornes tipo Molex de 4 pines
- Alimentación: 12-36V DC

Con el esquema disponible en internet podemos obtener la información del conexionado interno de la placa:

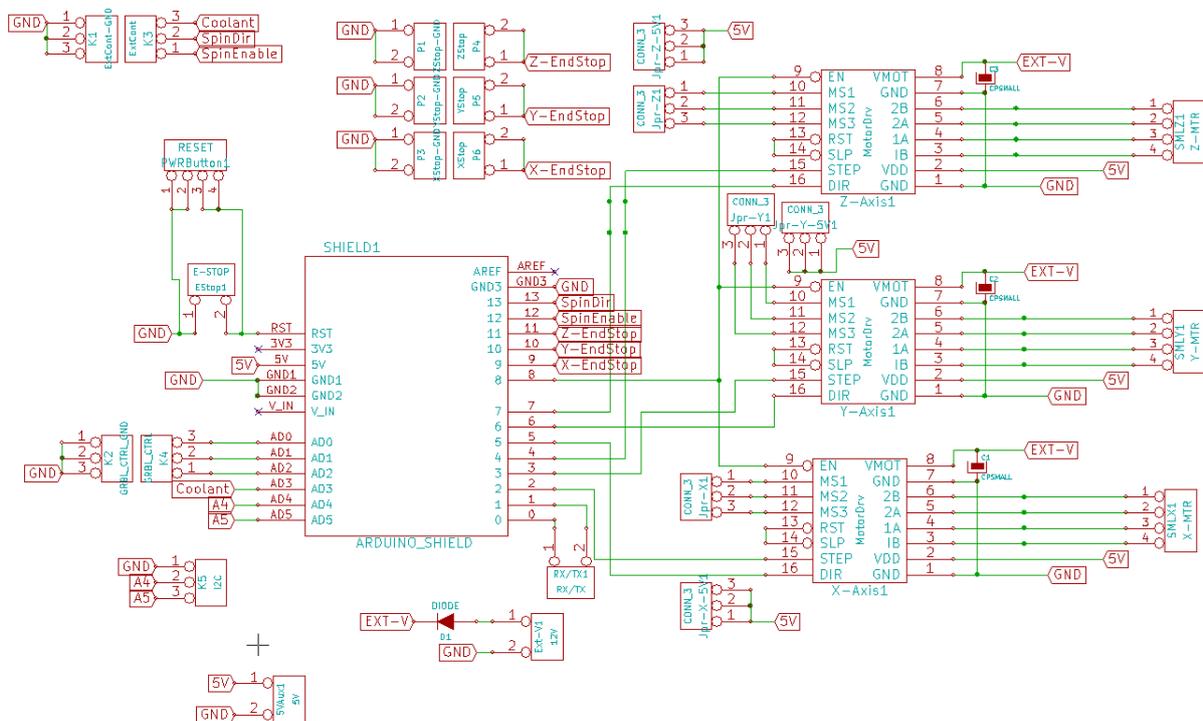


Ilustración 42 Esquema de conexiones de la Shield CNC

Se opta por la utilización de la Shield CNC porque supone una solución de bajo coste para obtener la conexión de todos los elementos electrónicos que se necesitan en el proyecto.

La placa posee un borne para la conexión de la alimentación y un botón de reset. Analizando el esquema de la placa (Ilustración 42) determinamos los pines conectados a las entradas analógicas de la placa Arduino UNO. Para el control de los tres potenciómetros se elegirán las entradas AD3, AD4 y AD5 que corresponden con el Coolant, SDA y SCL. La placa dispone de un pin de 5V y otro de GND. A continuación, se muestra la localización de los pines:

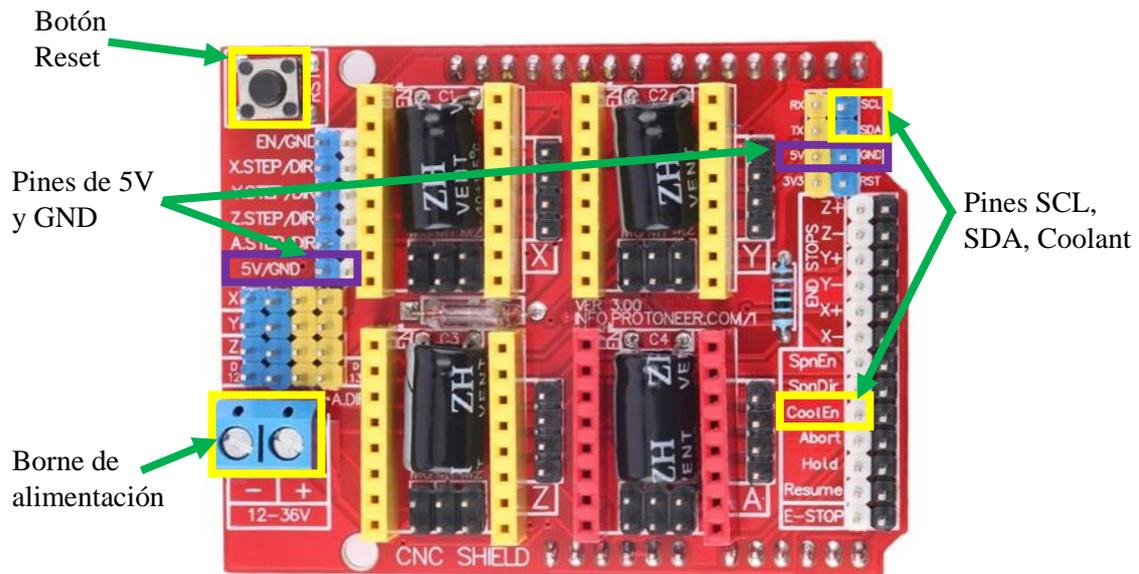


Ilustración 43 CNC shield localizando los pines SCL, SDA y Coolant además de los pines de 5V, GND, el botón de reset y el borne de conexión de la alimentación

Según la configuración de excitación deseada se deben puentear los jumpers correspondientes a MS1, MS2 y MS3. Se elige la configuración de semipisos o  $\frac{1}{2}$  de paso por tanto según la tabla de configuraciones (ilustración 26, pagina 22) se debe puentear MS1 y dejar MS2 y MS3 en estado bajo. El driver de potencia se debe conectar en zócalo. El motor se conecta en los pines localizados al lado de los pines destinados al driver de potencia.

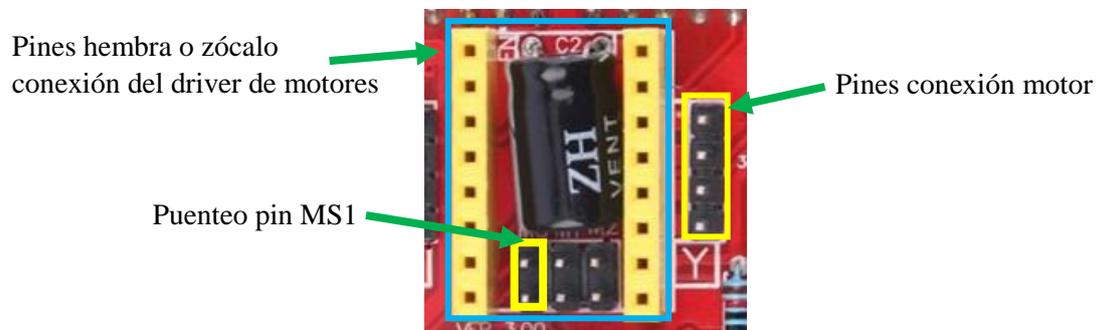


Ilustración 44 Conexión de pines del motor, conexión de driver de potencia y pines MS1, MS2 y MS3

## 4.5 Diseño de la estructura mecánica

### 4.5.1 Elección del modelo

Como hemos visto con anterioridad, y según las especificaciones de diseño se debe elegir un modelo de entre los disponibles comercialmente. Entre las marcas y modelos existentes en el mercado nos centraremos en el IRB 460 de ABB y el KR 40 PA de KUKA los dos son robots paletizadores de cuatro ejes.

En la página web de cada una de las marcas podemos extraer los CAD de los robots que servirán de apoyo para el diseño:

[-http://new.abb.com/products/robotics/es/robots-industriales/irb-460/cad](http://new.abb.com/products/robotics/es/robots-industriales/irb-460/cad)

[https://www.kuka.com/es/es/servicios/descargas?terms=Category:CAD;product\\_name:KR%2040%20PA;Language:en:1;Language:es:1;](https://www.kuka.com/es/es/servicios/descargas?terms=Category:CAD;product_name:KR%2040%20PA;Language:en:1;Language:es:1;)

Los modelos descargados son de tipo DXF un formato de archivo para dibujos de diseño asistido por ordenador.

### 4.5.2 Diseño 2D AutoCAD

Para la exploración de los planos 2D descargados y el diseño del robot del presente proyecto se va a utilizar el programa de diseño AutoCAD. AutoCAD es un software de diseño asistido por ordenador reconocido internacionalmente gracias a sus amplias capacidades de diseño y su facilidad de uso. Existen otras alternativas de diseño CAD, pero la Universidad Politécnica de Valencia ofrece a sus alumnos el software de manera gratuita bajo la licencia de estudiantes.

Se procede a la apertura de los modelos en AutoCAD:

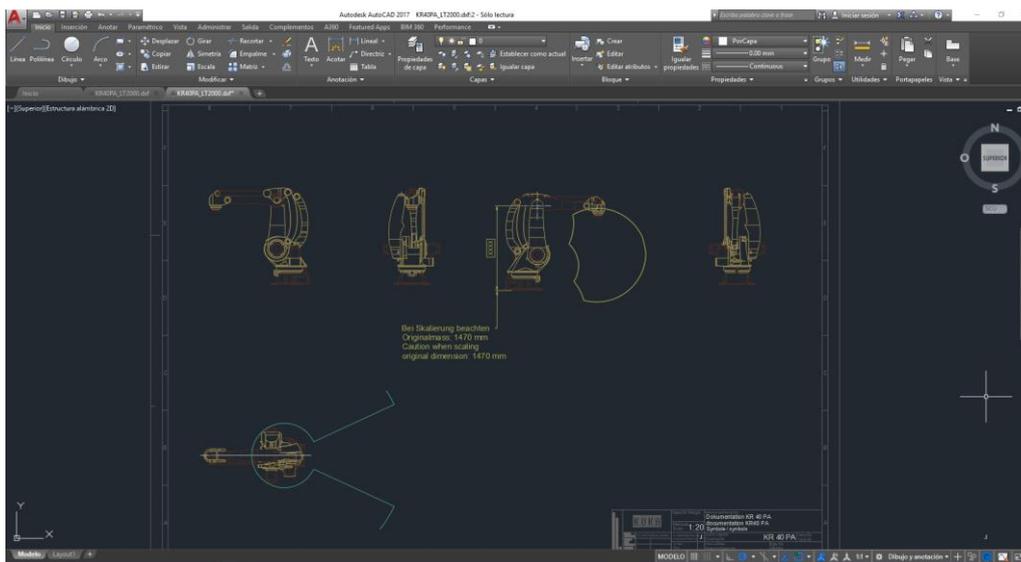


Ilustración 45 Modelo 2D KR 40 PA de KUKA abierto en AutoCAD

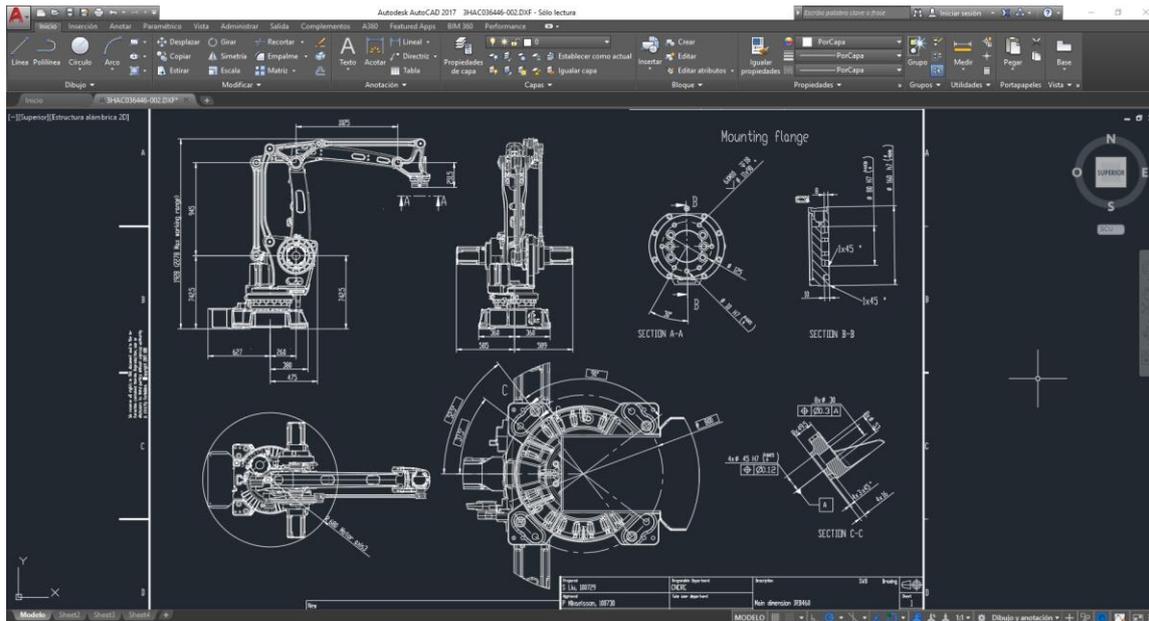


Ilustración 46 Modelo 2D del IRB 460 de ABB abierto en AutoCAD

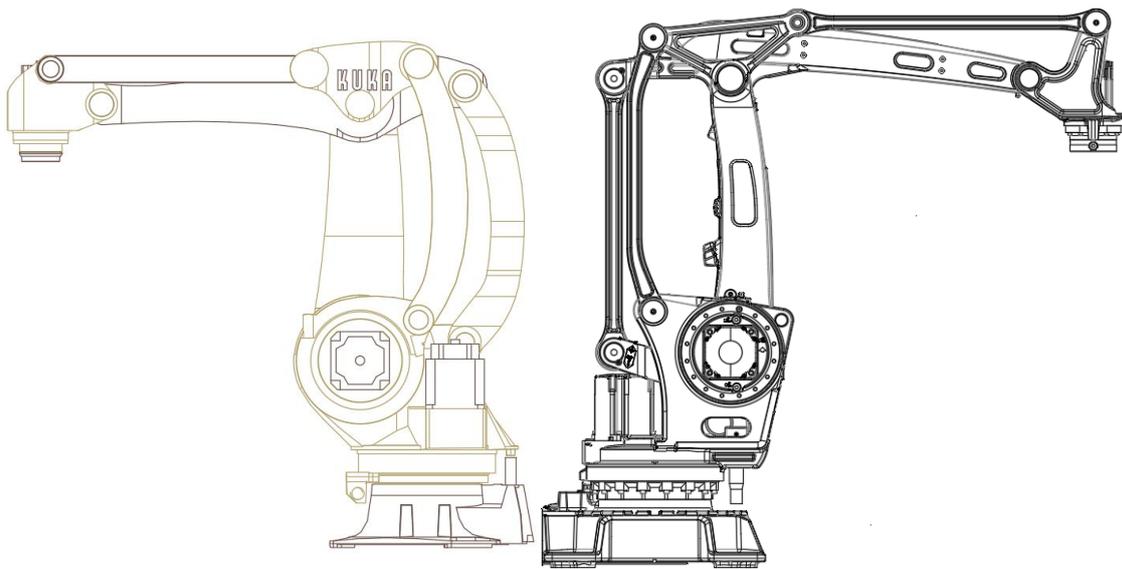


Ilustración 47 Modelos de los robots Kuka y ABB con mayor detalle

Los modelos DXF descargados servirán como referencia para el diseño de las piezas del robot del presente proyecto, se reinventarán las piezas respetando la geometría para que el robot resultante se comporte como los modelos reales. AutoCAD maneja una propia extensión de archivos denominada DWG es similar a la DXF.

Se realiza un dibujo sencillo de la estructura del robot en AutoCAD, con sus diferentes elementos bien definidos y agujeros detallados, se elabora un dibujo esquemático de la base y se diseña un logotipo para el brazo robot. Estos dibujos tendrán una forma sencilla por que la elaboración más compleja y detallada se mostrará en el apartado de diseño 3d. El robot está escalado para no superar el tamaño descrito en las especificaciones técnicas. El robot diseñado quedará de la siguiente forma:

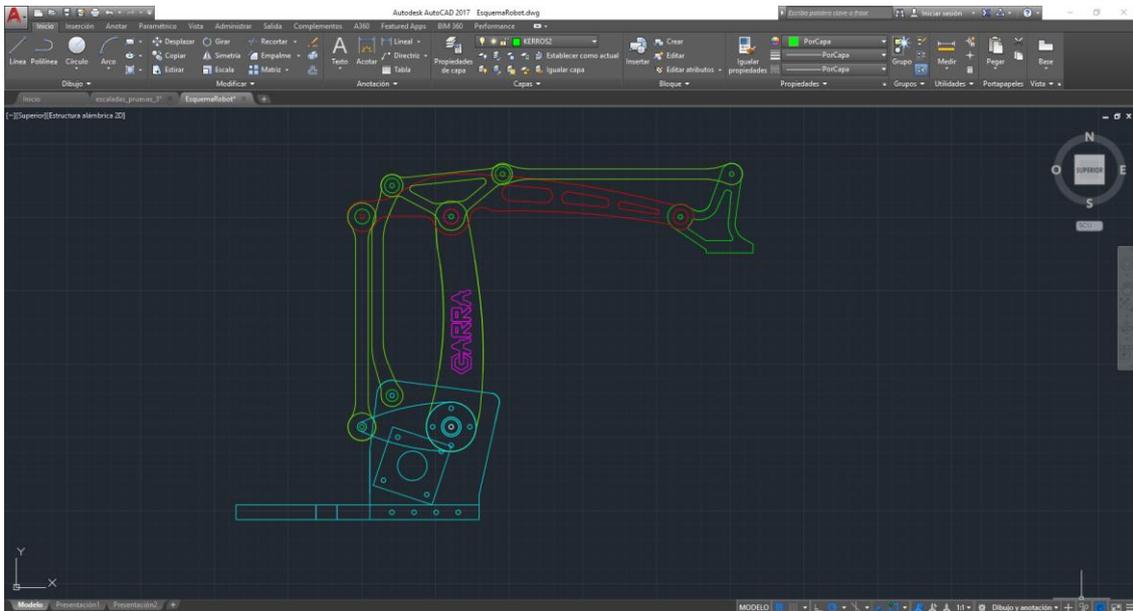


Ilustración 48 Dibujo esquemático de la idea general respetando la geometría en sus elementos

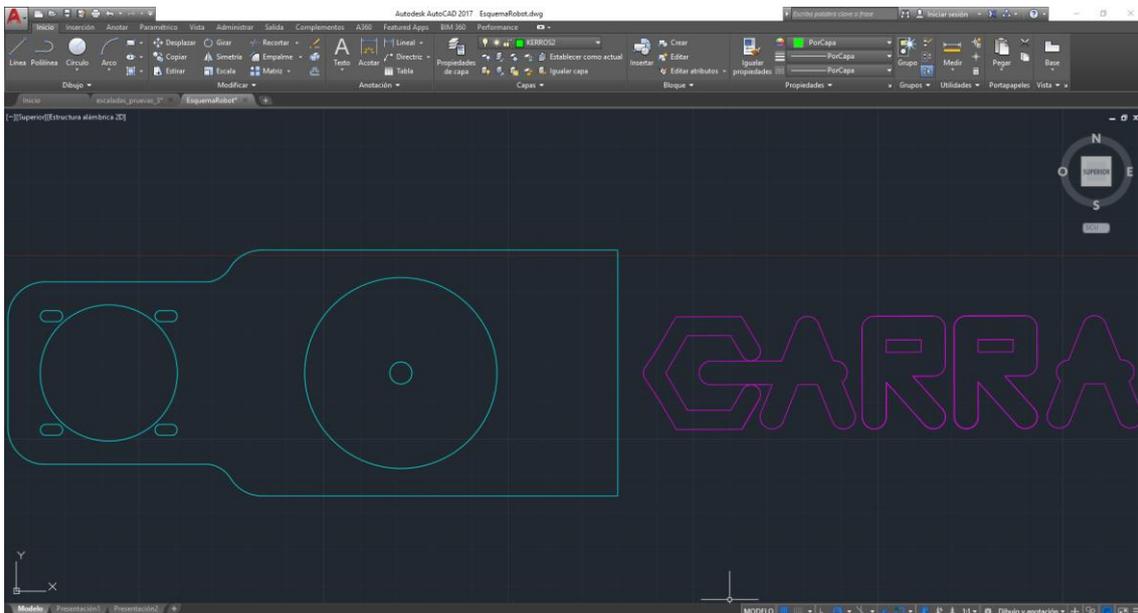


Ilustración 49 Dibujo esquema de la base y logotipo del robot

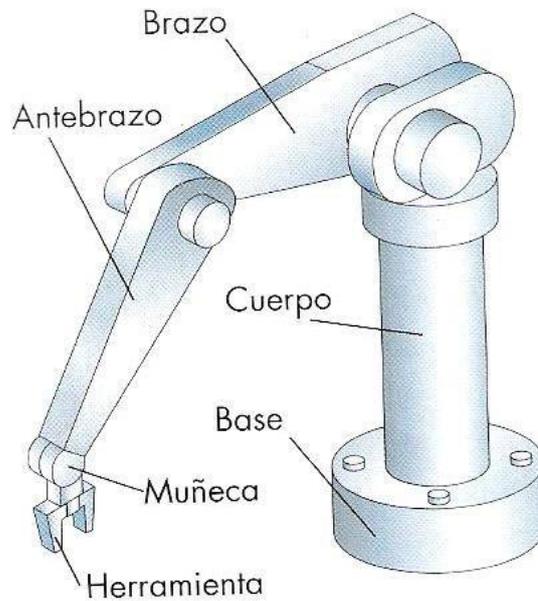
### 4.5.3 Diseño 3D SolidWorks

Una vez diseñadas las piezas en 2D con AutoCAD a partir de los modelos obtenidos de los robots paletizadores de ABB y Kuka, se procede al diseño 3D de las piezas del robot. Para ello se hará uso del diseño realizado en 2D separando cada una de las piezas en un archivo DWG que servirá de base para trabajar en un programa de diseño asistido por ordenador 3D.

Existen muchas opciones de software CAD 3D, pero optaremos por SolidWorks, se trata de un software potente de creación y modelización no solo de piezas sino de conjuntos. El programa se compone de múltiples herramientas de diagnóstico, cálculo, análisis y simulación de piezas. El software permite importar archivos en formato AutoCAD para ser editados. Los alumnos de la UPV gozan de una licencia de estudiantes gratuita para su uso académico.

El brazo robótico se compone de diversas piezas por lo que se dividirá en subpartados para abordar cada una de dichas piezas con un mayor detalle. Esta fase es de las más importantes, se ha de prestar especial atención al modelado 3D de las piezas ya que determinará el éxito del diseño global del sistema.

Como hemos visto con anterioridad dividiremos el diseño 3D de acuerdo con los elementos principales que componen el brazo (ilustración 50).

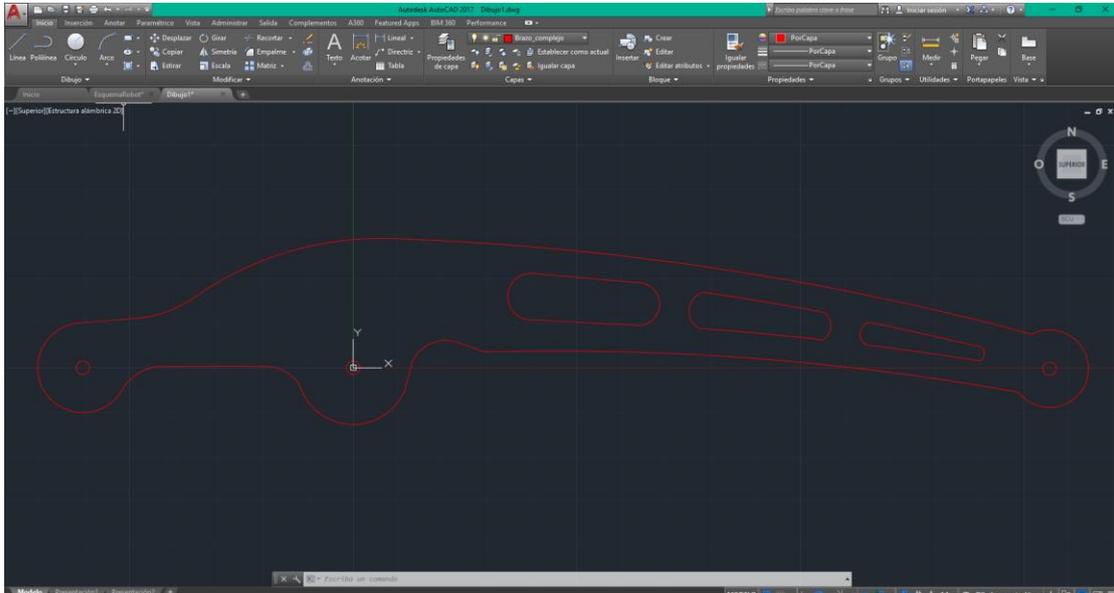


*Ilustración 50 Elementos que componen un robot*

### 4.5.3.1 Antebrazo, Importación y diseño detallado

En este primer apartado se explica cómo importar documentos DWG de AutoCAD en SolidWorks para poder trabajar en el programa de diseño 3D. En los siguientes apartados se obviará la información por que sigue el mismo procedimiento.

Se presenta la pieza diseñada y aislada del conjunto:



Se procede a guardar el documento en Archivo → Guardar Como → Dibujo (DWG)

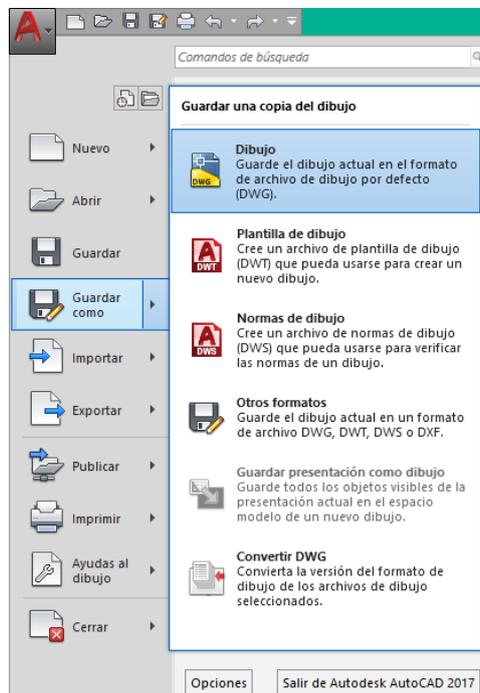


Ilustración 51 Guardar documentos AutoCAD en formato DWG

Se inicia el SolidWorks con la espera necesaria para que cargue todos sus complementos hasta que aparezca el asistente de ayuda. En el asistente de ayuda, click en abrir y se elige el archivo en este caso “antebrazo” en formato DWG. Una vez se abre el asistente de importación (ilustración 52) se hace click en “importar nueva pieza como:” y se elige la opción “Croquis 2D” y por último click en “Finalizar”, el programa abre el croquis del modelo AutoCAD diseñado (ilustración 53), se puede modificar el croquis o se puede comenzar a su modelado 3D.

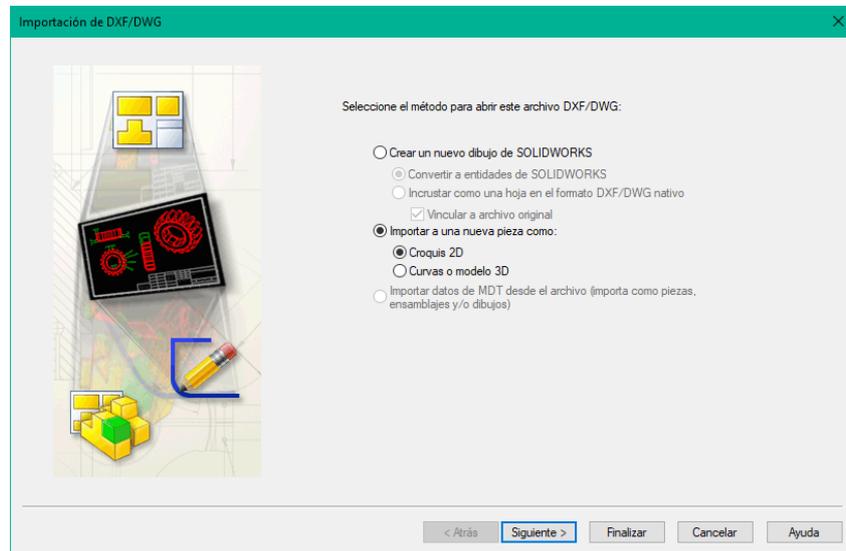


Ilustración 52 Importación del modelo DWG 2D de AutoCAD en SolidWorks

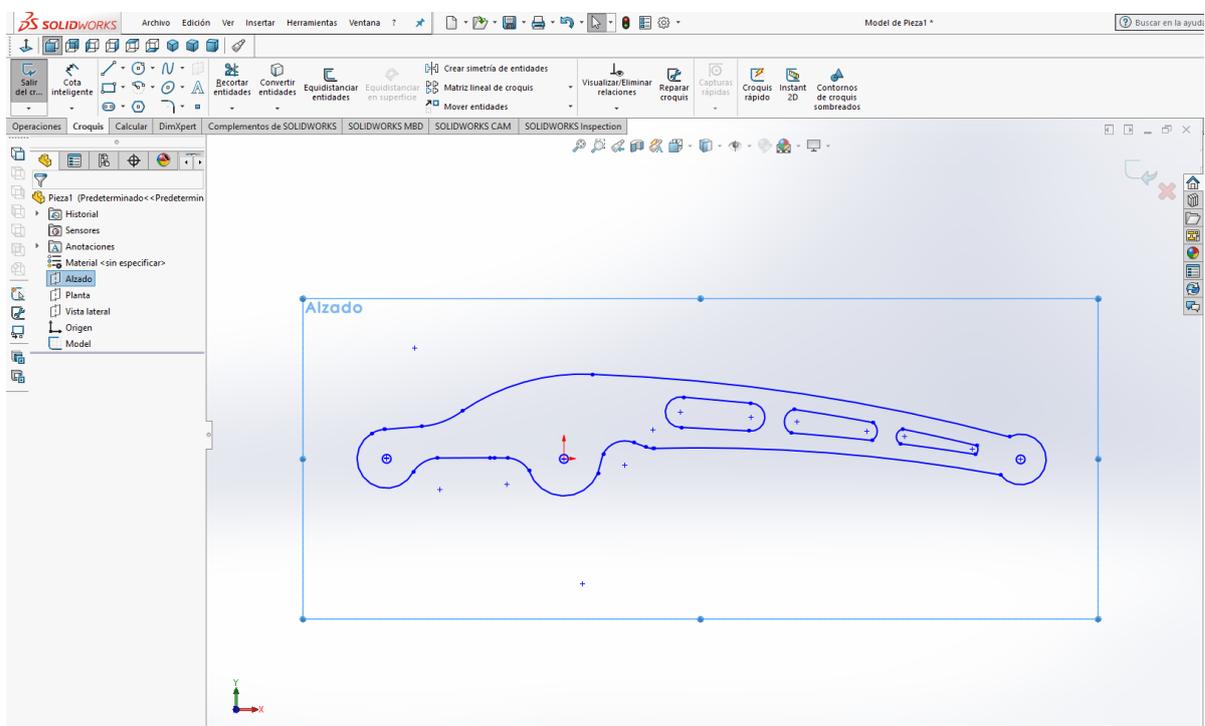


Ilustración 53 Croquis del modelo DWG importado

Para comenzar a modelar en 3D, se elige la opción de la barra superior “extruir base/saliente” esta herramienta extruye, en el plano en el que se está trabajando, una longitud de pieza que determina el usuario. Se elige el plano (en verde), la cara de la pieza (en rojo) y se determina la distancia a extruir:

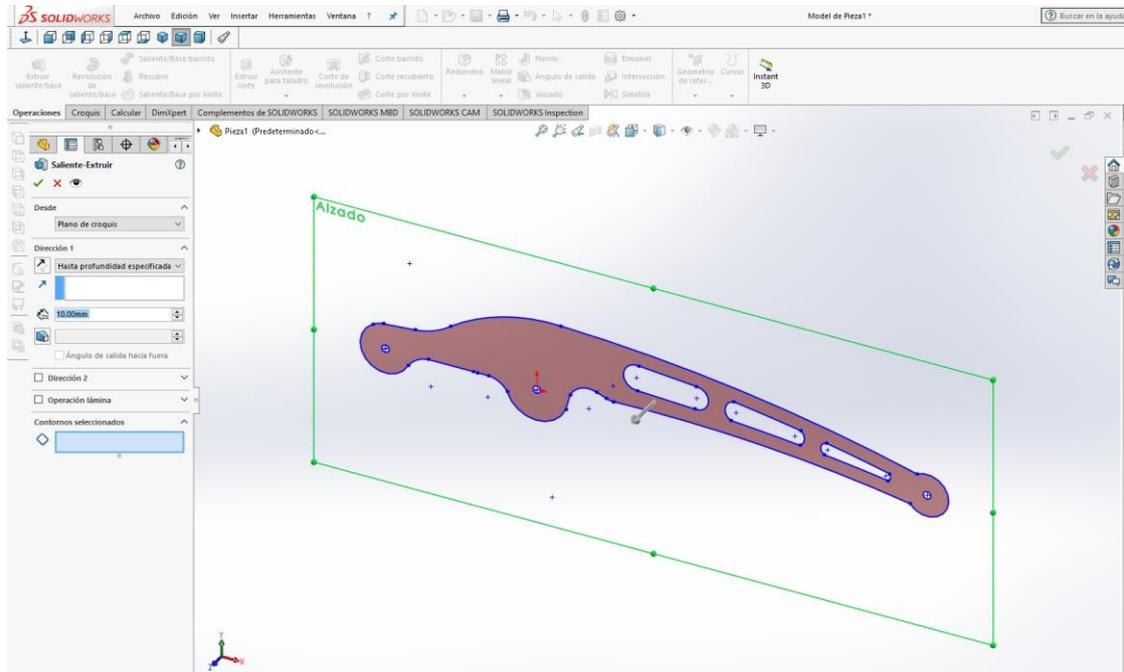


Ilustración 54 Selección del plano (verde) y cara de la pieza (rojo) para la extrusión

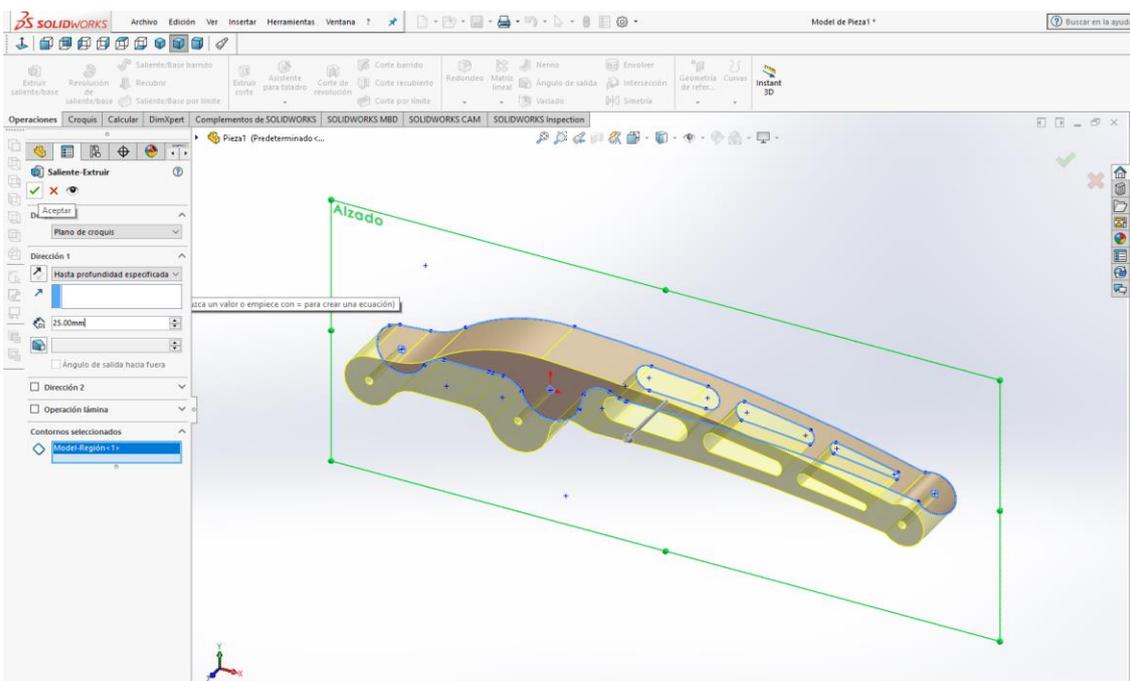


Ilustración 55 Simulación de la pieza tras introducir la distancia de extrusión

Una vez determinada la extrusión, se comienzan a realizar los detalles, en este caso se van a generar los orificios para introducir los rodamientos, primeramente, se selecciona una cara para trabajar sobre el modelo, se abre el editor de croquis y se dibujan dos circunferencias con el tamaño y la localización donde se alojan los rodamientos. Posteriormente se elige la opción de “extruir corte” y se especifica la profundidad con tal de retirar “material” del espacio que se quiere vaciar para introducir el rodamiento.

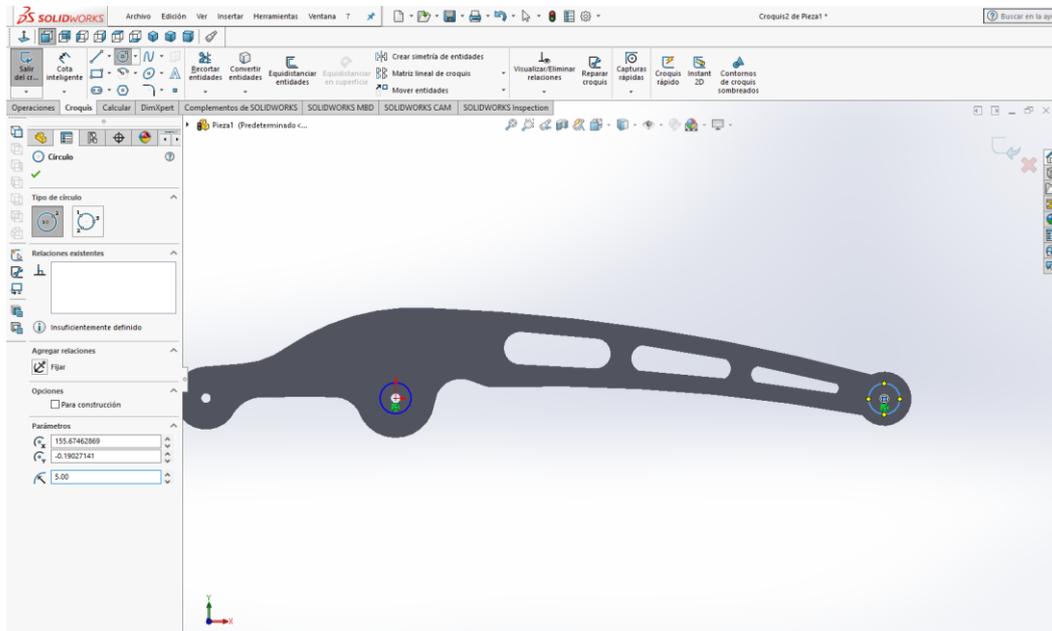


Ilustración 56 Diseño de croquis para los agujeros

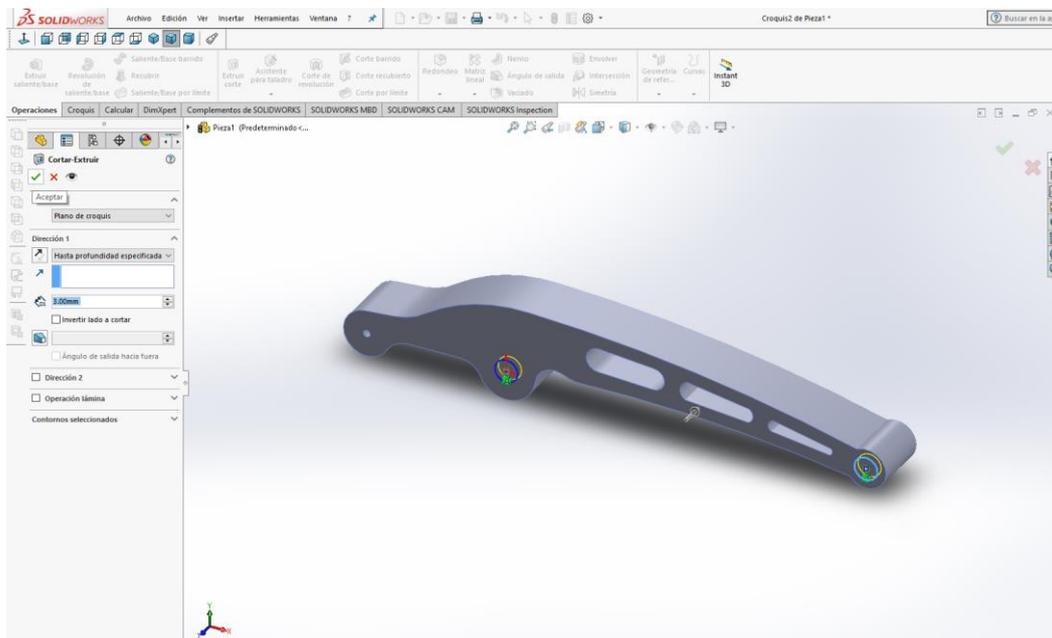
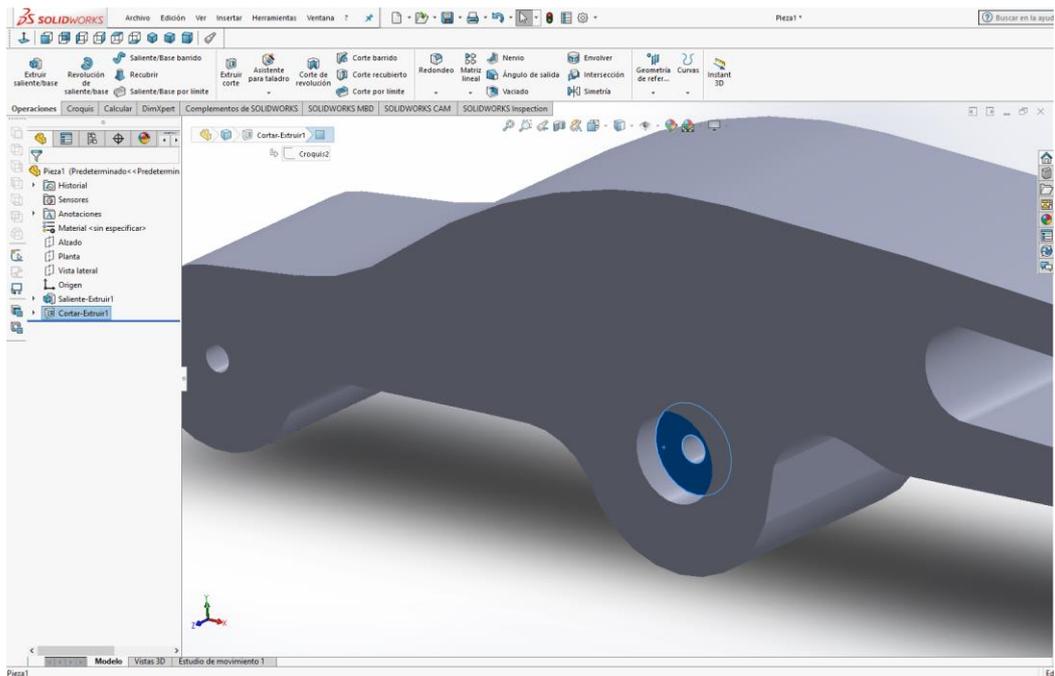
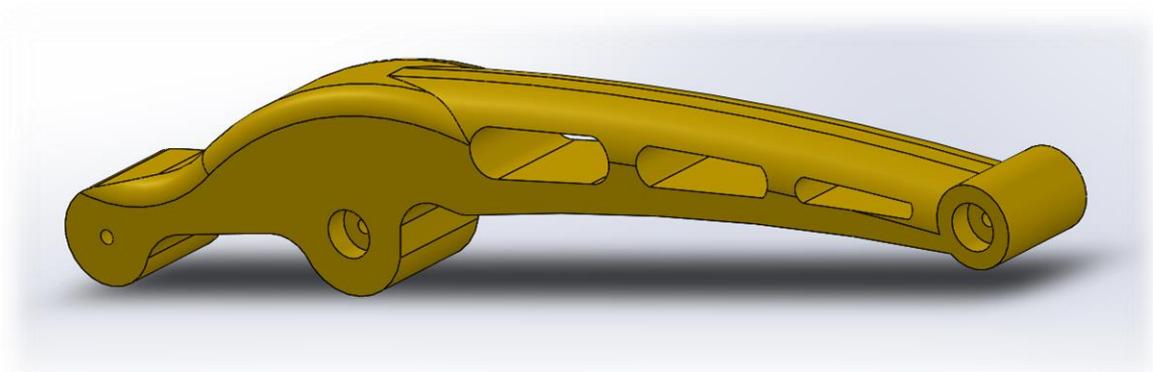


Ilustración 57 Herramienta de corte para vaciar el agujero



*Ilustración 58 Detalle del resultado final del corte*

El antebrazo del robot es una pieza que debe resistir esfuerzos por lo que presenta una estructura sólida y resistente. Posee unos agujeros pensados para reducir el peso de la pieza manteniendo su resistencia. Posee cuatro cavidades pensadas para albergar los rodamientos. Entre otras herramientas, SolidWorks permite suavizar bordes y redondear esquinas, el resultado final se muestra a continuación:



*Ilustración 59 Diseño final del antebrazo*

Como se ha dicho anteriormente, los futuros apartados se limitan a explicar el diseño y funcionamiento final de las piezas sin especificar los pasos a seguir para el diseño de estas.

#### 4.5.3.2 Brazo, diseño detallado

A continuación, se muestra el diseño del brazo del robot, se trata de una pieza que al igual que el antebrazo, debe soportar carga y por tanto resistir esfuerzos, es por esto que presenta una estructura sólida y resistente acompañada de tres cavidades principales que permiten reducir el peso respetando la resistencia de la pieza. Esta pieza es el nexo de unión entre la base del robot y el antebrazo por lo tanto presenta en los extremos los zócalos para alojar las piezas y unir los ejes. Esta pieza es movida mediante el 2º eje por tanto presenta 4 agujeros dispuestos en el lateral izquierdo para fijar la corona del engranaje. Se puede observar el detalle del logotipo “Garra” situado en ambos laterales de la pieza.

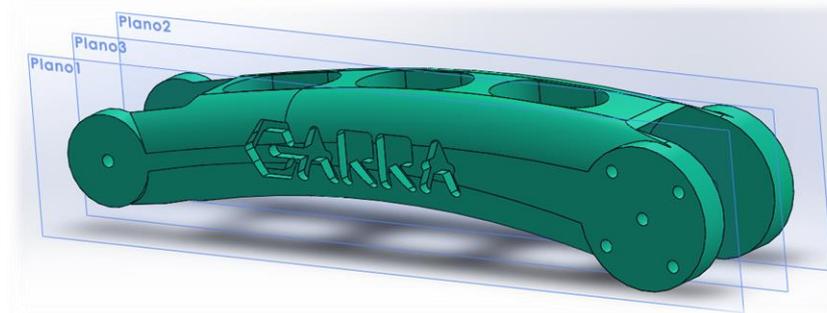


Ilustración 60 Diseño en del brazo del robot con detalle del logotipo

#### 4.5.3.3 Muñeca, diseño detallado

La muñeca supone el extremo del robot destinado a interactuar con objetos del entorno. El robot paletizador, en la muñeca, posee el cuarto eje del robot encargado de dar la orientación del efector final siendo siempre paralelo al suelo. La muñeca a su vez es la última articulación y por tanto en ella se encuentra alojada la herramienta del robot. La muñeca se divide en 4 piezas principales.

- Las piezas laterales dan la forma a la muñeca, son idénticas salvo que la pieza derecha posee un separador de 5 mm para ajustar la distancia de las barras. Son un total de dos piezas:

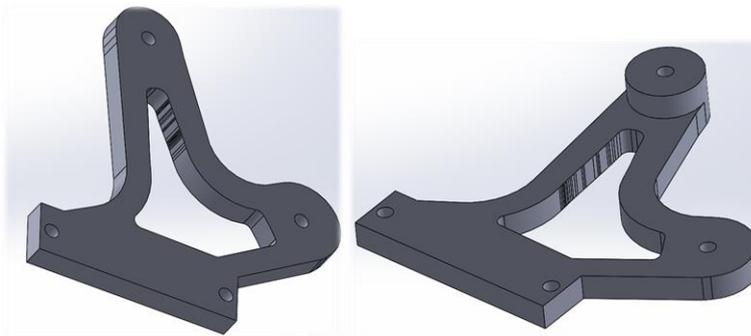
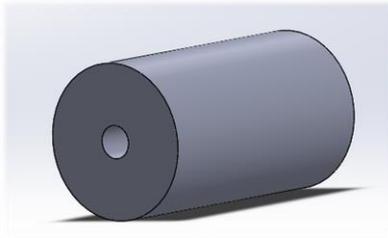


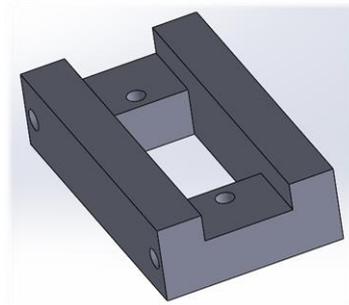
Ilustración 61 Piezas laterales izquierda y derecha, la derecha con separador

- Separador cilíndrico cuya misión es sujetar los dos laterales:



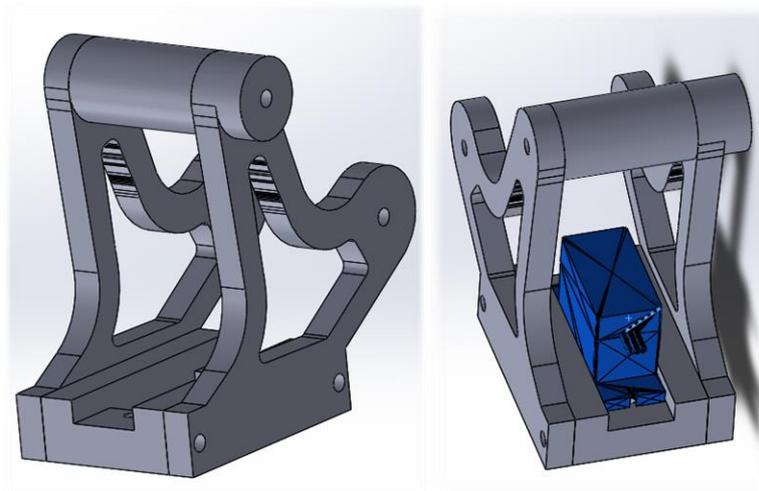
*Ilustración 62 Separador cilíndrico*

- Zócalo servomotor. Se diseña una estructura que permita alojar el servomotor TowerPro 9g y fijarlo a la muñeca, además se diseña de tal forma que pueda unirse al conjunto de laterales y separador mediante tornillos:



*Ilustración 63 Zócalo para servomotor*

El programa de diseño SolidWorks posee una herramienta capaz de crear ensamblajes a partir de piezas por separado creando relaciones de posición y movimiento. El diseño del conjunto queda de la siguiente forma:



*Ilustración 64 Ensamblaje de la muñeca, la segunda imagen muestra alojado el servomotor 9g*

#### 4.5.3.4 Engranajes, diseño detallado

SolidWorks posee una herramienta denominada Toolbox, en ella se puede encontrar sistemas y elementos mecánicos clasificados en países según su métrica, norma o estándar. Esta herramienta solo funciona en el modo ensamblaje ya que introduce piezas nuevas en el modelo.

Se va a optar por el uso de dicha herramienta para el diseño de los engranajes ya que de forma sencilla la herramienta es capaz de calcular y ofrecer soluciones estandarizadas en base los parámetros introducidos. Para acceder a ella solo se debe hacer click en el botón “activar Toolbox” situado en lateral derecho del explorador. Realizada esta acción se abrirá un menú desplegable donde aparecen normas y estándares de diferentes países. Como toda la mecánica y electrónica elegida sigue el estándar europeo se selecciona la opción “ISO” que desplegará diferentes opciones, se debe elegir “Transmisión de potencia” y luego “Engranajes”. Por último, en la pestaña inferior se elegirá el tipo de engranaje a utilizar, en nuestro caso “Engranaje recto”:

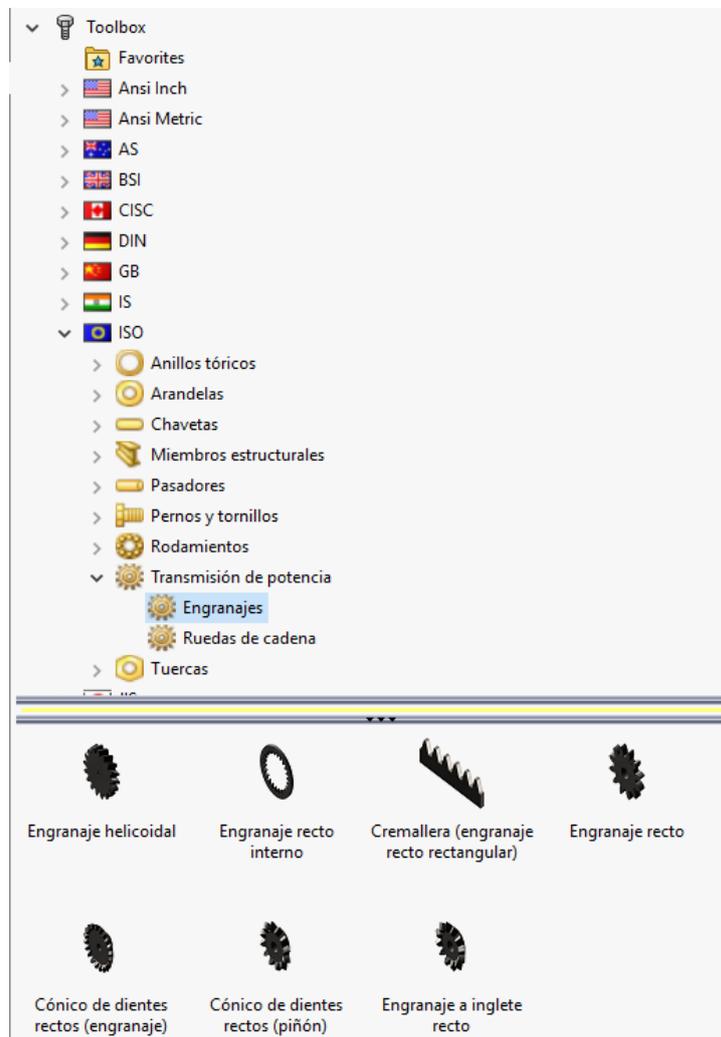
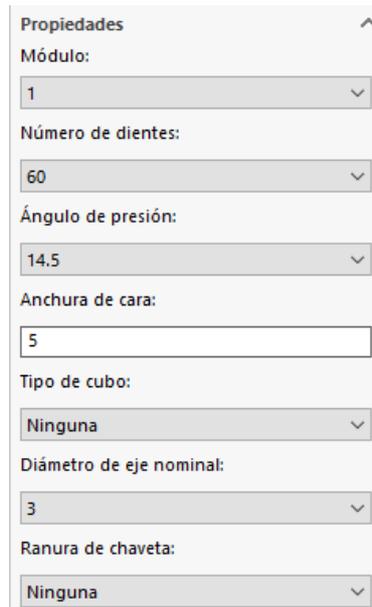


Ilustración 65 Herramienta Toolbox con la elección de engranajes disponibles

Tras elegir la opción se despliega el menú que ayuda a realizar el diseño de forma sencilla, se muestra a continuación:



Propiedades	
Módulo:	1
Número de dientes:	60
Ángulo de presión:	14.5
Anchura de cara:	5
Tipo de cubo:	Ninguna
Diámetro de eje nominal:	3
Ranura de chaveta:	Ninguna

Los parámetros a modificar son:

- Módulo: es la relación que existe entre el diámetro primitivo o de paso del engranaje y su número de dientes, coincide con la relación existente en el paso del engranaje y  $\pi$ . A la hora de diseñar una cadena de engranajes, el módulo debe ser el mismo para que puedan trabajar. Esta opción permite elegir un número en una lista de selección.
- Número de dientes: determina el número de dientes que va a poseer el engranaje, es una variable importante ya que determinará la relación de movimiento frente a otros engranajes. La opción se elige en una lista de selección.
- Ángulo de presión: se trata del ángulo entre la dirección de la fuerza de contacto de los dientes y la dirección de la velocidad en el engranaje conducido. La opción se elige en una lista de selección.
- Anchura de cara: determina el espesor del engranaje. Se puede elegir libremente el espesor.
- Tipo de cubo: permite la creación de material adicional en la pieza a modo de separador o guía. La opción se elige en una lista de selección y permite elegir en uno, los dos lados o ninguno.
- Diámetro de eje nominal: se trata del diámetro del eje del propio engranaje. La opción se escoge en una lista de selección.
- Ranura de la chaveta: La chaveta es una pieza rectangular o cuadrada que permite transmitir potencia y evitar que se produzcan deslizamientos entre dos elementos que deben ser solidarios entre sí. La opción permite añadir un orificio al engranaje para alojar la chaveta.

Existen otros parámetros importantes a la hora del diseño de un sistema de engranajes, pero Toolbox maneja estas variables internamente para obtener un diseño eficiente y de forma sencilla. Muchas opciones del Toolbox manejan listas de selección porque se basan en el estándar seleccionado, en este caso el ISO.

Pese a la facilidad de uso del Toolbox, se precisa de la utilización de ecuaciones fundamentales del diseño de engranajes para el cálculo de éstos. Sabemos que la relación de transmisión de movimiento es 4:1 es decir que la corona tendrá cuatro veces más dientes que el piñón. Se decide elegir 15 dientes para el piñón lo que resulta:

$$\text{Relación} = 4:1, \quad N^{\circ}\text{Dientes}_{\text{piñón}} = 15,$$

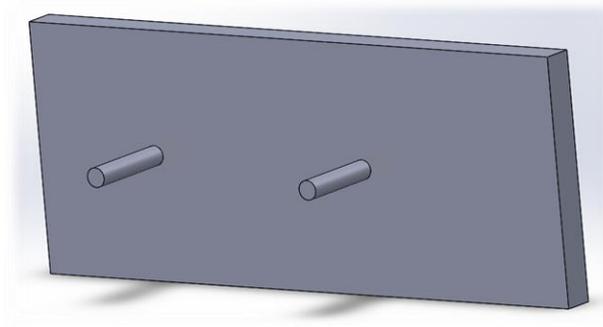
$$N^{\circ}\text{Dientes}_{\text{corona}} = N^{\circ}\text{Dientes}_{\text{piñón}} \times 4 \rightarrow N^{\circ}\text{Dientes}_{\text{corona}} = 15 \times 4 = 60$$

Obtenido el número dientes de cada engranaje, se precisa la obtención de la distancia entre centros o ejes de los engranajes. Se conoce el número de dientes de los engranajes y se determina el módulo=1 en ambos, obteniendo así:

$MDL = \text{Módulo}$ ,  $N = \text{Numero de dientes}$ ,  $PD = \text{Diametro de paso}$ ,  $D = \text{Distancia ejes}$

$$MDL = \frac{PD}{N}, \quad D = \frac{PD_{\text{corona}} + PD_{\text{piñón}}}{2} \rightarrow D = \frac{60 + 15}{2} = 37.5 \text{ mm}$$

Se crea una plataforma sencilla que nos servirá para el diseño del piñón y la corona, se determina que el diámetro del eje del engranaje es de 3 mm siguiendo el estándar de M3 elegido en los tornillos.



*Ilustración 66 Diseño de pedestal para diseñar los engranajes*

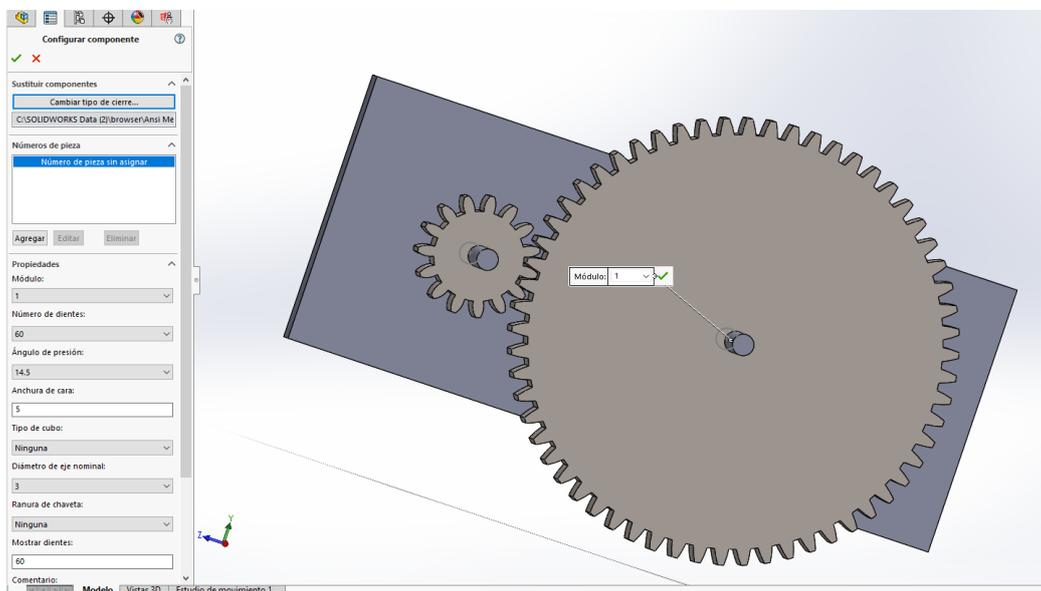
La velocidad máxima a la que puede girar el eje del motor nema 17 de acuerdo con la hoja de especificaciones es de 1800 RPM, con la nueva configuración de transmisión, el piñón asociado al eje del motor (engranaje motriz) girará a velocidad máxima de 1800RPM mientras que la corona, que se trata del engranaje arrastrado, su máxima velocidad se puede calcular como:

$$\text{Relación} = 4:1, \quad V_{\text{max piñón}} = 1800 \text{ RPM}, \quad V_{\text{max corona}} = \frac{1800}{4} = 450 \text{ RPM}$$

A continuación, se expondrá una tabla donde aparecen las configuraciones de los engranajes introducidas en el Toolbox para su diseño:

<i>Módulo</i>	1	1
<i>Número de dientes</i>	60	15
<i>Ángulo de presión</i>	14.5	14.5
<i>Anchura de cara</i>	5	5
<i>Tipo de cubo</i>	Ninguna	Ninguna
<i>Diámetro de eje nominal</i>	3	3
<i>Ranura de chaveta</i>	Ninguna	Ninguna

Tras su introducción y creación en Toolbox se ensambla en la plataforma diseñada y se crean las relaciones de posición y movimiento para comprobar que funciona correctamente de acuerdo con las especificaciones de diseño.



*Ilustración 67 Diseño de los engranajes en Toolbox y comprobación*

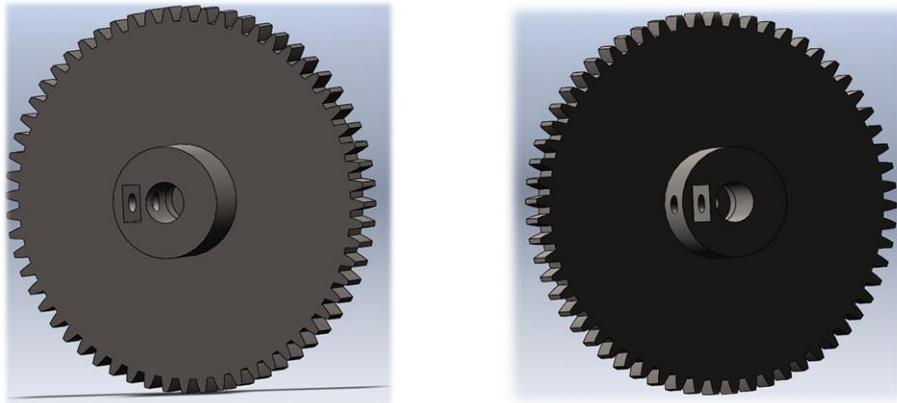
Una vez verificado su correcto funcionamiento, se exporta el piñón y la corona. Al ser elementos creados con el Toolbox, estos no pueden ser editados bajo ningún parámetro que no sea interno del Toolbox. Para poder editar su diseño se deben convertir en archivos editables de tipo pieza con el fin de que SolidWorks interprete que se tratan piezas y no interactúe el Toolbox.

Dentro del directorio C:\Archivos de programa\SolidWorks\SolidWorks\Toolbox\data utilities" se encuentra la aplicación "sldsetdocprop.exe" que permite cambiar las propiedades de las piezas seleccionadas. Se debe cambiar de "tipo: componente toolbox" a "tipo: pieza". Editada esta propiedad, SolidWorks interpretará las piezas como archivo tipo pieza y se podrán editar libremente sin incompatibilidades ni intromisión del Toolbox.

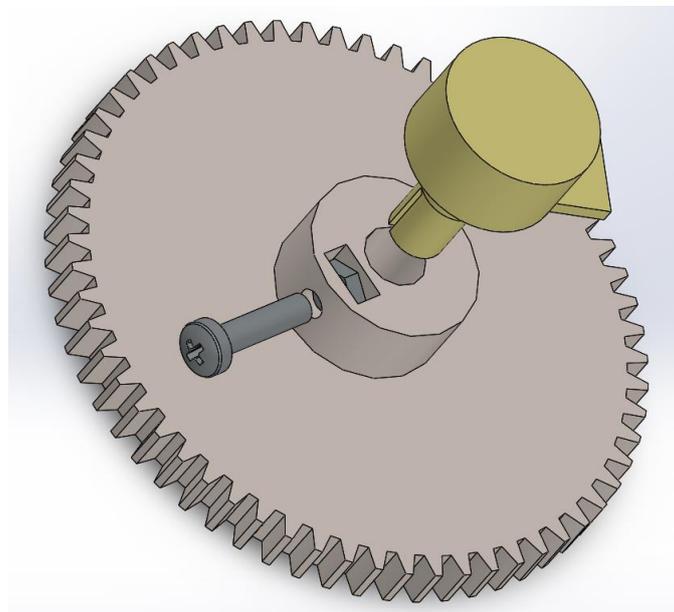
Los engranajes son las piezas móviles asociadas a los tres ejes principales, representan el movimiento angular real de cada uno de los ejes por tanto son los elementos idóneos para obtener un sentido de la posición mediante los potenciómetros. En su diseño se va a añadir un sistema que permita alojar el sensor de posición (potenciómetro) y fijarlo al engranaje.

Una vez editadas las propiedades, se procede a mejorar el diseño del engranaje añadiendo un cilindro saliente con una ranura para alojar una tuerca de M3 y un agujero para un tornillo pasante de 3 mm. Al igual que una chaveta, este sistema permitirá fijar el movimiento de rotación generado en la corona y permitirá transmitirlo al eje del potenciómetro.

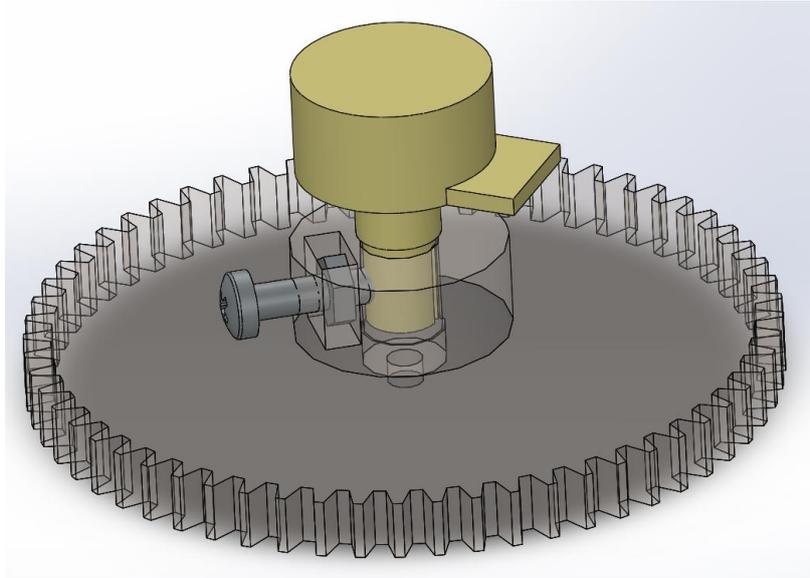
El diseño queda de la siguiente forma:



*Ilustración 68 Corona con diferente orientación, detalle del sistema de fijación con el potenciómetro*

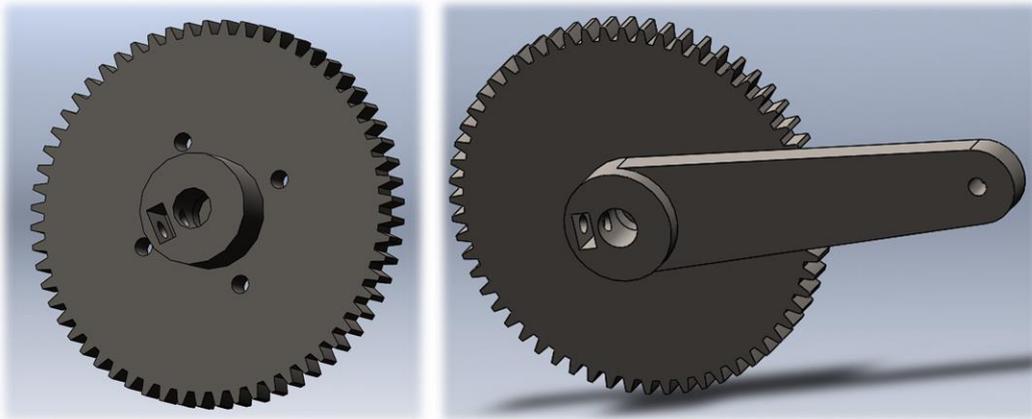


*Ilustración 69 Detalle del sistema de fijación del potenciómetro*



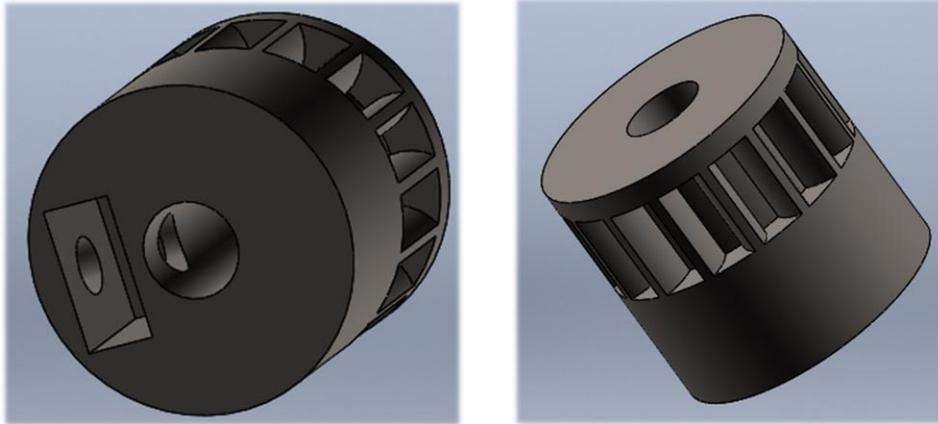
*Ilustración 70 Detalle de fijación del sensor de posición*

Una vez diseñado el zócalo para la fijación del sensor, se modifica la pieza para obtener dos engranajes correspondientes a los ejes 2 y 3 del robot. Al primer engranaje se le añaden cuatro orificios que encajan con los descritos en el brazo, estos cuatro orificios permitirán alojar cuatro tornillos para la fijación del engranaje al brazo del robot. El segundo engranaje se encarga de mover el antebrazo a través de un sistema de barras por tanto se le añade una barra respetando la geometría descrita.

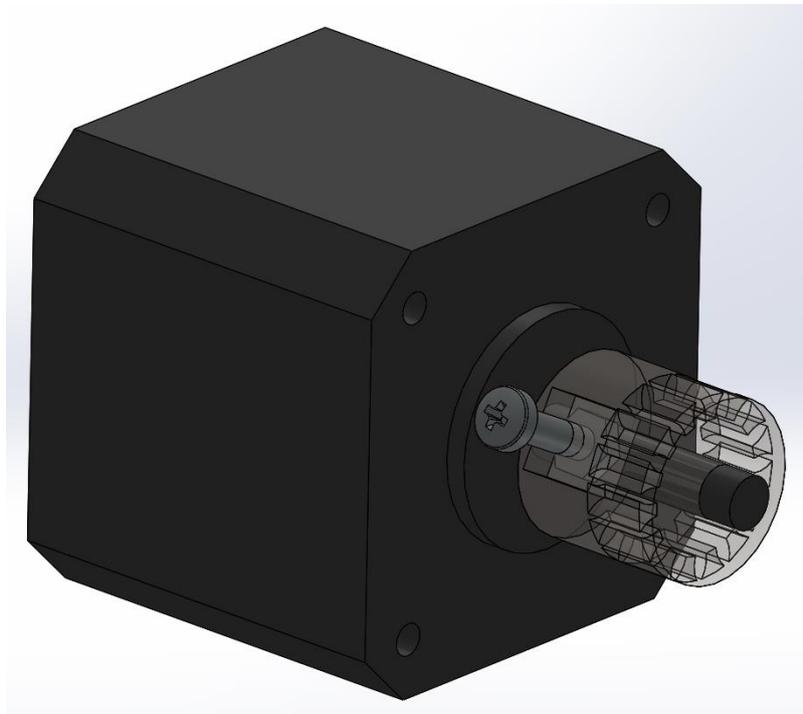


*Ilustración 71 Engranaje del eje 2 con agujeros, y engranaje del eje 3 con barra*

Siguiendo el mismo procedimiento que en la corona, se añade un sistema de fijación en el piñón que permita mantener solidario el piñón al eje del motor. En este caso ambos dos piñones de los ejes 2 y 3 son idénticos



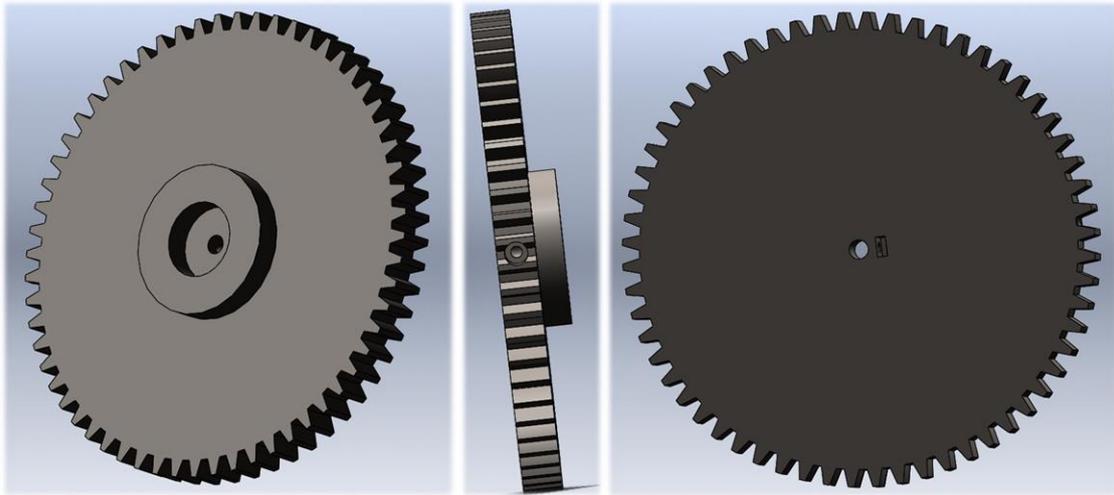
*Ilustración 72 Piñón con diferentes orientaciones, detalle de fijación*



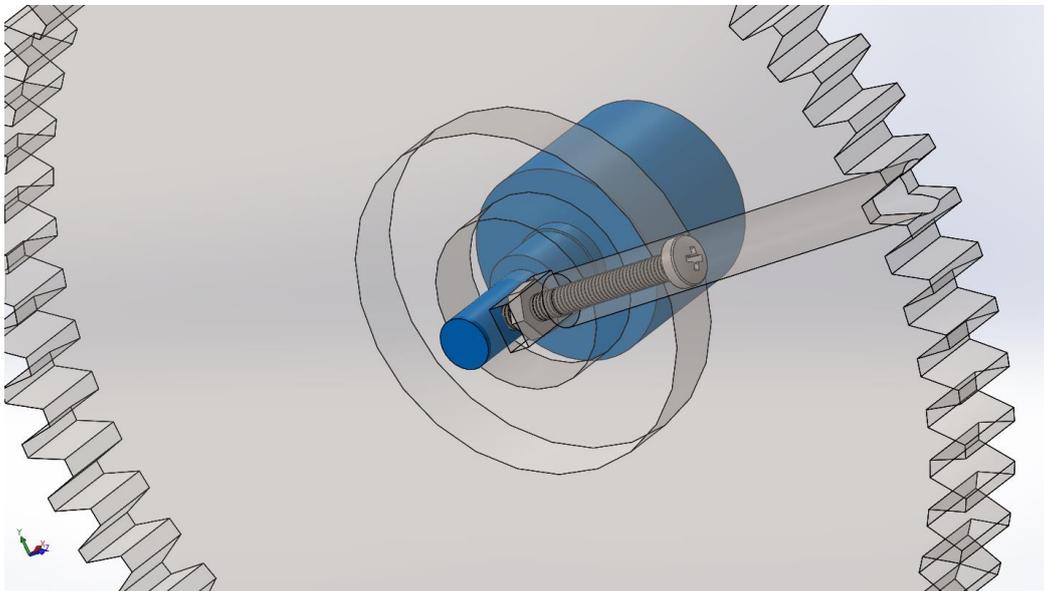
*Ilustración 73 Detalle del sistema de fijación piñón-eje motor*

En el diseño de engranajes de la base se ha evitado utilizar el sistema de diseño mediante el Toolbox. Para una mayor facilidad, se ha elegido la rueda y piñón anteriormente diseñados y se ha escalado en proporción. SolidWorks posee una herramienta llamada “escalar” que permite como su propio nombre indica, escalar la pieza en base a una proporción propuesta.

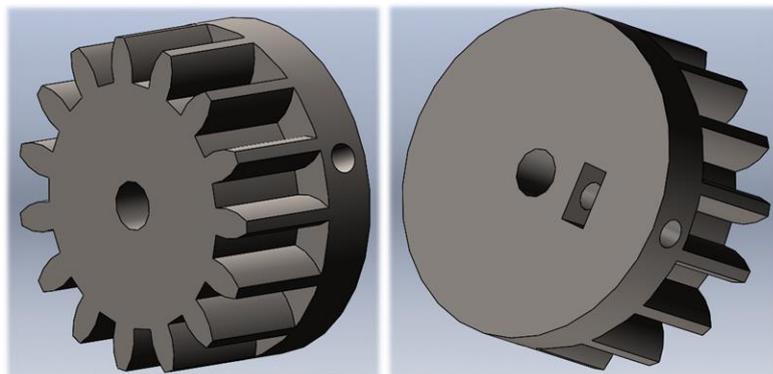
Para su diseño se ha tenido en cuenta la disposición de un cilindro en la corona para poder acoplar el rodamiento P19. Tanto la corona como el piñón poseen un sistema de acople para fijarse al eje (piñón) y al potenciómetro multivuelta (corona).



*Ilustración 74 Corona de la base con sistema de fijación al potenciómetro*



*Ilustración 75 Detalle de sistema de fijación del potenciómetro multivuelta a la corona*



*Ilustración 76 Piñón de engranaje de la base con sistema de fijación al eje del motor*

#### 4.5.3.5 Barras, diseño detallado

Como se ha descrito con anterioridad, el robot posee un sistema de barras colocadas geoméricamente que permiten que el efector final se encuentre paralelo al suelo, se han tenido en cuenta los agujeros para tornillos M3 y los orificios para los rodamientos 623Z. No se ahondará en su diseño específico, a continuación, se muestran imágenes del diseño de las piezas en el SolidWorks:

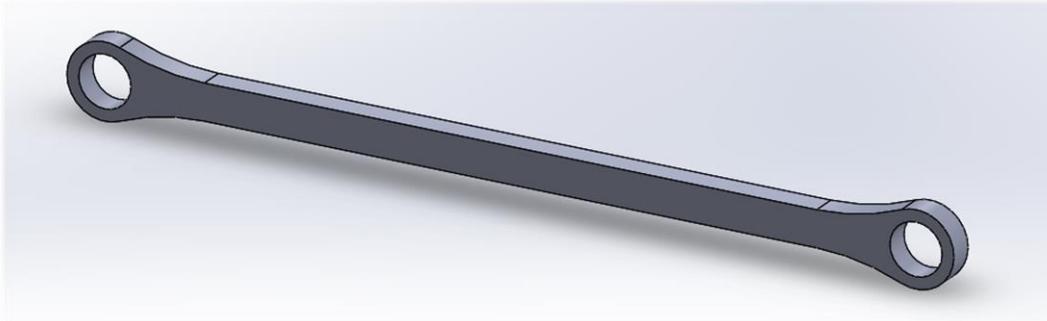


Ilustración 77 Barra de unión del enlace triangular con la muñeca

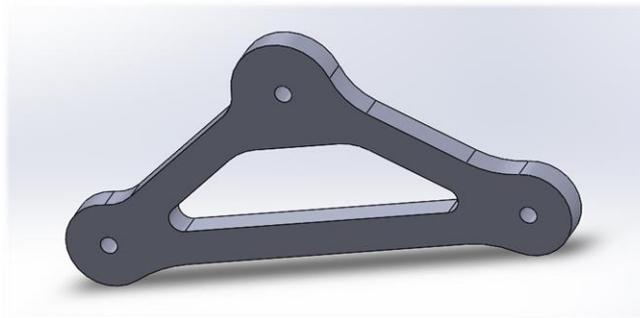


Ilustración 78 Enlace triangular alojado en la unión entre el antebrazo y brazo

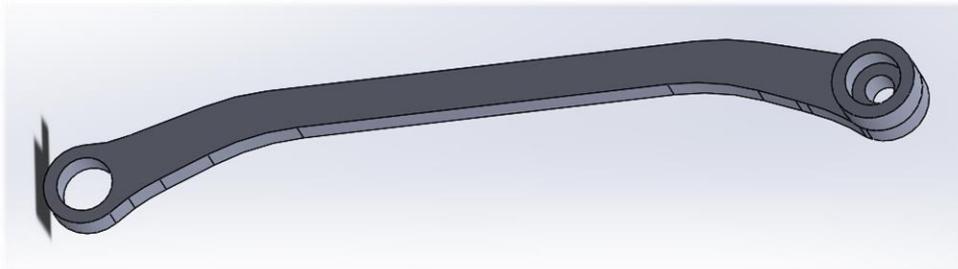


Ilustración 79 Barra de unión entre el enlace triangular y la base



Ilustración 80 Barra de unión entre el extremo del antebrazo y el engranaje con barra

#### 4.5.3.6 Base y laterales, diseño detallado

La base del robot es una pieza fundamental que representa la estructura del robot, ha sido diseñada junto con los laterales para que el centro de masas se encuentre lo más cercano al suelo. La base del robot esta diferenciada por dos zonas. En las ilustraciones 80 y 81 se puede observar que en la parte izquierda existe un orificio para albergar el engranaje solidario al motor del eje 1, se puede ver con más detalle en la ilustración 81 como existen unos orificios que permiten ajustar tolerancias ente el contacto de los engranajes. En medio se sitúa un orificio en el que se fijará el potenciómetro multivuelta.

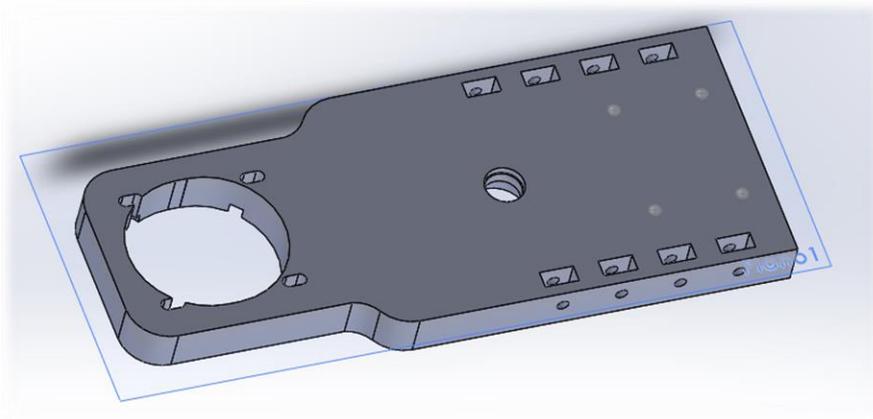


Ilustración 81 Base del robot, detalle de diseño de la parte superior

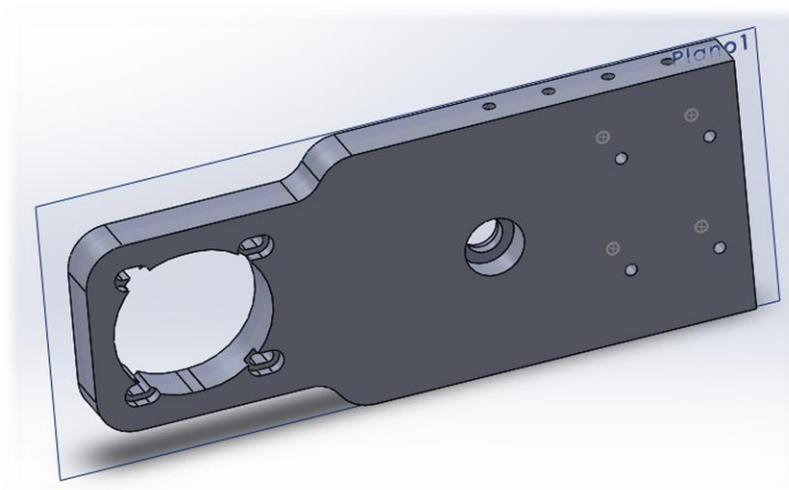
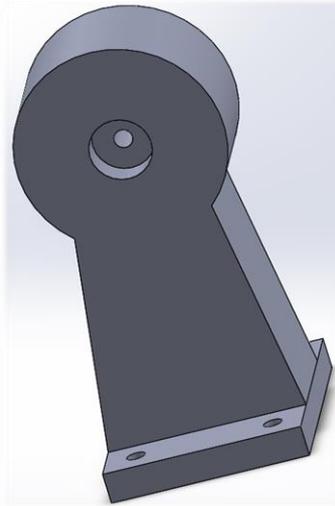


Ilustración 82 Base del robot, detalle de diseño de la parte inferior

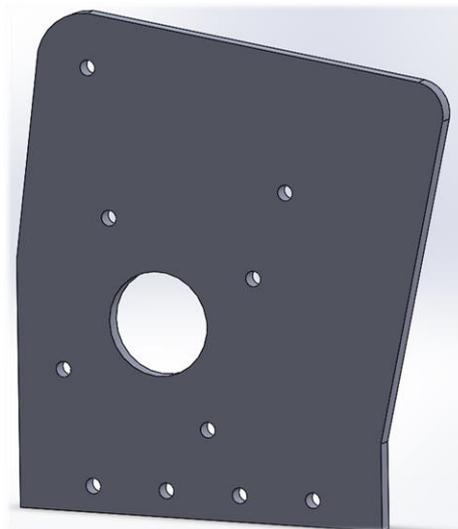
Los ocho orificios rectangulares están diseñados para albergar tuercas de M3, junto con un tornillo pasante por el lateral permitirá fijar los laterales. Los orificios circulares permiten fijar el soporte del brazo.

El soporte del brazo posee dos cavidades para alojar dos rodamientos 623Z y una estructura circular en la parte superior que imita el comportamiento rotativo que describe el brazo, además de los cuatro agujeros para fijar la pieza mediante tornillos en la base del robot.



*Ilustración 83 Soporte del brazo*

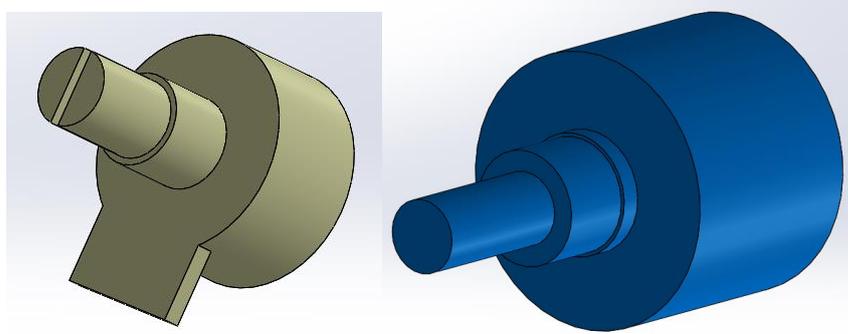
Las dos piezas laterales del robot son idénticas. Poseen orificios para fijarse a la base del robot, fijar el motor a la pieza, fijar el potenciómetro y en el caso de la pieza derecha, fijar la barra descrita en el anterior apartado.



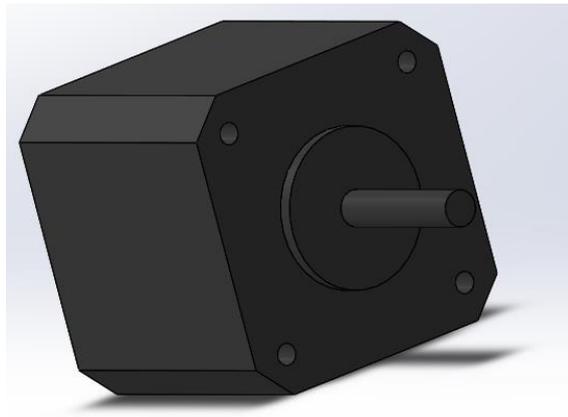
*Ilustración 84 Lateral del robot*

#### 4.5.3.7 Piezas auxiliares, diseño detallado

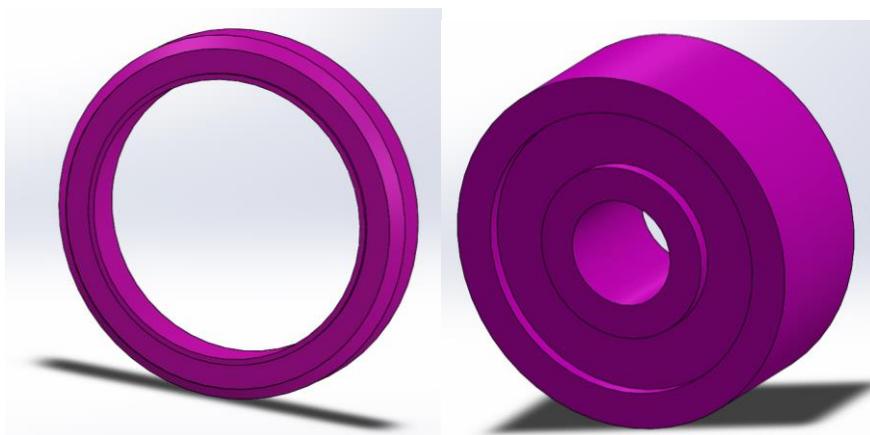
Para la ayuda en el diseño de las piezas del robot, se ha requerido de la modelado de ciertos componentes del robot:



*Ilustración 85 Izquierda potenciómetro normal, derecha potenciómetro multivuelta*



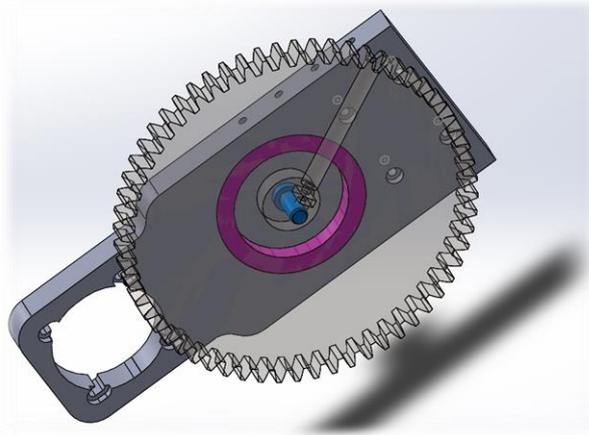
*Ilustración 86 Diseño motor nema 17*



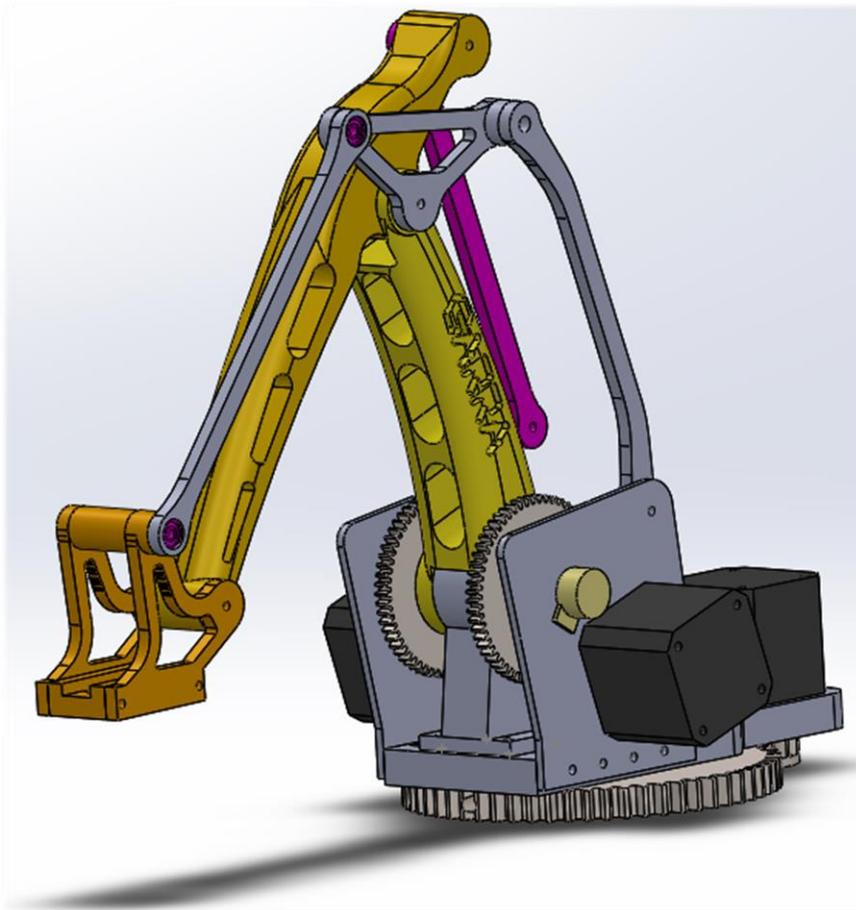
*Ilustración 87 Izquierda rodamiento P19, derecha rodamiento 623Z*

#### 4.5.3.8 Ensamblaje del modelo completo, diseño detallado

Con la herramienta ensamblaje que dispone SolidWorks y con todas las piezas diseñadas junto con el uso de las piezas auxiliares se construye el ensamblaje completo del robot:



*Ilustración 88 Ensamble de la base del robot junto con el sistema de engranajes*



*Ilustración 89 Ensamble completo del robot*

#### 4.5.3.9 Cinemática del Robot diseñado

Un robot paletizador, como se ha definido con anterioridad, tiene la propiedad de mantener el efector final siempre paralelo al suelo. Las estructuras conformadas por tres paralelogramos articulados consiguen mantener el paralelismo entre las diferentes articulaciones del robot logrando la que el efector cumpla su función. En la siguiente figura se puede observar el sistema donde el primer paralelogramo (descrito en color verde) consigue mover el eje 2. El segundo paralelogramo transmite el movimiento del motor desde la base del robot para efectuar el movimiento del eje 3 (descrito en color azul). El tercer paralelogramo (descrito en color rojo) junto con el primer paralelogramo se encarga de mantener la muñeca o efector final con la orientación paralela al suelo. El balancín o enlace triangular respeta las mismas propiedades que el efector final, su orientación no varía.

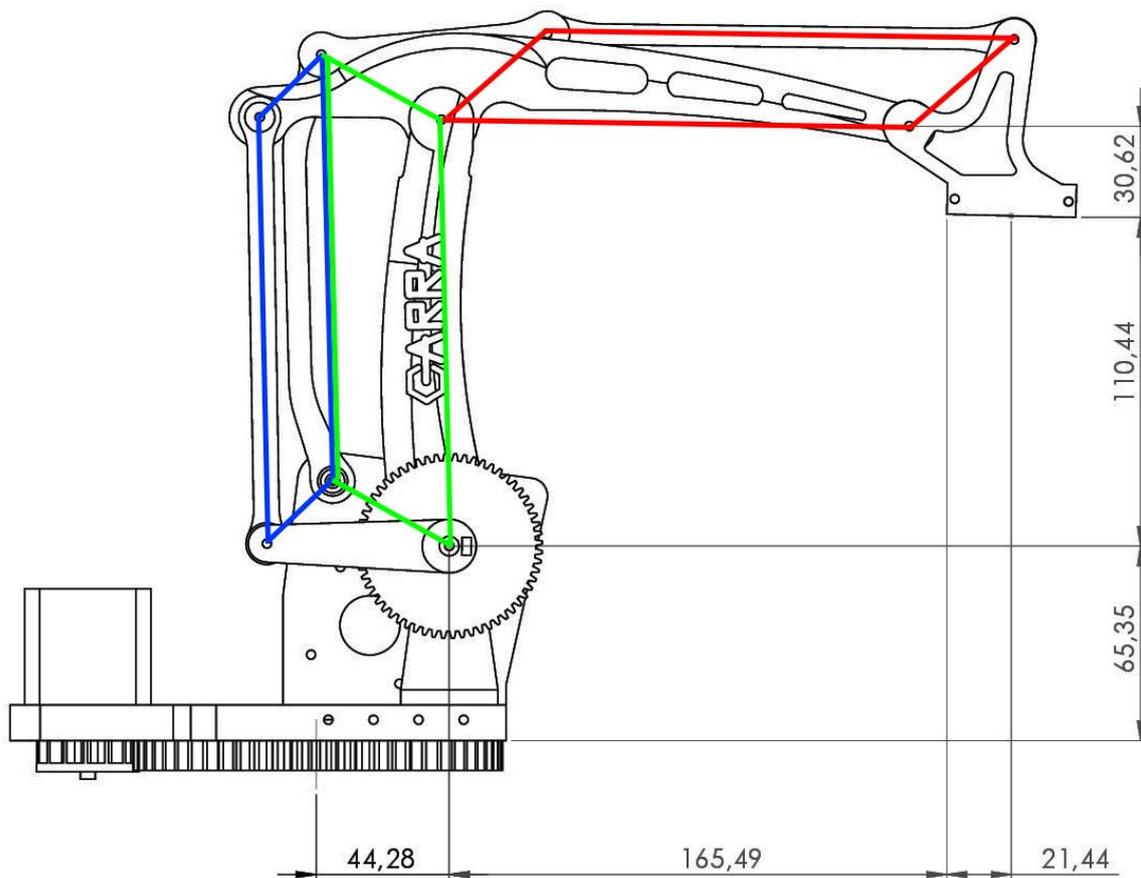


Ilustración 90 Esquema del robot don la distribución de los paralelogramos.  
Paralelogramo 1 verde. Paralelogramo 2 azul. Paralelogramo 3 rojo.

Una vez descrito el funcionamiento de los paralelogramos del robot, se procede a insertar cada uno de los ejes de las articulaciones para obtener los parámetros Denavit-Hartenberg. Como el sistema de paralelogramos transmite el movimiento, los sistemas de referencia se sitúan directamente en cada uno de los ejes del robot.

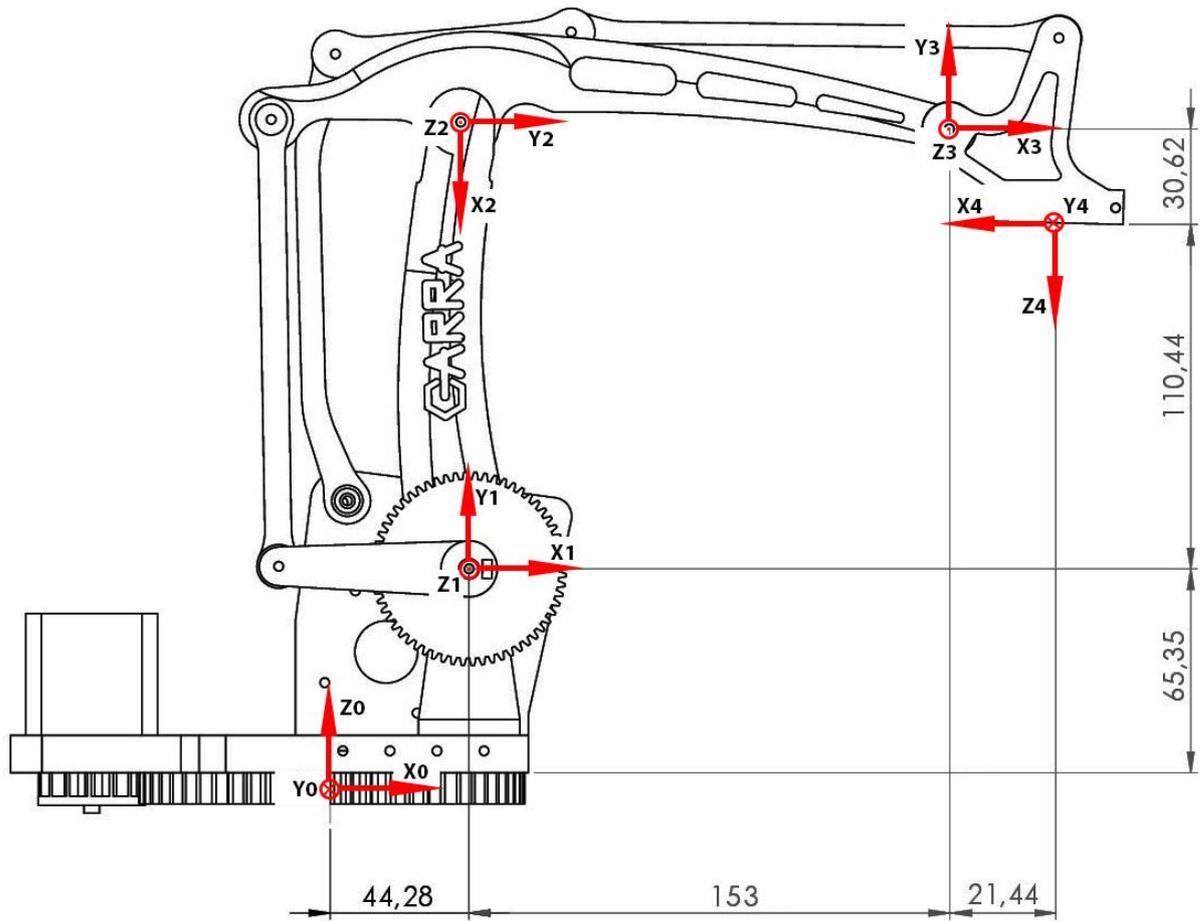


Ilustración 91 Ejes de Denavit-Hartenberg y cotas del robot

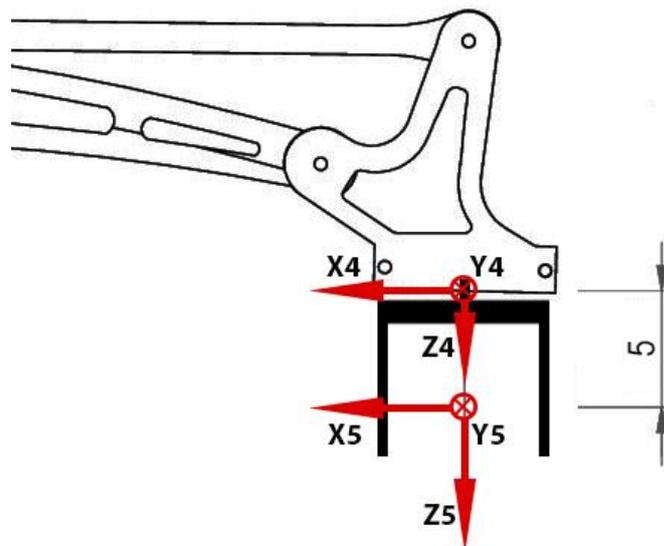


Ilustración 92 Detalle de la muñeca con los ejes de Denavit-Hartenberg y cota

- Parámetros Denavit-Hartenberg.

Situados los ejes se procede a obtener los cuatro parámetros de Denavit-Hartenberg ( $d, a, \theta, \alpha$ ) del ensamble completo del robot:

Articulación	$\theta_i$ (°)	$d_i$ (mm)	$a_i$ (mm)	$\alpha_i$ (°)
1	$q_1$	65.35 mm	44.28 mm	$90^\circ$
2	$q_2 - 90^\circ$	0 mm	-141.06 mm	$0^\circ$
3	$q_3 + 90^\circ$	0 mm	153 mm	$0^\circ$
4	$180^\circ$	30.62 mm	21.44 mm	$-90^\circ$
5	$q_4$	5 mm	0	$0^\circ$

- Resolución del problema cinemático directo de posición.

Para la resolución del problema cinemático directo de posición se requiere la obtención de la matriz de transformación del sistema completo. Por ello se requiere primeramente obtener por separado cada una de las matrices de transformación de los eslabones  $n$  y  $n+1$ .

Matriz de transformación entre el eslabón 0 y el eslabón 1:

$$A_{01}(\theta_1, d_1, a_1, \alpha_1) \rightarrow \begin{pmatrix} \cos(q_1) & 0 & \sin(q_1) & 44.28 \times \cos(q_1) \\ \sin(q_1) & 0 & -\cos(q_1) & 44.28 \times \sin(q_1) \\ 0 & 1 & 0 & 65.35 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Matriz de transformación entre el eslabón 1 y el eslabón 2:

$$A_{12}(\theta_2, d_2, a_2, \alpha_2) \rightarrow \begin{pmatrix} \cos(q_2 - 90^\circ) & -\sin(q_2 - 90^\circ) & 0 & -141.06 \times \cos(q_2) \\ \sin(q_2 - 90^\circ) & \cos(q_2 - 90^\circ) & 0 & -141.06 \times \sin(q_2) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Matriz de transformación entre el eslabón 2 y el eslabón 3:

$$A_{23}(\theta_3, d_3, a_3, \alpha_3) \rightarrow \begin{pmatrix} \cos(q_3 + 90^\circ) & -\sin(q_3 + 90^\circ) & 0 & 1300 \times \cos(q_3) \\ \sin(q_3 + 90^\circ) & \cos(q_3 + 90^\circ) & 0 & 1300 \times \sin(q_3) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Matriz de transformación entre el eslabón 3 y el eslabón 4:

$$A_{34}(\theta_4, d_4, a_4, \alpha_4) \rightarrow$$

$$\begin{pmatrix} \cos(-90) & 0 & \sin(-90) & 21.44 \times \cos(-90) \\ \sin(-90) & 0 & -\cos(-90) & 21.44 \times \sin(-90) \\ 0 & 1 & 0 & 30.62 \\ 0 & 0 & 0 & 1 \end{pmatrix} \times \begin{pmatrix} \cos(180) & -\sin(180) & 0 & 21.44 \times \cos(180) \\ \sin(180) & \cos(180) & 0 & 21.44 \times \sin(180) \\ 0 & 0 & 1 & 30.62 \\ 0 & 0 & 0 & 1 \end{pmatrix} =$$

$$\begin{pmatrix} 0 & 0 & -1 & -30.62 \\ 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 30.62 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Matriz de transformación entre el eslabón 4 y el eslabón 5:

$$A_{45}(\theta_5, d_5, a_5, \alpha_5) \rightarrow \begin{pmatrix} \cos(q_4) & -\sin(q_4) & 0 & 0 \\ \sin(q_4) & \cos(q_4) & 0 & 0 \\ 0 & 0 & 1 & 5 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

La matriz de transformación entre el eslabón 5 y el sistema de coordenadas de referencia queda de la siguiente forma:

$$T = A_{01}(\theta_1, d_1, a_1, \alpha_1) \times A_{12}(\theta_2, d_2, a_2, \alpha_2) \times A_{23}(\theta_3, d_3, a_3, \alpha_3) \times A_{34}(\theta_4, d_4, a_4, \alpha_4) \times A_{45}(\theta_5, d_5, a_5, \alpha_5)$$

## Capítulo 5. Impresión 3d

La tecnología 3D está revolucionando el mundo tal y como se conoce permitiendo a cualquier usuario con conocimientos básicos de diseño obtener modelos reales de piezas prototipadas reduciendo el coste de los prototipos. En la impresión 3D confluyen un grupo de tecnologías que permiten crear un objeto tridimensional mediante la adición de material mediante capas sucesivas. Las impresoras 3D son baratas, fáciles de utilizar y para la impresión de un prototipo son relativamente rápidas en comparación con otras técnicas de fabricación industrial. Cuando el número de elementos a imprimir supone una cantidad considerable, las técnicas de inyección de plástico con moldes reducen el tiempo de fabricación, una menor tolerancia y la repetitividad es mayor al tratarse de un único molde invariable. El crecimiento mayoritario del uso y venta de las impresoras 3D aumentó en 2003 con la consecuente reducción del coste de éstas. Ampliamente usadas en el campo de la ingeniería, joyería, calzado, diseño industrial, arquitectura, automoción, medicina e incluso alimentación.

### 5.1 Proceso

Primeramente, la pieza se diseña mediante un software CAD de diseño 3D, una vez la pieza esté finalizada, se exporta en el formato de archivo STL (*STereo Lithography*). El formato STL es el tipo de archivo el cual las impresoras, o el software de impresión permite identificar la geometría de la pieza despreciando el color, textura, o comportamiento físico de la pieza, información que incluyen otros formatos. Este archivo permite obtener la geometría de la pieza diseñada mediante devanados/facetas triangulares desestimando información inútil para la impresora y reduciendo el tamaño del archivo para mejorar la carga y la lectura en las impresoras y por tanto reduciendo el cálculo computacional. Cuanto menor es el tamaño de la faceta, se respeta con mayor medida la geometría original de la pieza diseñada y por tanto la calidad de ésta.

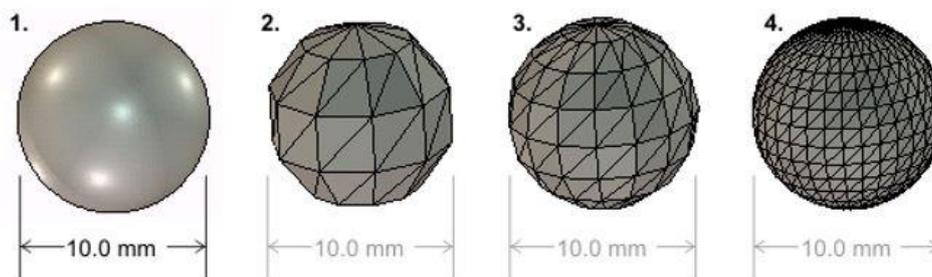


Ilustración 93 1: Esfera original; 2,3,4: Esfera convertida en STL con orden creciente en calidad, orden descendiente en tamaño de las facetas triangulares

Una vez exportado el archivo con extensión STL se abre utilizando el software de impresión proporcionado por la impresora 3D que se esté utilizando. Este software descompone la pieza en cortes o capas horizontales paralelas. Entre otros en el software se determinan las características de la impresión, como pueden ser el relleno, el suavizado, el tipo de adhesión, la anchura de las

paredes, el entramado interno etc. Una vez configuradas las características para la pieza, se genera un archivo de código máquina donde se encuentran las coordenadas, movimientos y acciones que debe realizar la máquina. El software convierte un modelo digital en instrucciones de impresión.

Se procede a abrir el archivo creado y una vez confirmado el inicio de la impresión la máquina realizará todas las acciones que se encuentren en el archivo generando un modelo 3D de la pieza. La máquina comenzará la impresión realizando capas horizontales de material creando la pieza hasta que el proceso esté completo.

## 5.2 Métodos

Existen diferentes métodos y tecnologías para la obtención de modelos 3D mediante impresión, las diferencias entre estos métodos son fundamentalmente basadas en la forma en la que las capas que conforman la pieza son unidas. Cada método posee ventajas e inconvenientes por lo que, dependiendo del material de fabricación de la pieza, velocidad, coste del material, color, geometría o uso de la pieza se recomienda un método u otro. En la siguiente tabla se pueden observar la comparación entre los diferentes tipos, tecnologías y materiales

<b>Tipo</b>	<b>Tecnologías</b>	<b>Materiales</b>
Extrusión	Modelado por deposición fundida	Termoplásticos metales eutécticos y materiales comestibles
Hilado	Fabricación por haz de electrones	Casi cualquier aleación
Granulado	Sinterizado directo de metal por láser, por calor o proyección de aglutinante	Casi cualquier aleación
Laminado	Laminado de capas	Papel, papel de aluminio, capa de plástico
Fotoquímicos	Estereolitografía	fotopolímero

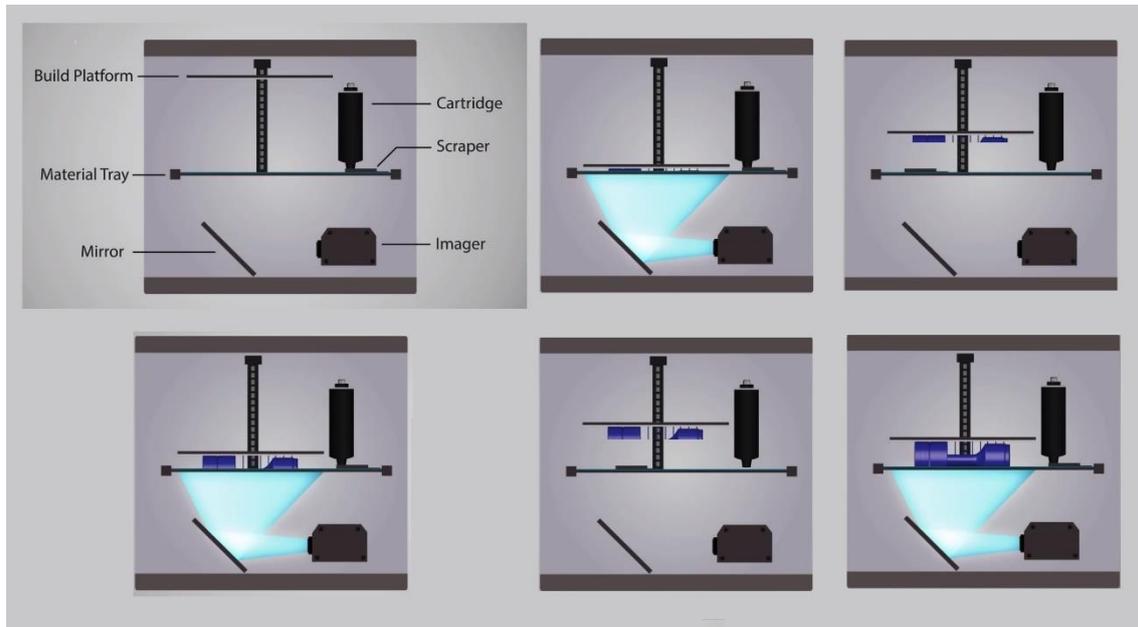
De entre todos los tipos de impresión 3D, se destaca el modelado por deposición fundida y la estereolitografía ya que representan los sistemas habituales de impresión 3D en las impresoras convencionales de escritorio.

### 5.2.1 Estereolitografía

Se trata de una tecnología que utiliza resinas líquidas foto poliméricas. Estas resinas tienen la propiedad de solidificarse cuando son expuestas a un haz de luz ultravioleta emitido por un láser. De este modo se crean capas superpuestas de resina solidificada que en su conjunto van creando el objeto. En las impresoras 3D estereolitográficas, la impresión se suele realizar rotada hacia abajo permitiendo que el exceso de resina líquida no solidificada se vuelva a depositar en el recipiente contenedor.

La impresora 3D estereolitográfica se compone de:

- Láser: proyecta la luz ultravioleta, el láser manda la forma de la capa mediante luz
- Espejo: proyecta y refleja la luz del láser hacia la superficie de impresión
- Bandeja o depósito de material: se encuentra de forma líquida la resina acoge el exceso de resina de la pieza.
- Plataforma de impresión: se trata de una superficie con un sistema que permite ascender y descender, de esta forma la superficie aumenta en altura conforme se van creando las capas de resina solidificada.

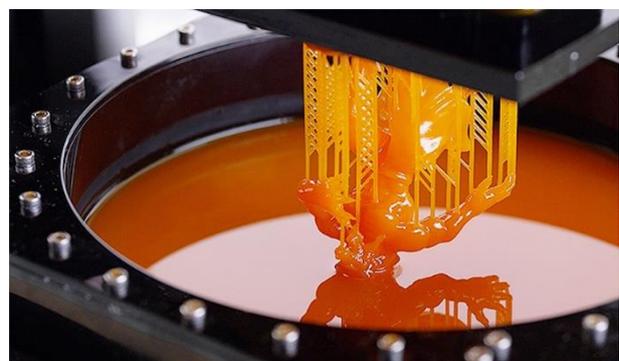


*Ilustración 94 Funcionamiento y partes de una impresora estereolitográfica*

Materiales:

- Resina blanca opaca
- Resina blanca ABS Especial con infiltraciones para mejorar sus propiedades mecánicas
- Resina translúcida.

La principal ventaja de este método es la rapidez frente a otras alternativas. El acabado final de las piezas es suave y preciso, no se generan restos sólidos y la pieza queda endurecida e impermeable. Por el contrario, pueden quedar restos de resina no endurecida, semisólida que generan acabados pegajosos. Aunque resisten los esfuerzos mecánicos, no son la mejor opción para el diseño de piezas mecánicas y las propiedades de resistencia disminuyen con el paso del tiempo debido a su exposición con el medio y la radiación solar.

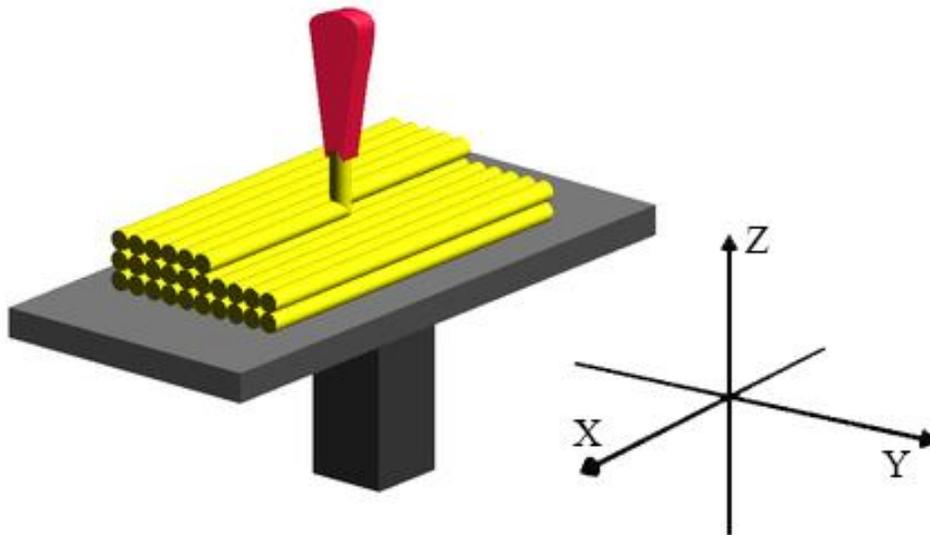


*Ilustración 95 Impresión 3D estereolitográfica*

## 5.2.2 Modelado por deposición de fundente

Se trata de un método de impresión 3D utilizado para prototipos o producción de modelos a pequeña escala. Se basa en el sistema de impresión por inyección. Es una técnica aditiva en la que se deposita material en capas, la sucesión de las capas genera la pieza. La técnica se basa en las propiedades de fundición de los materiales termoplásticos, estos materiales son fundidos y empujados por una boquilla, la máquina mueve la boquilla para dibujar las capas y se deposita el material encima de la plataforma, rápidamente el material fundido se solidifica y se procede a dibujar la siguiente capa de material sobre la capa solidificada.

La impresoras 3D de modelado por deposición fundente o MDF suelen ser de tipo cartesianas donde la plataforma se mueve en eje Z y es la boquilla la que se mueve en los ejes X e Y, también es común ver que la plataforma se mueva en eje X y sea la boquilla quien se mueva en Y y Z.

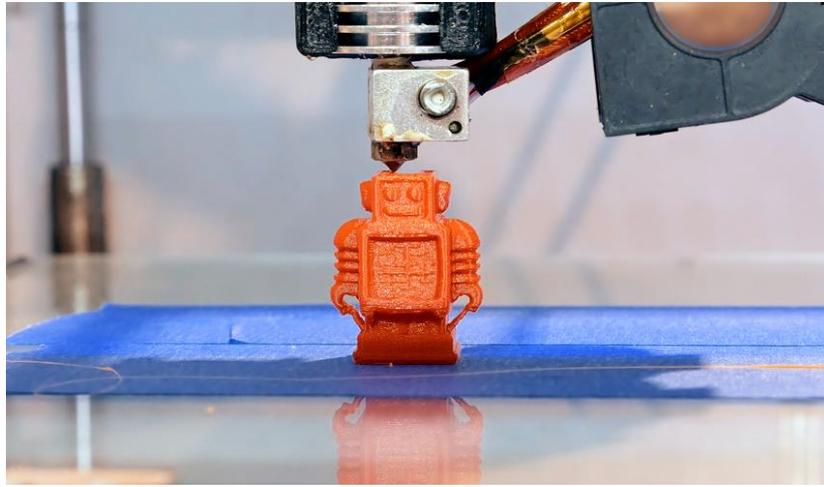


*Ilustración 96 Esquema de deposición del fundente con sistema de coordenadas*

Algunos de los materiales usados son los siguientes:

- Acrilonitrilo butadieno estireno (ABS)
- Poliláctico (PLA)
- PolicarbonatoPolicaprolactona (PCL)
- Polifenilsulfona (PPSU)
- Polieterimida (PEI)
- Ceras
- Chocolate y otros alimentos para uso en repostería
- Acetato de polivinilo (PVA) utilizado para soportes hidrosolubles

Esta técnica es ampliamente utilizada en la industria del prototipado y fabricación rápida debido a su alta velocidad de impresión. Los materiales son baratos y en el caso del PLA incluso biodegradables. Es la única tecnología capaz de crear modelos a todo color y que permite crear extraplanos o salientes.



*Ilustración 97 Impresión de una pieza con salientes*

Dado su bajo coste de producción, los precios de las máquinas son relativamente económicos, y por tanto se ha popularizado mucho a nivel doméstico. Cabe mencionar RepRap, se trata de un proyecto con una gran comunidad detrás que tiene la iniciativa de crear una máquina autorreplicable. Es decir que una impresora 3D sea capaz de crear piezas para construir otras impresoras. Este proyecto se popularizó mucho entre los jóvenes creadores dando lugar a la creación de impresoras DIY que abaratan el coste permitiendo que entusiastas de la tecnología tuviesen impresoras 3D en sus hogares. Entre todas las impresoras en el proyecto RepRap, se destaca la Prusa i3 que se ha consolidado como la impresora más replicada en los últimos años.



*Ilustración 98 Impresora RepRap Prusa i3*

## 5.3 Impresión de las piezas del robot

### 5.3.1 Impresoras elegidas

Las impresoras que se van a utilizar para la realización de las piezas son:

BQ Prusa i3 Hephestos: es una reedición de la famosa impresora Prusa i3 desarrollada por el departamento de innovación y robótica de BQ. Añade mejoras de estabilidad y firmware además de la modificación de piezas para una mayor robustez. Se trata de una impresora de hardware y software libre. Impresora propia.

Zortrax M200: impresora plug and play, la impresora no requiere de montaje simplemente se conecta a la alimentación y comienza a funcionar. Capaz de auto calibrarse y con precisión de 400 micras, se trata de una impresora potente y silenciosa con gran acabado en las piezas. Impresora disponible en el departamento de impresión de la ETSID.



Ilustración 99 Hephestos BQ

### 5.3.2 Material seleccionado

El material elegido es el PLA, al tratarse de un material no contaminante, biodegradable y que permite la utilización en entornos cerrados como estudios sin suponer un peligro para la salud de los que se encuentran en el entorno. Concretamente los dos materiales que se utilizarán serán:

PLA BQ: filamento de 1.75mm bobinado de calidad para la obtención de piezas con excelente acabado. Material 100% compatible con la Hephestos. El color usado será el negro.

Z-Ultrat: material no contaminante diseñado para las Zortrax, es el más eficiente y completo del mercado. Proporciona un acabado excelente a las piezas, es resistente y con buenas propiedades mecánicas. Será usado por la impresora Zortrax M200. El color dependerá de la disponibilidad de material en el laboratorio.



Ilustración 100 Filamentos PLA de BQ y Z-Ultrat de Zortrax

### 5.3.3 Exportación del modelo

Primeramente, se deben exportar cada una de las piezas del robot en el formato STL anteriormente descrito. Dentro del SolidWorks, hacemos click en “Archivo”, posteriormente en “Guardar como”, una vez se despliegue la pestaña de guardado, se elige un nombre y en “Tipo” elegimos “STL”, una vez guardado aparecerá una ventana emergente y el modelo adquiere un esqueleto formado por triángulos como se muestra a continuación:

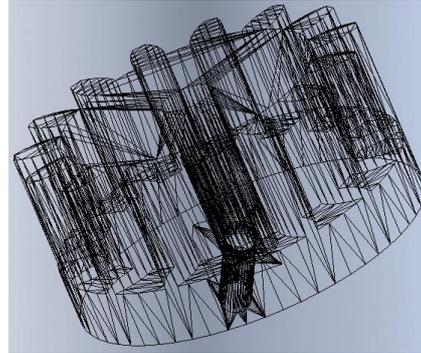
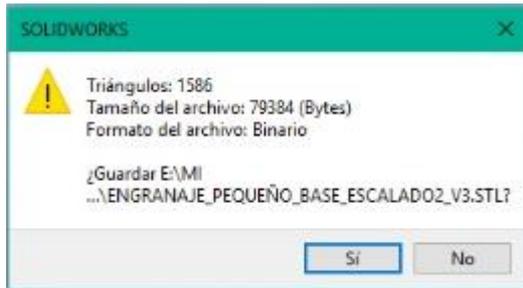


Ilustración 101 Ventana de exportación STL en SolidWorks y esqueleto conformado por las facetas triangulares

El dato de triángulos indica la cantidad de facetas triangulares que componen la pieza que se va a generar, SolidWorks calcula por defecto la cantidad necesaria de facetas para obtener una pieza con alta calidad. Estas características se pueden modificar en las propiedades del programa.

Una vez generada la pieza en STL se procede a abrir el software de impresión para convertir el modelo en instrucciones de impresión. Entre los distintos softwares de laminación destacamos los más utilizados:

- Slic3r: es una plataforma de código libre con una gran comunidad de usuarios que trabajan para mejorar las características del software cada día. Fue la herramienta precursora del resto de softwares de laminación.
- Cura: se trata de un software gratuito de la marca *Ultimaker* con compatibilidad con practicante cualquier impresora del mercado, un software muy potente y completo.
- Repetier-Host: entre otros el software tiene una herramienta de laminado que es un plugin de Slic3r, además se puede observar el proceso de laminación mediante simulación y modificar directamente el código.
- Simplify3D: herramienta de pago con infinidad de posibilidades, la más potente en el mercado, permite analizar puntos críticos en la impresión y prestar soluciones para corregir los posibles problemas.
- Z-Suite: se trata del software de laminación para las impresoras Zortrax, es gratuito.

Una vez se haya obtenido el modelo laminado, este puede introducirse directamente en la máquina si dispone de ranura para alojar una tarjeta SD o se requiere de un host que controle la impresión. Un programa host en impresión se trata de un software que controla completamente cualquier parámetro de la impresora, siendo ésta completamente esclava del software. Sirve principalmente

para calibrar los ejes y sensores y comprobar el correcto funcionamiento de la mecánica de la máquina. Se pueden importar archivos laminados y sirve para transmitir las ordenes de impresión.

Existen muchas opciones entre las que destacamos tres:

- **Pronterface:** software de licencia libre, simple pero muy potente, compatible con casi el 100% de las impresoras. Permite controlar toda la máquina y la impresión completa de una pieza. Es robusto pero austero.
- **Cura:** aparte de ser un software de laminación, Cura tiene la posibilidad de controlar la impresión lo que facilita que con un único programa se importe la pieza y se imprima sin más software de por medio. Sencillo, intuitivo y potente. Menor capacidad de control sobre la impresora que Pronterface. La última versión permite la impresión a distancia vía internet.
- **Repetier-Host:** de igual forma que Cura combina el software de laminación con software host. Tiene menor compatibilidad con impresoras que los anteriores.
- **Slic3r:** a igual que sus programas compañeros, se está desarrollando una versión que permita ser host de la impresora, pero necesita madurar.

### 5.3.4 Laminación de las piezas

Se elige el software de Cura para la impresión de piezas con la Hephestos al tratarse de un programa completo, intuitivo y fácil de utilizar que combina la laminación y el servicio host. En el caso de la Zortrax M200 se utilizará el software propio de impresión, el Z-Suite y se introducirán los archivos directamente a la impresora mediante memoria SD.

El proceso es similar en ambos softwares, se hace el uso del Cura como ejemplificación. Importamos la pieza en Cura y en la parte derecha observamos las características y propiedades de impresión que definirán el proceso.

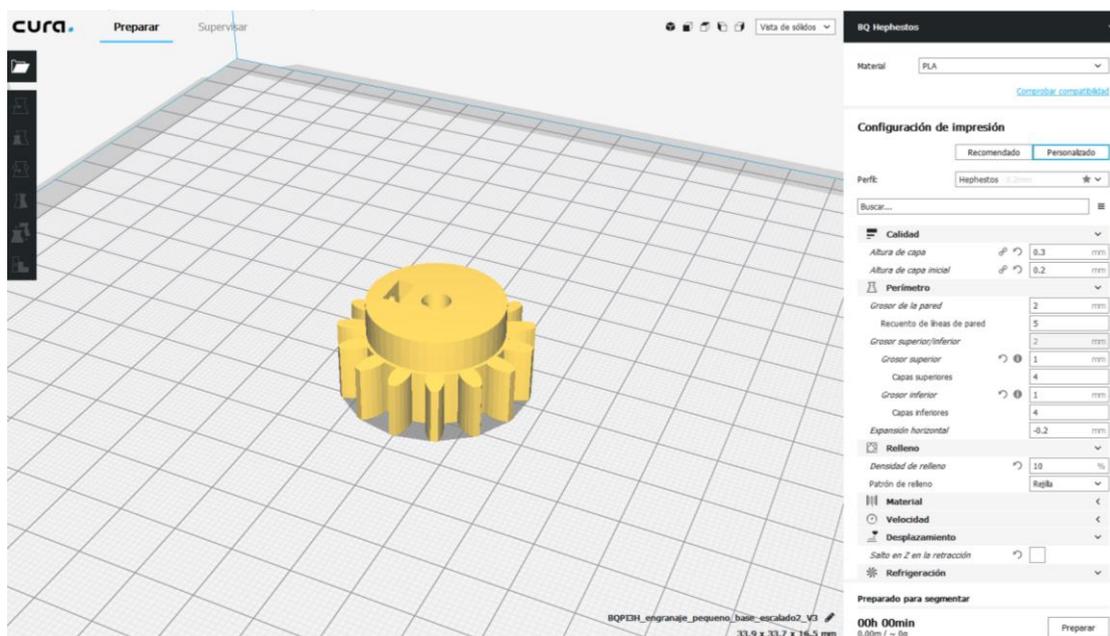


Ilustración 102 Software Cura

Las propiedades principales a definir son:

- **Altura de capa:** se trata de un parámetro crucial que define la calidad final del objeto. Esta propiedad define la altura existente entre capa y capa, cuanto más pequeña es, se aprecia en menor medida el salto entre capas y por tanto el acabado final es más suave. Alturas de capa pequeñas mejoran la calidad, pero requieren de un mayor tiempo de impresión. Alturas altas reducen tiempo de impresión sacrificando un acabado detallado.



Ilustración 103 Piezas impresas con diferentes alturas de capa con sus correspondientes tiempos de impresión y calidad final

- **Grosor de la pared:** define los milímetros de espesor entre la pared externa y el relleno de la pieza, si las paredes son más delgadas el tiempo de impresión es menor pero la pieza genera estrías en sus paredes externas debido al relleno de la pieza, las estrías producen un acabado de menor calidad.

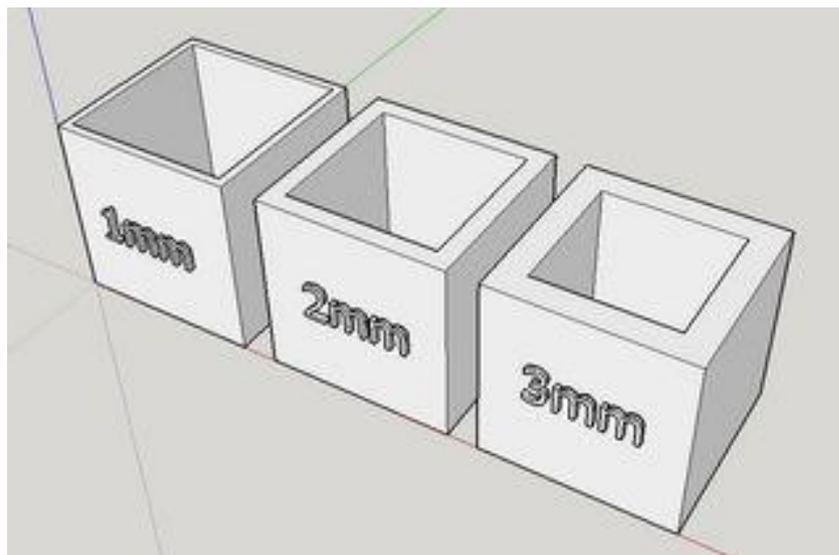
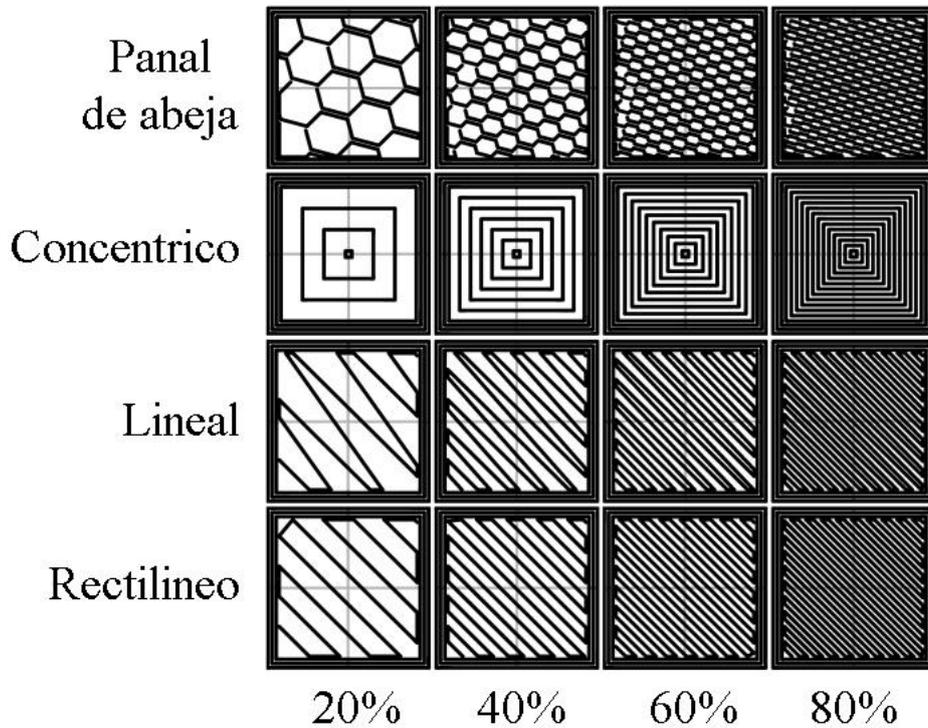


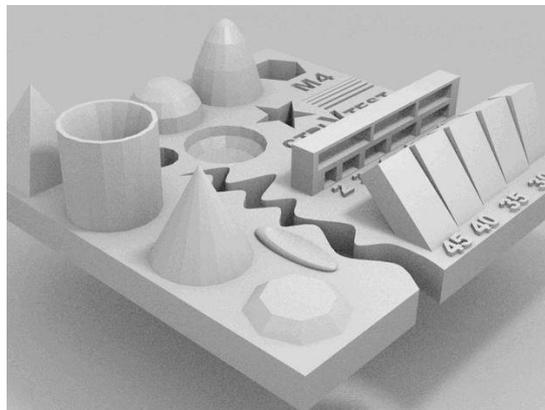
Ilustración 104 Diferentes grosores de pared

- Relleno: indica el porcentaje de materia y de hueco interno que tendrá la pieza, repercute directamente en el tiempo de impresión, piezas con poco relleno disminuyen el tiempo de impresión, pero a su vez merma la resistencia mecánica de la pieza. Piezas diseñadas para soportar grandes esfuerzos requieren de mayor porcentaje de densidad de relleno. A parte de elegir el porcentaje de relleno, se puede elegir el tipo de relleno, es decir la forma que realiza el relleno.



*Ilustración 105 Tipo de relleno y porcentaje de relleno*

- Expansión horizontal: las propiedades termoplásticas de los plásticos utilizados provocan que al enfriarse las piezas estas se retraigan lo que ocasiona tolerancias entre la pieza diseñada y la pieza impresa. Utilizando un patrón de calibración se calcula la expansión horizontal que compensa la tolerancia y se reduce el error.



*Ilustración 106 Patrón de calibración*

- Soporte: ciertas piezas contienen salientes o cavidades huecas, para facilitar y obtener una mayor precisión en estas partes se puede elegir la creación de soporte. Si se elige la opción, Cura calcula la localización y cantidad necesaria automáticamente. Una vez impresa la pieza se puede retirar fácilmente el soporte.



*Ilustración 107 Generación de soporte*

- Refrigeración: para un buen acabado en materiales con fundición a bajas temperaturas se requiere de un sistema que enfríe las capas. Las impresoras presentan sistemas de ventiladores que enfrían en material por capas.



*Ilustración 108 Izquierda: pieza de PLA impresa sin refrigeración, Derecha: pieza con refrigeración*

- Adherencia de la pieza: ciertas impresoras poseen una plataforma de impresión calefactable lo que evita que las piezas se separen de ésta en mitad del proceso. Existen además sistemas de adherencia que aseguran una correcta fijación a la plataforma:
  - Raft: se crea una cama de material de poco relleno que se fija fuertemente a la superficie de impresión. A su vez sirve para asegurar una adherencia correcta ya que la impresión se realiza encima de la cama.

- Skirt: no se trata directamente de un sistema de adherencia, es un sistema que genera perímetros exteriores que rodean a la pieza para delimitar su geometría, la misión de esta herramienta es limpiar el cabezal e iniciar una impresión estable.
- Brim: imita el funcionamiento del Skirt, pero en este caso el Brim se encarga de crear una falda o primera capa con 100% de relleno lo que aumenta la superficie de adherencia y el contacto con la plataforma de impresión

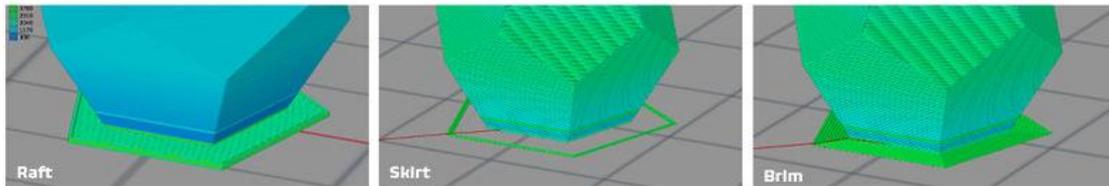


Ilustración 109 Diferentes tipos de adherencia a la plataforma de impresión

Cura calcula una estimación del tiempo de impresión y la cantidad de material.

La cantidad de piezas a imprimir suma un total de 20. Se van a imprimir las piezas de mayor tamaño que a su vez requieren mayor precisión con la Zortrax M200 utilizando el material Z-Ultrat. Las piezas de menor tamaño se realizarán en la impresora del estudio Hephestos. Las piezas mecánicas como engranajes, barras o brazos serán impresos con la mayor resolución y con un alto porcentaje de relleno puesto que son las piezas destinadas a soportar grandes esfuerzos mecánicos. Piezas secundarias, soportes y demás se imprimirán con menor detalle y relleno para economizar el tiempo.

A continuación, se muestra una tabla donde aparecen las piezas junto con las propiedades de altura de capa, relleno, material y tiempo de impresión.

Nombre pieza	Altura capa	Relleno	Material	Tiempo impr.	Cantidad Mat.
Antebrazo	0.14 mm	35%	Z-Ultrat	6h, 37 min	77g
Brazo	0.14 mm	35%	Z-Ultrat	13h, 38 min	63g
Corona base	0.14 mm	35%	Z-Ultrat	7h, 2 min	84g
Base	0.14 mm	35%	Z-Ultrat	8h, 7 min	74g
Piñón base	0.2 mm	30%	PLA BQ	49 min	8g
Piñón motor (2pcs)	0.2 mm	30%	PLA BQ	23 min	3g
Corona 4 agujeros	0.2 mm	30%	PLA BQ	1h, 30 min	13g
Corona con barra	0.2 mm	30%	PLA BQ	1h,51 min	16g
Barra enlace triang.	0.2 mm	30%	PLA BQ	46 min	7g
Enlace triangular	0.2 mm	30%	PLA BQ	49 min	7g
Barra triang-base	0.2 mm	30%	PLA BQ	54 min	8g
Barra enlace antebr.	0.2 mm	30%	PLA BQ	51 min	7g
Lateral izq. muñeca	0.2 mm	30%	PLA BQ	51 min	8g
Lateral der. muñeca	0.2 mm	30%	PLA BQ	55 min	8g
Separador cilíndric.	0.2 mm	30%	PLA BQ	22 min	4g
Zócalo servomotor	0.2 mm	30%	PLA BQ	48 min	7g
Soporte brazo	0.3 mm	20%	PLA BQ	1h, 39min	26g
Lateral (2pcs)	0.3 mm	20%	PLA BQ	2h, 1 min	22g

## Capítulo 6. Software

El software que se va a utilizar para desarrollar la aplicación que controlará el brazo será Labview. Labview propiedad de National Instruments, es un software de ingeniería destinado al diseño de aplicaciones que requieren pruebas, medidas, control, acceso rápido a datos y a hardware. El enfoque de programación es gráfico, permite visualizar cada aspecto de la aplicación de manera intuitiva y visual. Posibilita la creación de GUIs (*Graphical User Interface*) o interfaz gráfica de usuario, entornos visuales en el que el operador tiene el control del sistema que se requiere supervisar con actuadores virtuales en forma de botones, diales... y visualización de datos mediante gráficas, cursores... El principal propósito es diseñar un entorno virtual y sencillo que permita la comunicación entre el ordenador y la máquina a controlar.

El software es ideal para sistemas de hardware y software tanto simulado como real, y sistemas embebidos, recomendado porque acelera la productividad. El lenguaje se llama lenguaje G (*Graphic*).

Entre otras muchas destacamos las características más importantes de este software:

- Interfaces de comunicaciones compatibles:
  - Puerto serie
  - Puerto paralelo
  - GPIB
  - PXI
  - VXI
  - TCP/IP, UDP, DataSocket
  - Irda
  - Bluetooth
  - USB
  - OPC...
- Capacidad de interactuar con otros lenguajes y aplicaciones:
  - DLL: librerías de funciones
  - .NET
  - ActiveX
  - Multisim
  - Matlab/Simulink
  - AutoCAD, SolidWorks, etc
  - C++
- Herramientas gráficas y textuales para el procesado digital de señales.
- Visualización y manejo de gráficas con datos dinámicos.
- Adquisición y tratamiento de imágenes.
- Control de movimiento (combinado incluso con todo lo anterior).
- Tiempo Real estrictamente hablando.
- Programación de FPGAs para control o validación.
- Sincronización entre dispositivos.

## 6.1 Funcionamiento LabVIEW

A la hora de creación de un nuevo documento en Labview, se muestran las diferentes opciones con sus correspondientes características. Dependiendo el tipo de proyecto al que vaya destinada la aplicación se elige la opción:

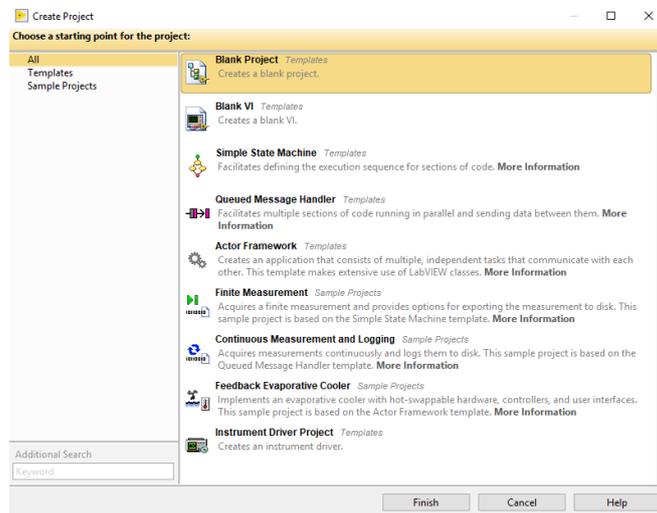


Ilustración 110 Ventana de creación de proyectos

La opción principal para la creación de un programa es la de “Blank VI”, crea un documento de programación para ser editado por el usuario. La extensión del documento es “.vi”. El documento consta de dos partes diferenciadas:

- **Panel Frontal:** es la interfaz con el usuario, en este panel se diseña el entorno visual que será destinado al usuario final. Se podrán observar datos en tiempo real de diferentes formas (gráficos, indicadores, pilotos...) e interactuar con el sistema accionando botones, cursores, diales etc. El panel frontal se caracteriza por poseer una cuadrícula y un fondo gris oscuro.

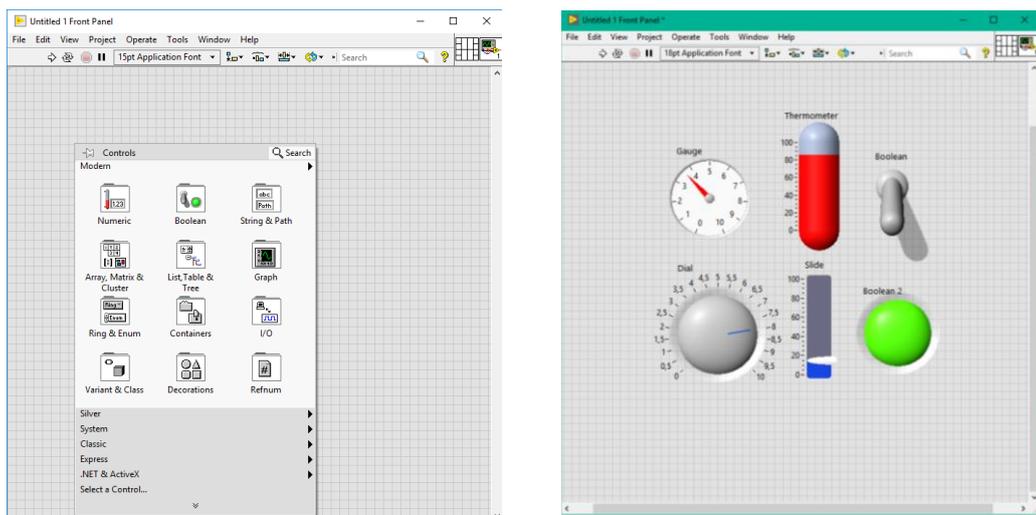
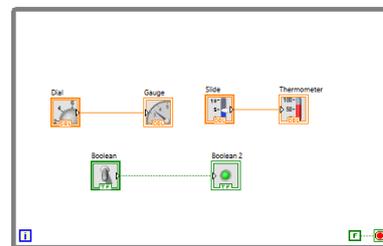
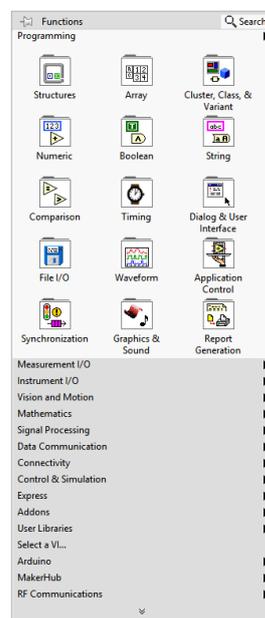


Ilustración 111 Menú de controles y panel frontal con distintos controles

Haciendo click derecho se despliega el menú de controles donde se agrupan según su comportamiento. Para insertar un control en el panel frontal basta con elegirlo del menú de controles y arrastrarlo al panel.

- **Diagrama de Bloques:** se encuentra el programa principalmente, es decir mediante bloques se define el funcionamiento del sistema que se verá reflejado en los controles del panel frontal. Los bloques siguen una lógica y manejan variables de distinto tipo. Los diferentes bloques se interconectan y junto con estructuras tipo bucle y condicionales se va generando el código. De igual forma que en el panel frontal, si se hace click derecho se despliega un menú complejo de funciones, en este menú se reflejan estructuras de bucles, arrays, sistemas de cálculo numérico, temporizadores, sistemas de lectura y escritura de archivos etc.



A parte de las funciones que incluye el propio software, existen *plug-ins* o complementos que se pueden descargar y añadir fácilmente mediante la herramienta “VI Package Manager” incluida en el software de National Instruments. Como se va a hacer uso de la tarjeta Arduino UNO para controlar el robot se opta por elegir el control del sistema mediante el “Toolkit de Arduino”.

- **Toolkit de Arduino:** es un kit de herramientas desarrollado por National Instruments para facilitar la conexión y la comunicación de las placas Arduino con el software Labview. Este kit está optimizado con bloques que hacen sencilla una programación visual con el Labview. Para descargarlo se debe abrir el “VI Package Manager”. Una vez abierto se realiza la búsqueda con la palabra clave “arduino”. Aparecen diferentes complementos, se selecciona haciendo doble click en “Labview Interface for Arduino”. Por último se procede a instalar el kit.



Por último, se requiere instalar el driver que permita comunicarse mediante puerto serie desde el LabVIEW al Arduino. Por ello LabVIEW dispone del driver NI-VISA que contiene las configuraciones necesarias para conectarse con diferentes protocolos de comunicación. En el enlace que se muestra a continuación se puede descargar el driver “NI-VISA 17.0”:

<http://www.ni.com/download/ni-visa-17.0/6646/en/>

## 6.2 Programación del brazo robot

Se ha diseñado en LabVIEW el programa que controla cada uno de los ejes del robot. El programa presenta una interfaz gráfica de usuario sencilla e intuitiva dividida en diferentes bloques. Cada bloque se encarga de una tarea específica por lo que se abordará la explicación de funcionamiento del programa en diferentes apartados según el bloque que corresponda.

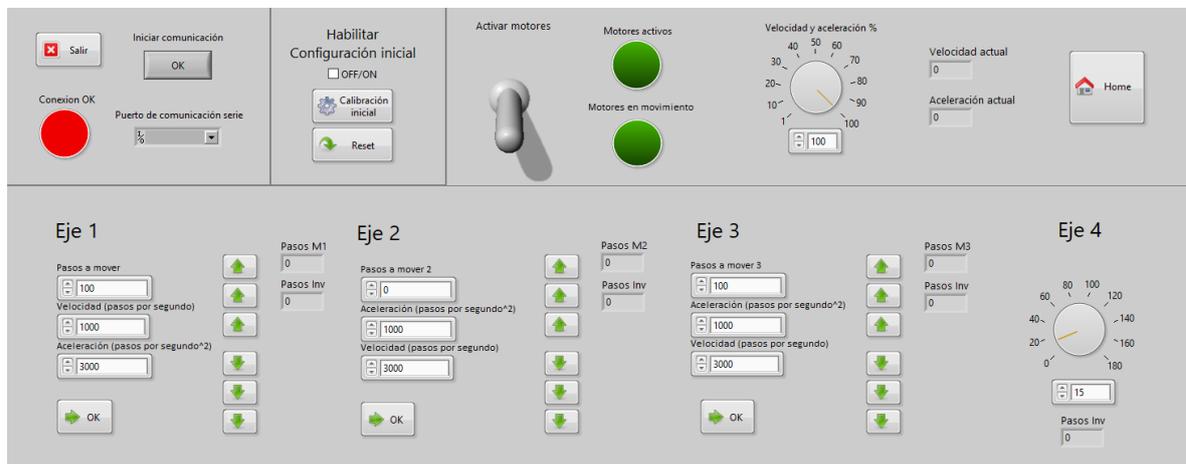


Ilustración 114 Interfaz gráfica de usuario del programa de control del robot

### 6.2.1 Bloque comunicación

Se trata de la sección de código diseñada para iniciar la comunicación con la placa Arduino. Dispone de dos botones, un indicador y un selector. El botón salir como su propio nombre indica permite abandonar el programa principal y cierra la aplicación. Para iniciar la comunicación con la placa, esta debe estar previamente conectada al ordenador. Una vez conectada y con la aplicación abierta, desplegamos la lista de selección “Puerto de comunicación serie” en ella aparecerán todos los puertos de comunicación serie conectados al ordenador. Si el numero de puerto del Arduino no aparece basta con seleccionar “Refresh” en la lista para actualizar los puertos. Una vez se ha seleccionado el puerto de comunicación se hace click en “OK” y se inicia la comunicación.



Ilustración 115 Detalle del bloque de comunicación

La sección de código correspondiente se muestra abajo en la Ilustración 113, en ella se puede observar un bucle infinito que monitorea el botón de inicio de comunicación. Este botón está conectado a la condición de rotura de su propio bucle y a la vez a la entrada del bucle condicional. Una vez se haga click en “OK” la señal mandará la salida de su propio bucle y a la vez activará la condición entrando en el de la derecha. Aparecerá un mensaje emergente que pedirá el reinicio de la placa Arduino debido a que necesita un refresco para asegurar una correcta comunicación con LabVIEW. A su vez activará el indicador tornándolo verde, de este modo se asegura que se ha iniciado la comunicación con la placa.

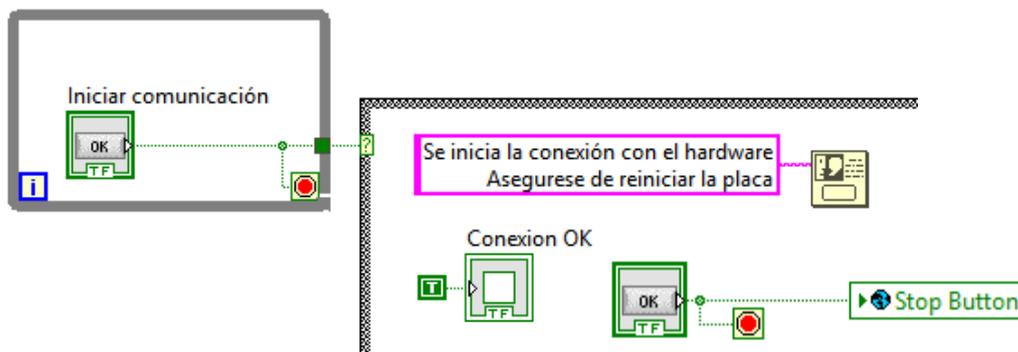


Ilustración 116 Sección de código correspondiente al inicio de la comunicación y cierre de la aplicación

Si se decide salir del programa, el botón “Salir” activa la variable global “Stop Button” y rompe el bucle.

Iniciada la comunicación, el programa se encarga de configurar la placa, en la configuración se recoge el puerto de comunicación serie, y la configuración de los puertos entrada y salida del Arduino. Por defecto el bloque de “INIT” o inicialización de la placa contiene el tipo de placa y la velocidad de comunicación. Por defecto está configurado a 9600 baudios, y el tipo de placa es Arduino UNO. Estas opciones son personalizables y se pueden crear indicadores para controlar dichas variables.

Tras iniciar la placa, observamos tres bloques de configuración de motores paso a paso incluidos en el Toolkit descargado. Estos bloques permiten configurar los pines Step y Dir de los drivers Pololu. Además, permiten elegir el tipo de protocolo de los drivers y asociar un numero de motor. En la configuración del protocolo de los drivers se selecciona “EasyDriver” cuyo protocolo es el

mismo que los drivers DRV8825 utilizados en el presente proyecto. Se asocian los pines que se usarán en la configuración de los drivers. Estos pines son elegidos respetando el esquema electrónico de la “Shield CNC” que indica donde se encuentran conectados. Este esquema se encuentra explicado en el apartado de *Elección del sistema de control*. Existe además un bloque de configuración entradas y salidas para configurar el pin 8 que controla el “enable” de los motores, como salida.

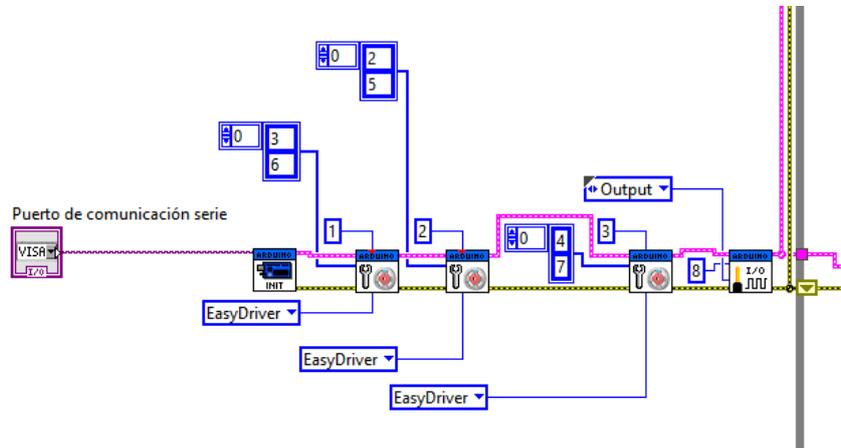


Ilustración 117 Sección de código para la configuración inicial de la placa

## 6.2.2 Bloque control de los motores

Este bloque abarca todas las funciones relacionadas con el movimiento de los motores. Se puede observar que cada motor presenta el mismo diagrama de control por lo que solo bastará explicar un diagrama. Cada motor presenta 3 bloques de texto. En ellos se pueden indicar los pasos a mover, la velocidad y la aceleración. Con el botón “OK” se valida la información integrada dentro de dichos bloques por lo tanto se ejecuta el movimiento del motor. Los botones con las flechas apuntando hacia arriba o hacia abajo sirven para ejecutar movimientos con pasos prefijados. El botón superior permite mover 100 pasos, el siguiente 10, el siguiente 1. Los tres últimos botones de igual forma que los anteriores, pero en este caso negativos. Siendo de arriba hacia abajo, -1, -10 y -100 pasos. Las velocidades y aceleraciones de movimiento vienen fijadas por los bloques de texto anteriormente descritos.



Ilustración 118 Bloque de control de los motores

LabVIEW permite agrupar un número elevado de bloques, funciones y bucles en subVI o sub-bloques. Agrupar todo el sistema en un bloque que permite simplificar y hacer más intuitivo el programa, además que permite con mayor facilidad duplicar acciones. El control de los motores se agrupa en un bloque denominado “MovimientoMotor.vi”, las variables de entrada principales son la velocidad de paso y la aceleración, en la siguiente imagen se muestra:

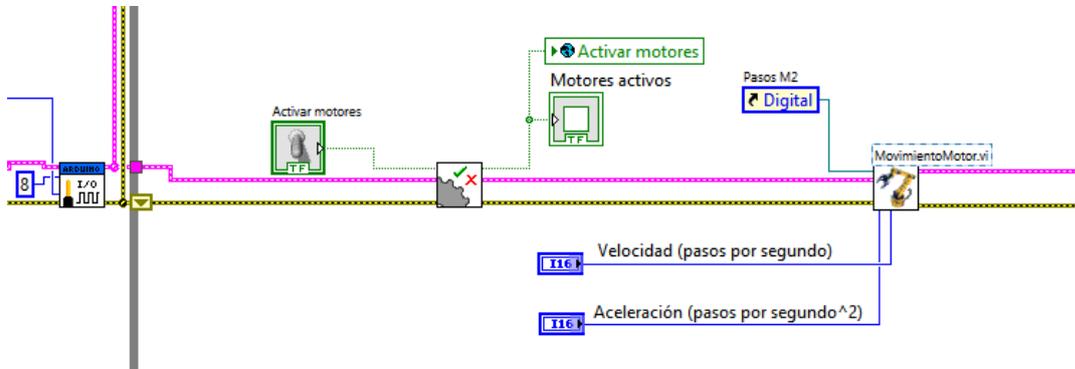


Ilustración 119 Bloque control del movimiento de los motores

Dentro del bloque “MovimientoMotor.vi”, nos encontramos con una estructura compuesta por un bucle condicional accionado por la salida de un bloque situado a la izquierda. Dentro del condicional se encuentra un bucle de instrucciones en código C. Este bucle de código C contiene la programación y movimiento de los motores. En este código se encuentran las siguientes variables:

#### Variables internas

- int32 D1=0; Indica la cantidad de pasos que debe mover el robot, es además una salida
- int32 maxi=448; En ella se introducirá la posición máxima de pasos que puede dar el motor desde la posición de home. En este ejemplo se han determinado 448 pasos de forma empírica al mover el motor 1 del robot.
- int32 min\_val2=0; De igual forma que en el anterior paso, esta variable almacena la posición mínima que puede alcanzar el robot en pasos. Dato obtenido empíricamente.
- int32 forward=0; Indica el número de pasos requeridos para devolver el motor a la posición de home.

#### Variables de entrada y salida

- H1: es una variable de entrada y salida que acumula el valor de la posición del motor en pasos
- AN1: entrada asociada al botón de -1 paso
- BN1: entrada asociada al botón de -10 pasos
- CN1: entrada asociada al botón de -100 pasos
- A1: entrada asociada al botón de 100 pasos
- B1: entrada asociada al botón de 10 pasos
- C1: entrada asociada al botón de 1 paso
- F1: entrada que indica el número de pasos
- J1: entrada asociada al accionamiento del movimiento de pasos
- forward: salida que indica el número de pasos movidos desde la posición de home

El código se puede dividir cuatro en secciones, la primera incluye la declaración de variables.

En la segunda sección se incluye la porción de código que se encarga de comprobar que el motor no se encuentre en la posición mínima. Dentro incluye los diferentes casos de movimiento de pasos negativos, comprobando previamente que el movimiento se puede realizar atendiendo a los valores mínimos de posición del motor.

En la sección 3 de igual forma que en la anterior se comprueba que el valor del motor no haya alcanzado el mínimo. Si no se ha alcanzado el límite, se permite el movimiento con sentido positivo comprobando previamente si el movimiento puede ser alcanzado respetando los límites del motor.

La cuarta sección calcula y devuelve el valor de pasos necesarios para volver a la posición inicial.

```
int32 D1=0;
int32 maxi=1600;
int32 min_val2=-1600;
int32 forward=0;

if(H1>=min_val2){

    if((AN1==1)&&((H1-1)>=min_val2)){
        D1 = -1;
        H1=H1-1;
    }
    if((BN1==1)&&((H1-10)>=min_val2)){
        D1 = -10;
        H1=H1-10;
    }
    if((CN1==1)&&((H1-100)>=min_val2)){
        D1 = -100;
        H1=H1-100;
    }
}

if(H1<maxi){
    if(C1==1){
        D1 = 1;
        H1=H1+1;
    }
    if((B1==1)&&((H1+10)<=maxi)){
        D1 = 10;
        H1=H1+10;
    }
    if((A1==1)&&((H1+100)<=maxi)){
        D1 = 100;
        H1=H1+100;
    }
}

if((J1==1)&&((H1+F1)<=maxi)&&((H1+F1)>=min_val2)){
    D1 = F1;
    H1=H1+F1;
}
forward=-H1;
```

Ilustración 120 Código C, movimiento de los motores

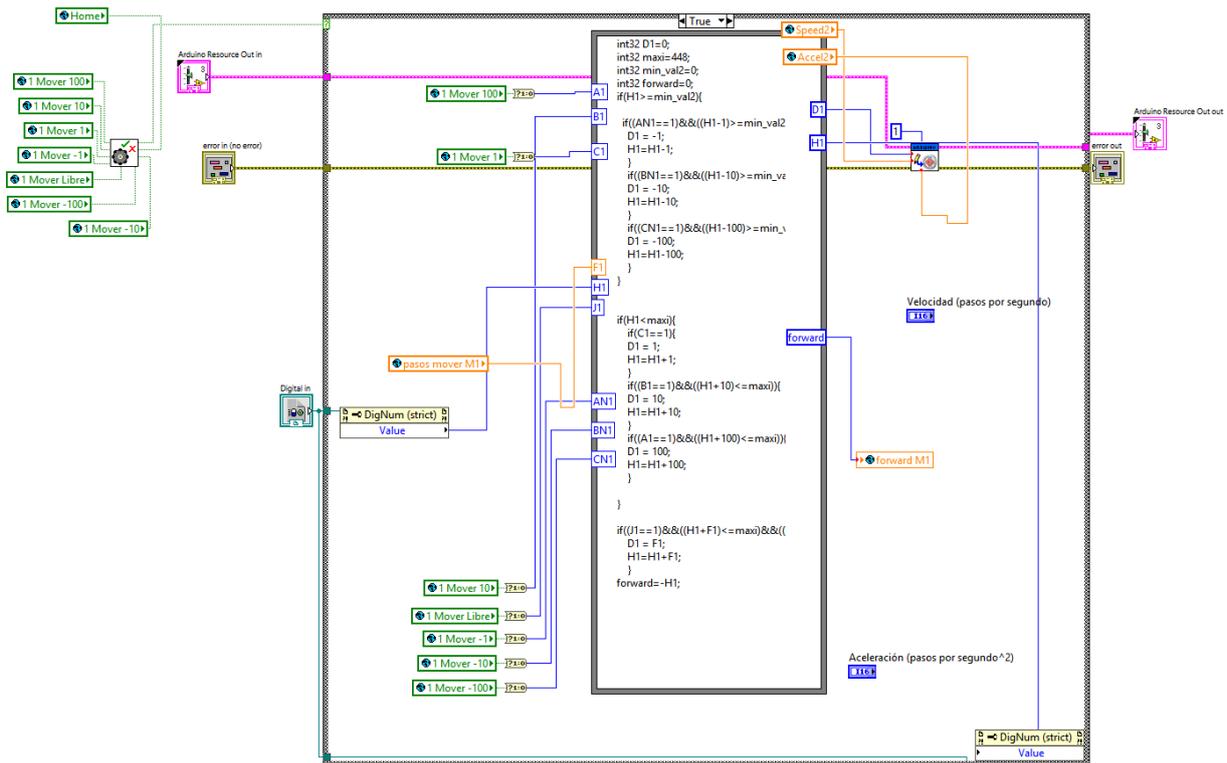


Ilustración 121 Bloque Movimiento de los motores

El bloque de escritura de motor o “Stepper write” situado al final del bloque de movimiento de los motores, lee las acciones de velocidad y aceleración globales, lee los pasos a girar y el motor que se debe accionar. El bloque “Stepper go to” ejecuta las ordenes escritas en el bloque de escritura de motor e informa del estado del motor, tanto si esta parado como moviendose. El estado del motor es registrado por la variable global “Motores en marcha”

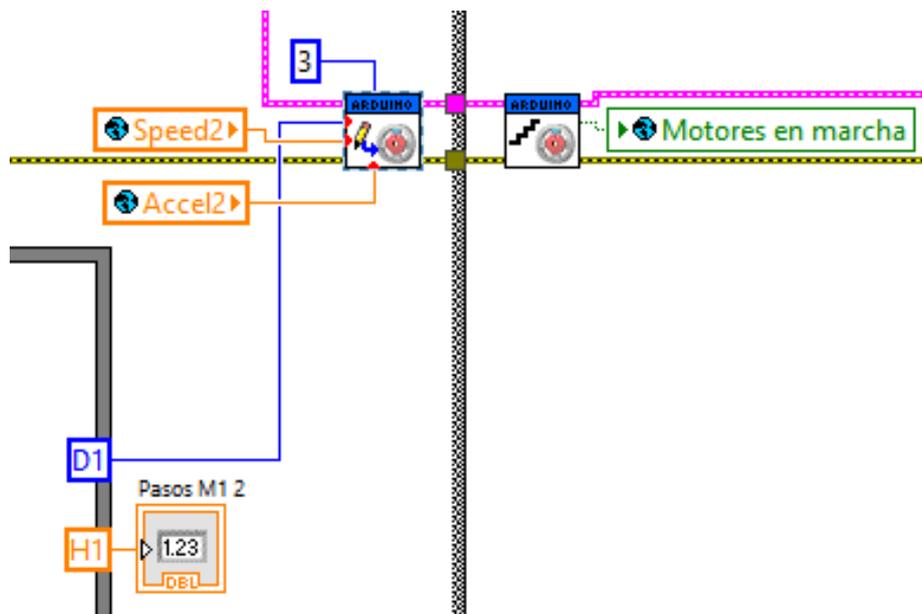


Ilustración 122 Bloque “Stepper write” en el motor PAP y bloque “Stepper Go To”

### 6.2.3 Bloque configuración y estado de los motores

En esta sección se muestran los controles e indicadores relacionados con la habilitación, uso y velocidad relativa de cada motor.

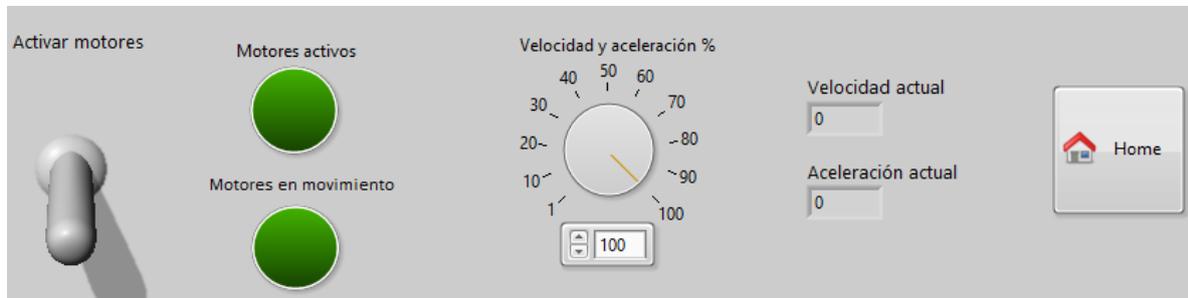


Ilustración 123 Bloque de configuración y estado de los motores, interfaz gráfica

Se observa un interruptor vertical que permite habilitar el uso de los motores. Si se habilita el uso de los motores el indicador de “Motores Activos” se enciende. Se recuerda que un motor habilitado puede ser controlado y cuando está parado el motor opone fuerza. Si se deshabilita la activación de motores, estos pueden moverse de manera manual y se pierde completamente la fuerza de los motores. Si se efectúa esta acción se pierde la referencia de posición, hay que ir con cautela cuando se realiza esta acción. Este interruptor está asociado al sub-VI de habilitación de los motores.

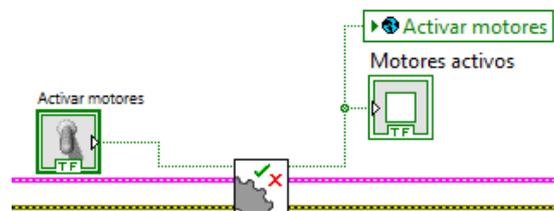


Ilustración 124 Bloque VI habilitación de motores

Se habilitan los motores cuando la salida 8 del Arduino se encuentra en estado alto. Por tanto, en el sub-VI se encuentra una estructura condicional con el bloque que habilita la salida a estado alto del Arduino conectada al “enable” de los drivers Pololu cuando el interruptor es activado. Se desactiva la salida cuando el interruptor se encuentra en estado bajo.

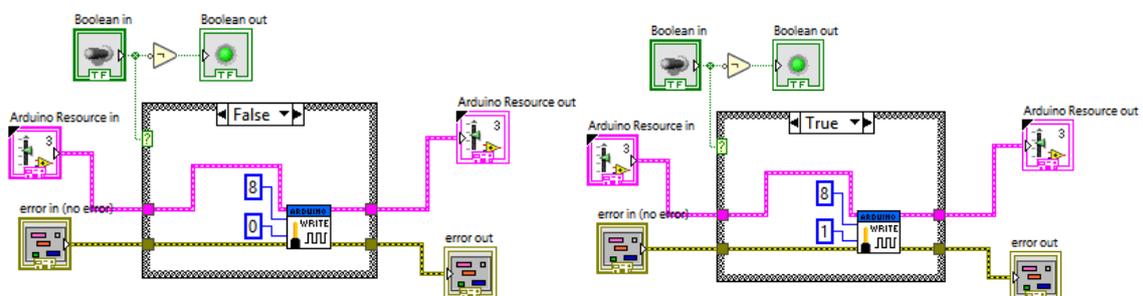


Ilustración 125 Diagrama de bloques del sub-VI de habilitación de motores con los dos estados posibles (true/false)

El indicador de “Motores en movimiento” se enciende cuando alguno de los motores está en marcha y se apaga cuando éstos están detenidos. Se controla mediante el sub-VI denominado “Comprobar\_mov\_ok”. Este sub-bloque gestiona las peticiones de movimiento de los motores y valida la posibilidad de realizar la acción si se cumplen las condiciones de habilitación de motores y que los motores se encuentren parados.

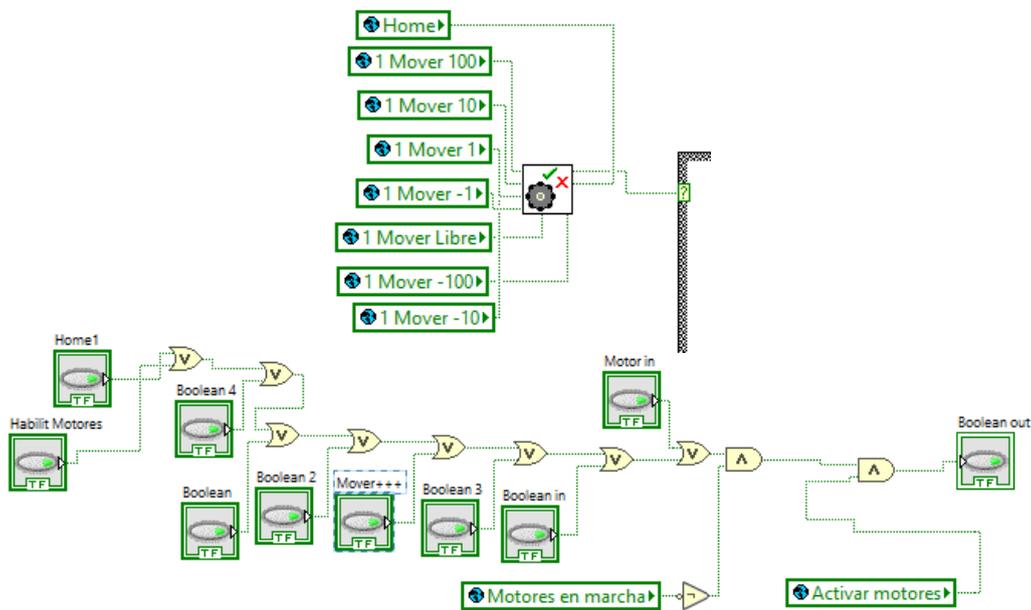


Ilustración 126 Arriba: sub-VI con diferentes entradas; Abajo: configuración interna de bloques del sub-VI

El dial de velocidad y aceleración controla el valor relativo de éstas en base a una proporción que introduce el usuario. Se trata de un sistema que lee las aceleraciones y velocidades de cada uno de los motores y aplica un porcentaje de movimiento. Este porcentaje puede controlarlo el usuario mediante el dial o la caja de introducción de datos. A la izquierda aparece la velocidad (pasos/min) y aceleración (pasos<sup>2</sup>/min) de los motores.

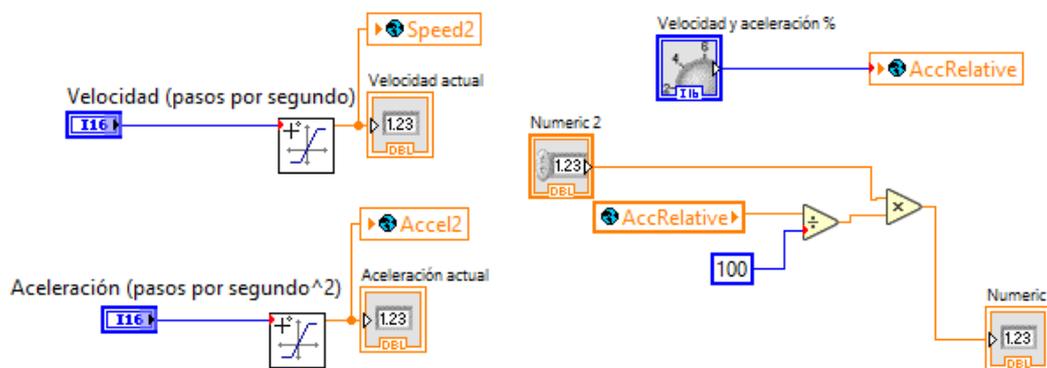


Ilustración 127 Control de velocidades y aceleraciones

## 6.2.4 Bloque configuración inicial

El robot cuando se inicia carece de coordenadas absolutas, es decir sus movimientos son relativos a la posición inicial en la que fue conectado. Para evitar errores de configuraciones incorrectas de ejes y límites, se debe llevar el robot a una posición conocida, en este robot es la posición de “Home”.



Ilustración 128 Bloque de calibración inicial en el GUI

Existen dos opciones:

- **Reset:** es un sistema que borra la posición de los ejes del robot guardados. El proceso de ejecución es, con los motores deshabilitados, mover cada uno de los ejes del robot hacia la posición “Home” conocida. Se realiza click en la caja de verificación y a continuación en “Reset”, posteriormente se desmarca la caja de verificación. De esta forma se reestablecen a 0 las coordenadas de los ejes.

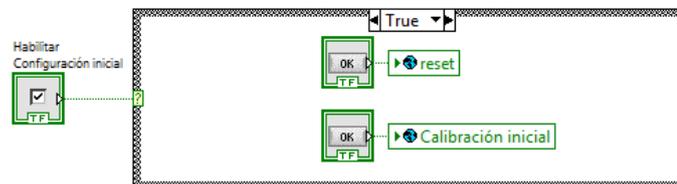


Ilustración 129 Caja de verificación, configuración de bloques

Cuando es oprimido el botón de “Reset” asociado a la variable global, la estructura condicional pone todos los valores de posición de los motores a cero.

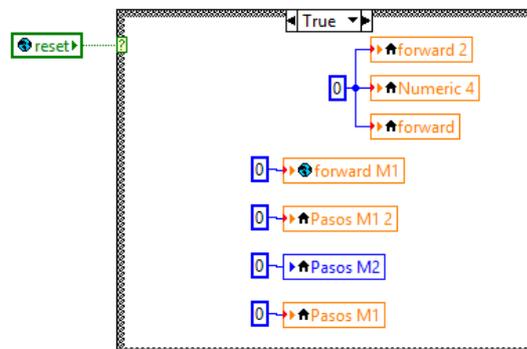


Ilustración 130 Bloque de “Reset”

- Calibración inicial: esta segunda opción permite realizar un calibrado inicial de manera automática, sin necesidad de manipular los ejes gracias a los sensores de posición angular. Para realizar dicha calibración, se pulsa el botón “Calibración inicial” y el sistema lee los sensores de posición y manda corregir las posiciones situando los motores en la posición de home.

Para la lectura de los sensores se hace uso del bloque “Read” del toolkit de Arduino, se encarga de leer el el puerto analógico y devolver un valor de 0-5 voltios. El muestreo se realiza dentro de un bucle infinito controlado por un timer. Dicho timer se puede ajustar al periodo deseado. Se ha añadido una estructura donde se lee y se analiza el valor del potenciómetro para convertirlo en un valor de posición. Esta posición se almacena en una variable global.

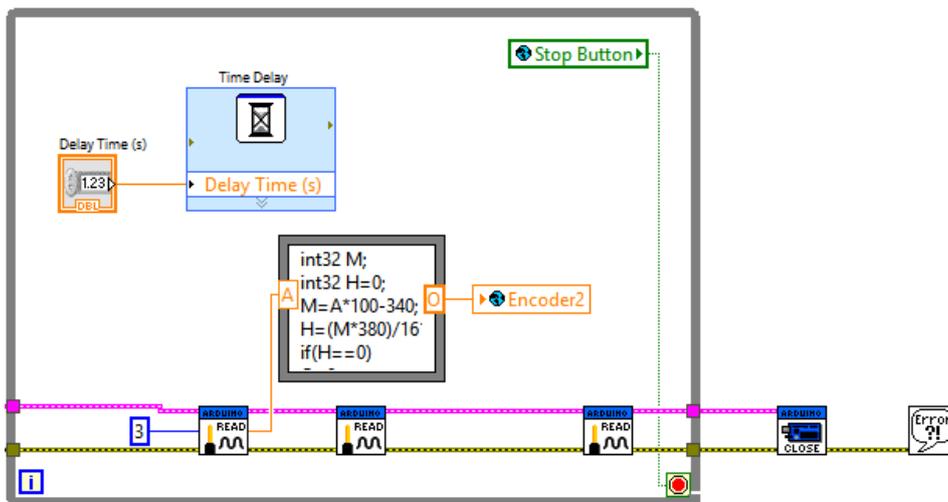


Ilustración 131 Bucle de adquisición de señal por el puerto analógico

Una vez leído el puerto y pulsado el botón, el sistema informa de la posición real que se encuentra el robot, lo devuelve a “Home” calculando los pasos necesarios para llevarlo a dicha posición. Por último, se requiere pulsar “Reset” para reestablecer los valores de posición.

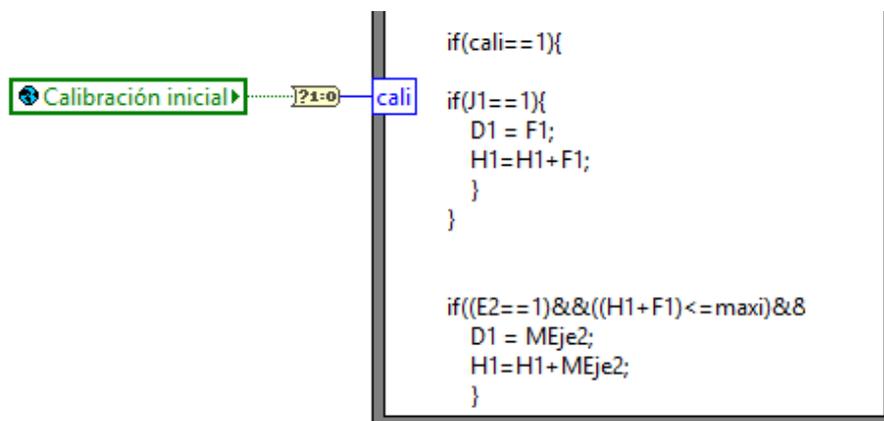


Ilustración 132 Calibración inicial, sección de código dentro del Script de movimiento de los motores

## 6.2.5 Variables de sistema

LabVIEW ofrece distintas formas de gestionar variables, principalmente las más usadas son dos:

- Variables Locales, caracterizadas por el símbolo de una casa en su declaración. Se pueden usar dentro de un mismo sub-VI. Facilitan y mejoran las conexiones con los diferentes bloques. Ayudan a evitar problemas de repetición de una misma variable, es decir se pueden tener variables con el mismo nombre en diferentes sub-VI.
- Variables Globales, pueden ser usadas en cualquier momento y lugar. No puede repetirse el nombre de la variable.



Ilustración 133 Izquierda: variables locales; Derecha: variables globales

## 6.2.6 Posición Home

Esta acción se enmarca dentro del bloque de motores, pero se precisaba una explicación previa de los apartados anteriores para comprender el funcionamiento. Se encarga de analizar la posición actual y calcular el recorrido y la cantidad de movimiento para regresar a la posición de “Home”. En caso de ser accionado introduce los pasos necesarios al sistema para que el robot vuelva a la posición de “Home”

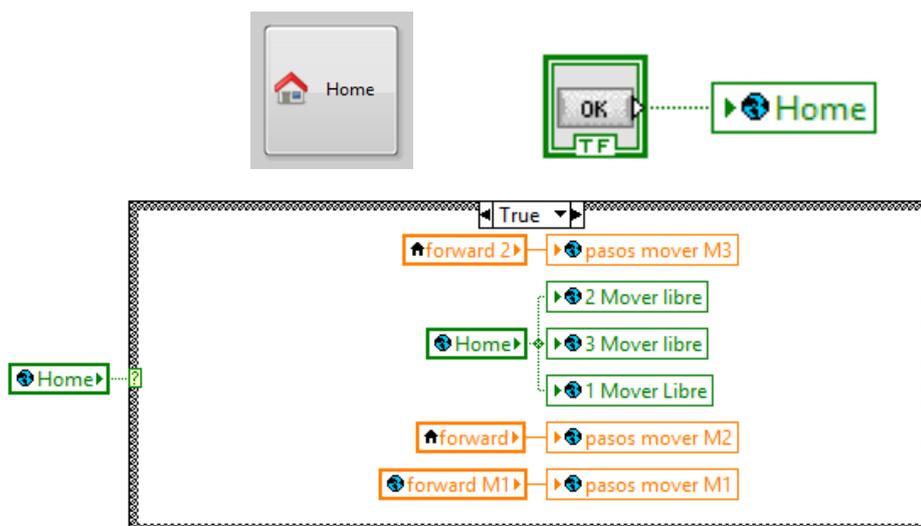


Ilustración 134 Botón de home, Variable global y Bucle de pasos para revertir a la posición de Home

## Capítulo 7. Pruebas y resultado final

Durante el proceso de ejecución del proyecto se han realizado pruebas y prototipos. En el aspecto mecánico se han descartado numerosas piezas impresas debido a fallos en tolerancias.

El apartado que ha supuesto una mayor prueba empírica ha sido la programación de software debido a la escasa información que existe del uso del Toolkit de Arduino. Cada uno de los módulos de programación se ha probado de manera independiente y posteriormente se han ido complementando para comprobar que no existan conflictos de funcionamiento.

Aspecto a destacar, se encuentra en la detección de los sensores de posición que tras fijarse a las tres coronas se ha tenido que realizar un ajuste proporcional para obtener la posición. Este ajuste se ha realizado dentro del ‘script’ del bucle de adquisición de señal de los sensores descrito anteriormente.

Moviendo el robot manualmente y con la información que se obtiene en la interfaz de usuario se ha determinado la posición extrema de cada una de las barras del robot.

Tras conectar tanto la alimentación como la conexión USB e iniciar el programa numerosas veces, se ha determinado el correcto funcionamiento de la comunicación entre el PC y el robot. A tener en cuenta que ciertas versiones de Arduino presentan el chip USB 2n2645H que presenta incompatibilidades con ciertos puertos USB.

Se han realizado movimientos, lentos y rápidos, se ha manejado el robot en todas sus posiciones incluso forzando a sus límites y se ha determinado un correcto funcionamiento. Hay tener en cuenta que cuando la carga del robot supera el par motor especificado, los motores de los ejes pueden sufrir pérdida de pasos. Respetar la carga máxima.

## Capítulo 8. Mejoras en el diseño

Al tratarse del primer prototipo del robot “Garra” existen numerosos aspectos que tras la realización del prototipo pueden ser mejorados.

La distribución de cables puede mejorar el aspecto visual del prototipo obteniendo las longitudes de cableado necesarias para que no sobre material ni queden elementos colgantes.

Tras las pruebas, los potenciómetros han dado malos resultados principalmente por el ruido eléctrico, además se suma que el pequeño recorrido que disponen propicia una baja resolución.

Se propone añadir un sistema de fijación al robot para la unidad de control.

Se recomienda añadir una parada de emergencia externa en caso de malfuncionamiento o posible daño a estructuras o al modelo en sí mismo.

## Capítulo 9. Conclusiones

El objetivo del presente proyecto es el diseño, construcción y programación de un robot paletizador a escala para el uso docente.

Primeramente, se ha comenzado por la búsqueda de robots industriales disponibles en el mercado para elegir la opción más acertada.

Una vez elegido el robot paletizador comienza la fase de diseño en la que se procede a la búsqueda de actuadores, sensores, unidad de control y mecanismos que se adapten a las especificaciones de diseño. Posteriormente con la ayuda de planos CAD de versiones comerciales, se inicia el diseño 2D y 3D del modelo.

Una vez se ha finalizado el diseño del robot, se procede a la materialización de las piezas mediante impresión 3D.

Una vez impresas cada una de las piezas, se ha realizado el ensamblaje completo del modelo incluyendo mecanismos, actuadores, sensores y el controlador.

Terminada toda la parte de diseño físico del robot, se inicia el diseño de la interfaz de usuario mediante LabVIEW. La interfaz compuesta de diferentes módulos presenta una apariencia sencilla e intuitiva. Se trata de una interfaz capaz de controlar cada uno de los motores, monitorizar la posición de las articulaciones y además permite realizar calibrados.

Los principales problemas que se han presentado han sido relacionados con las tolerancias con las piezas impresas debido a la expansión horizontal que se produce en los materiales termoplásticos de impresión 3D. Con ajustes en los parámetros del software de impresión se han solucionado dichos inconvenientes.

Problemas secundarios que se han producido han sido por la falta de información acerca del Toolkit de Arduino, concretamente en los bloques de movimiento de los motores paso a paso. Es por lo que la fase de pruebas que más tiempo ha ocupado ha sido la de Software. Durante la realización de las pruebas se determinó que existían incompatibilidades con ciertos puertos USB debido al chip 2n2645H que incorporan algunas versiones de Arduino.

Englobando el proyecto total se puede considerar que se trata de un proyecto completo, es decir agrupa diferentes aptitudes y disciplinas como son el modelado CAD 2D y 3D, conocimientos en sistemas mecánicos (cálculo de engranajes, rodamientos tornillería, sistemas de barras), conocimientos en electrónica (sensores, actuadores, drivers de motores), adquisición de datos, microcontroladores y conocimientos en programación para el software de Arduino y LabVIEW.

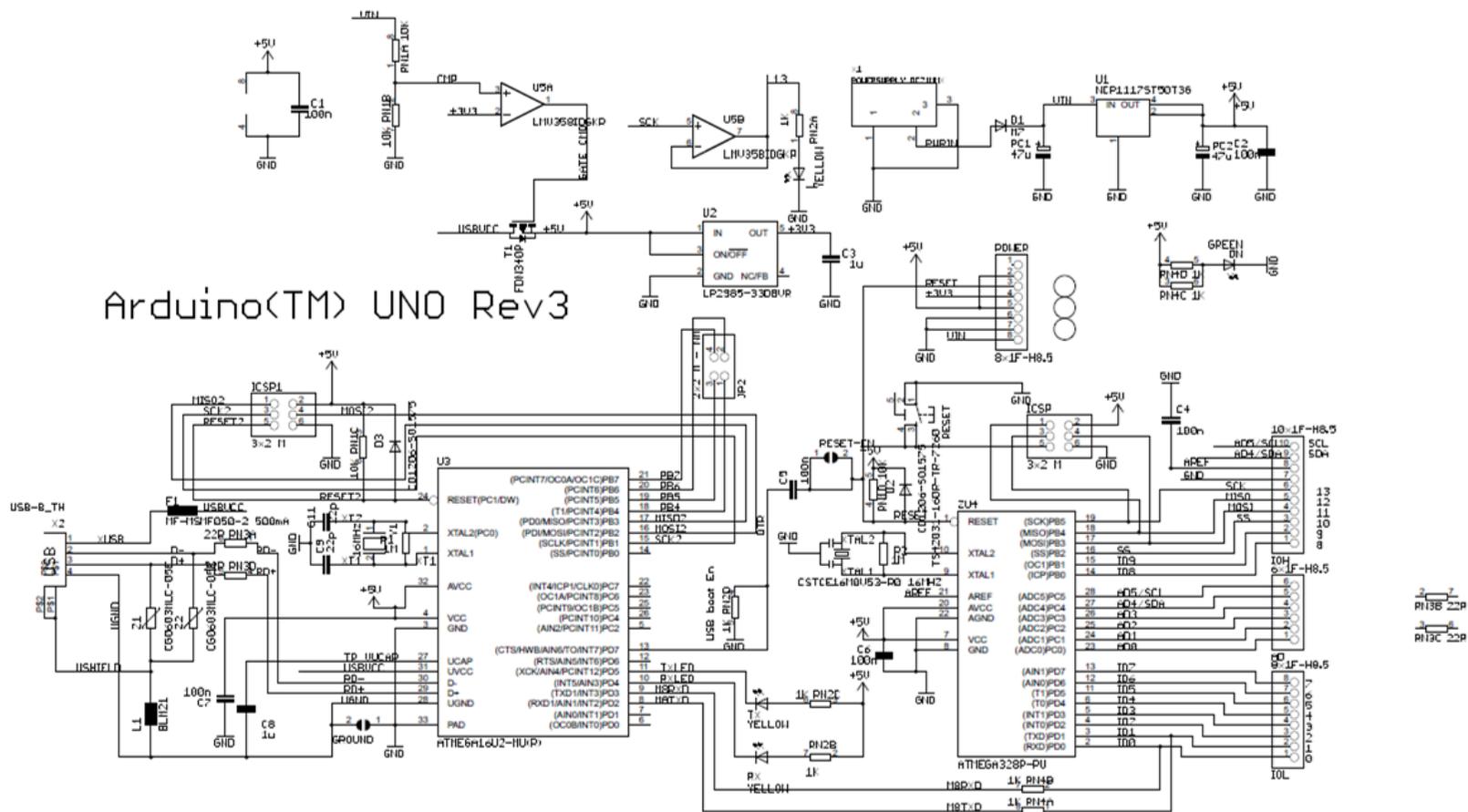
Para concluir, se ha diseñado un sistema robot que cumple todas las características de diseño establecidas, completamente funcional y preparado para la utilización en el ámbito de la docencia debido a su facilidad de uso y comunicación mediante software.

## Capítulo 9. Bibliografía

- [1] “Sitio web *POLOLU*” <https://www.pololu.com/product/1182>
- [2] “Sitio web *POLOLU*” <https://www.pololu.com/product/2132>
- [3] “Sitio web *CNC SHIELD*” <https://blog.protoneer.co.nz/arduino-cnc-shield/>
- [4] “Sitio web *ABB, robots industriales comerciales*” <https://new.abb.com/es>
- [5] “Sitio web *KUKA, robots industriales comerciales*” <https://www.kuka.com/es-es>
- [6] “Sitio web *FANUC, robots industriales comerciales*” <https://www.fanuc.com/>
- [7] “Sitio web *ARDUINO, comunidad DIY*” <https://www.arduino.cc/>
- [8] “Sitio web *LabVIEW*” <http://www.ni.com/es-es/shop/labview.html>
- [9] Universidad Politécnica de Valencia, departamento de sistemas electrónicos, ‘Control de motores paso a paso mediante microcontroladores’ <http://server-die.alc.upv.es/asignaturas/lсед/2002-03/MotoresPasoPaso/Motorespasoapaso.pdf>
- [10] Universidad Pública de Navarra ‘Control de un motor paso a paso: PIC, USB y C#’ <http://academica-e.unavarra.es/bitstream/handle/2454/3547/577435.pdf?sequence=1>
- [11] “Sitio web *MOVER MOTORES PASO A PASO CON ARDUINO*” <http://diymakers.es/mover-motores-paso-paso-con-arduino/>
- [12] “Sitio web *¿Qué es una Impresora 3D?*” <http://www.areatecnologia.com/informatica/impresoras-3d.html>
- [13] “Sitio web *Puertos Analógicos Arduino Avanzado*” <https://aprendiendoarduino.wordpress.com/tag/adc/>
- [14] Universidad Politécnica de Valencia, Departamento de ingeniería de Sistemas y Automática, “Cinemática Directa” y “Cinemática Inversa”
- [15] Universidad Politécnica de Valencia, Departamento de ingeniería de Sistemas y Automática, “Control de Fuerza y Velocidad”
- [16] Universidad Politécnica de Valencia, Departamento de ingeniería de Sistemas y Automática, “Sensores, Motores y Transmisiones”
- [17] Ministerio de Industria y Energía, Gobierno de España “Sistemas de automatización y robótica para las pymes españolas”
- [18] Universidad de Alcalá, “Guía docente, Sistemas Robotizados, 2016/2017”
- [19] “Sitio web El Encoder” <http://encoder.com/blog/encoder-basics/que-es-un-encoder/>
- [20] “Sitio web Resolver o Encoder” <http://www.infopl.net/blog4/2010/12/03/sistema-de-realimentacion-de-posicion-i-encoder-o-resolver/>



## **Capítulo 10. Anexos**

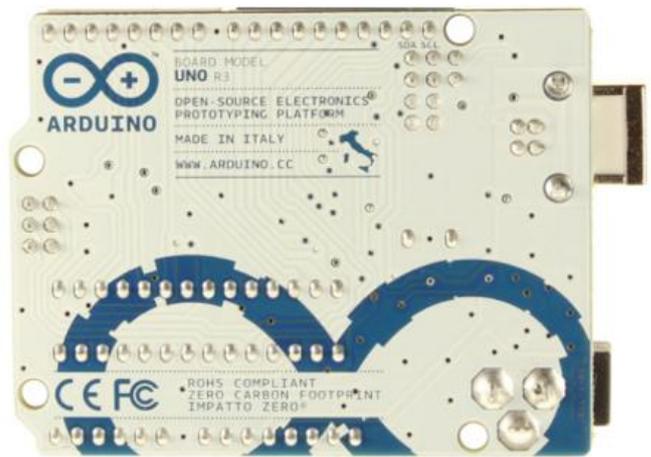


Reference Designs ARE PROVIDED "AS IS" AND "WITH ALL FAULTS. Arduino DISCLAIMS ALL OTHER WARRANTIES, EXPRESS OR IMPLIED, REGARDING PRODUCTS, INCLUDING BUT NOT LIMITED TO, ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Arduino may make changes to specifications and product descriptions at any time, without notice. The Customer must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Arduino reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The product information on the Web Site or Materials is subject to change without notice. Do not finalize a design with this information. ARDUINO is a registered trademark. Use of the ARDUINO name must be compliant with <http://www.arduino.cc/en/Main/Policy>

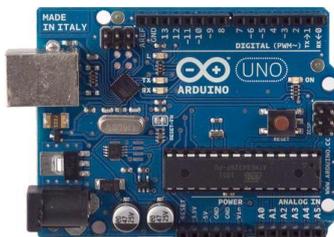
# Arduino Uno



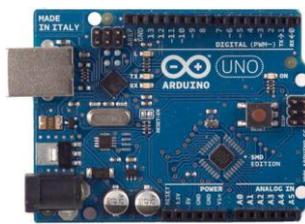
Arduino Uno R3 Front



Arduino Uno R3 Back



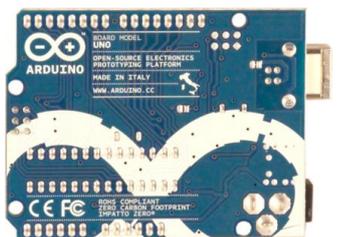
Arduino Uno R2 Front



Arduino Uno SMD



Arduino Uno Front



Arduino Uno Back

## Overview

The Arduino Uno is a microcontroller board based on the ATmega328 ([datasheet](#)). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.

The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter.

[Revision 2](#) of the Uno board has a resistor pulling the 8U2 HWB line to ground, making it easier to put into [DFU mode](#).

[Revision 3](#) of the board has the following new features:

- 1.0 pinout: added SDA and SCL pins that are near to the AREF pin and two other new pins placed near to the RESET pin, the IOREF that allow the shields to adapt to the voltage provided from the board. In future, shields will be compatible both with the board that use the AVR, which operate with 5V and with the Arduino Due that operate with 3.3V. The second one is a not connected pin, that is reserved for future purposes.
- Stronger RESET circuit.
- Atmega 16U2 replace the 8U2.

"Uno" means one in Italian and is named to mark the upcoming release of Arduino 1.0. The Uno and version 1.0 will be the reference versions of Arduino, moving forward. The Uno is the latest in a series of USB Arduino boards, and the reference model for the Arduino platform; for a comparison with previous versions, see the [index of Arduino boards](#).

## Summary

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (recommended)	7-12V

Input Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Clock Speed	16 MHz

## Schematic & Reference Design

EAGLE files: [arduino-uno-Rev3-reference-design.zip](#) (NOTE: works with Eagle 6.0 and newer)

Schematic: [arduino-uno-Rev3-schematic.pdf](#)

**Note:** The Arduino reference design can use an Atmega8, 168, or 328, Current models use an ATmega328, but an Atmega8 is shown in the schematic for reference. The pin configuration is identical on all three processors.

## Power

The Arduino Uno can be powered via the USB connection or with an external power supply. The power source is selected automatically.

External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The power pins are as follows:

- **VIN.** The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- **5V.** This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 12V), the USB connector (5V), or the VIN pin of the board (7-12V). Supplying voltage via the 5V or 3.3V pins bypasses the regulator, and can damage your board. We don't advise it.
- **3V3.** A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- **GND.** Ground pins.

## Memory

The ATmega328 has 32 KB (with 0.5 KB used for the bootloader). It also has 2 KB of SRAM and 1 KB of EEPROM (which can be read and written with the [EEPROM library](#)).

## Input and Output

Each of the 14 digital pins on the Uno can be used as an input or output, using [pinMode\(\)](#), [digitalWrite\(\)](#), and [digitalRead\(\)](#) functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- **Serial: 0 (RX) and 1 (TX).** Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.
- **External Interrupts: 2 and 3.** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the [attachInterrupt\(\)](#) function for details.
- **PWM: 3, 5, 6, 9, 10, and 11.** Provide 8-bit PWM output with the [analogWrite\(\)](#) function.

- **SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK).** These pins support SPI communication using the [SPI library](#).
- **LED: 13.** There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

The Uno has 6 analog inputs, labeled A0 through A5, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though it is possible to change the upper end of their range using the AREF pin and the [analogReference\(\)](#) function. Additionally, some pins have specialized functionality:

- **TWI: A4 or SDA pin and A5 or SCL pin.** Support TWI communication using the [Wire library](#).

There are a couple of other pins on the board:

- **AREF.** Reference voltage for the analog inputs. Used with [analogReference\(\)](#).
- **Reset.** Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

See also the [mapping between Arduino pins and ATmega328 ports](#). The mapping for the Atmega8, 168, and 328 is identical.

## Communication

The Arduino Uno has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega328 provides UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An ATmega16U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The '16U2 firmware uses the standard USB COM drivers, and no external driver is needed. However, [on Windows, a .inf file is required](#). The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (but not for serial communication on pins 0 and 1).

A [SoftwareSerial library](#) allows for serial communication on any of the Uno's digital pins.

The ATmega328 also supports I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus; see the [documentation](#) for details. For SPI communication, use the [SPI library](#).

## Programming

The Arduino Uno can be programmed with the Arduino software ([download](#)). Select "Arduino Uno" from the **Tools > Board** menu (according to the microcontroller on your board). For details, see the [reference](#) and [tutorials](#).

The ATmega328 on the Arduino Uno comes preburned with a [bootloader](#) that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol ([reference](#), [C header files](#)).

You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header; see [these instructions](#) for details.

The ATmega16U2 (or 8U2 in the rev1 and rev2 boards) firmware source code is available. The ATmega16U2/8U2 is loaded with a DFU bootloader, which can be activated by:

- On Rev1 boards: connecting the solder jumper on the back of the board (near the map of Italy) and then resetting the 8U2.
- On Rev2 or later boards: there is a resistor that pulling the 8U2/16U2 HWB line to ground, making it easier to put into DFU mode.

You can then use [Atmel's FLIP software](#) (Windows) or the [DFU programmer](#) (Mac OS X and Linux) to load a new firmware. Or you can use the ISP header with an external programmer (overwriting the DFU bootloader). See [this user-contributed tutorial](#) for more information.

## Automatic (Software) Reset

Rather than requiring a physical press of the reset button before an upload, the Arduino Uno is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the ATmega8U2/16U2 is connected to the reset line of the ATmega328 via a 100 nanofarad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip. The Arduino software uses this capability to allow you to upload code by simply pressing the upload button in the Arduino environment. This means that the bootloader can have a shorter timeout, as the lowering of DTR can be well-coordinated with the start of the upload. This setup has other implications. When the Uno is connected to either a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via USB). For the following half-second or so, the bootloader is running on the Uno. While it is programmed to ignore malformed data (i.e. anything besides an upload of new code), it will intercept the first few bytes of data sent to the board after a connection is opened. If a sketch running on the board receives one-time configuration or other data when it first starts, make sure that the software with which it communicates waits a second after opening the connection and before sending this data.

The Uno contains a trace that can be cut to disable the auto-reset. The pads on either side of the trace can be soldered together to re-enable it. It's labeled "RESET-EN". You may also be able to disable the auto-reset by connecting a 110 ohm resistor from 5V to the reset line; see [this forum thread](#) for details.

## **USB Overcurrent Protection**

The Arduino Uno has a resettable polyfuse that protects your computer's USB ports from shorts and overcurrent. Although most computers provide their own internal protection, the fuse provides an extra layer of protection. If more than 500 mA is applied to the USB port, the fuse will automatically break the connection until the short or overload is removed.

## **Physical Characteristics**

The maximum length and width of the Uno PCB are 2.7 and 2.1 inches respectively, with the USB connector and power jack extending beyond the former dimension. Four screw holes allow the board to be attached to a surface or case. Note that the distance between digital pins 7 and 8 is 160 mil (0.16"), not an even multiple of the 100 mil spacing of the other pins.

## DRV8825 Stepper Motor Controller IC

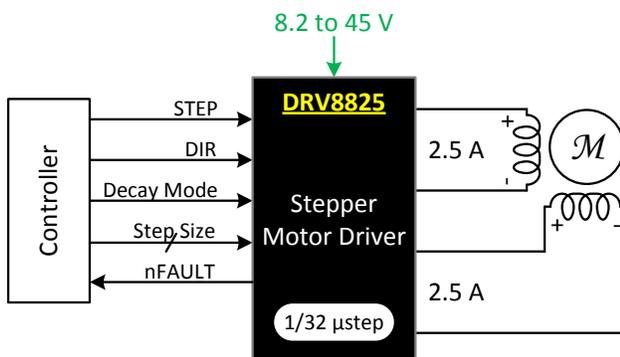
### 1 Features

- PWM Microstepping Stepper Motor Driver
  - Built-In Microstepping Indexer
  - Up to 1/32 Microstepping
- Multiple Decay Modes
  - Mixed Decay
  - Slow Decay
  - Fast Decay
- 8.2-V to 45-V Operating Supply Voltage Range
- 2.5-A Maximum Drive Current at 24 V and  $T_A = 25^\circ\text{C}$
- Simple STEP/DIR Interface
- Low Current Sleep Mode
- Built-In 3.3-V Reference Output
- Small Package and Footprint
- Protection Features
  - Overcurrent Protection (OCP)
  - Thermal Shutdown (TSD)
  - VM Undervoltage Lockout (UVLO)
  - Fault Condition Indication Pin (nFAULT)

### 2 Applications

- Automatic Teller Machines
- Money Handling Machines
- Video Security Cameras
- Printers
- Scanners
- Office Automation Machines
- Gaming Machines
- Factory Automation
- Robotics

### 4 Simplified Schematic



### 3 Description

The DRV8825 provides an integrated motor driver solution for printers, scanners, and other automated equipment applications. The device has two H-bridge drivers and a microstepping indexer, and is intended to drive a bipolar stepper motor. The output driver block consists of N-channel power MOSFET's configured as full H-bridges to drive the motor windings. The DRV8825 is capable of driving up to 2.5 A of current from each output (with proper heat sinking, at 24 V and 25°C).

A simple STEP/DIR interface allows easy interfacing to controller circuits. Mode pins allow for configuration of the motor in full-step up to 1/32-step modes. Decay mode is configurable so that slow decay, fast decay, or mixed decay can be used. A low-power sleep mode is provided which shuts down internal circuitry to achieve very low quiescent current draw. This sleep mode can be set using a dedicated nSLEEP pin.

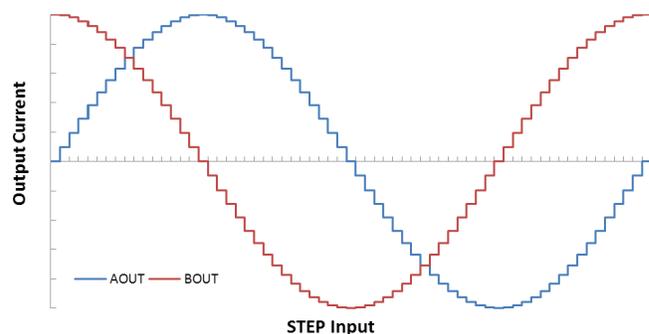
Internal shutdown functions are provided for overcurrent, short circuit, under voltage lockout and over temperature. Fault conditions are indicated via the nFAULT pin.

#### Device Information<sup>(1)</sup>

PART NUMBER	PACKAGE	BODY SIZE (NOM)
DRV8825	HTSSOP (28)	9.70 mm x 6.40 mm

(1) For all available packages, see the orderable addendum at the end of the data sheet.

Microstepping Current Waveform



## Table of Contents

<b>1 Features</b> .....	<b>1</b>	8.3 Feature Description.....	<b>11</b>
<b>2 Applications</b> .....	<b>1</b>	8.4 Device Functional Modes.....	<b>17</b>
<b>3 Description</b> .....	<b>1</b>	<b>9 Application and Implementation</b> .....	<b>18</b>
<b>4 Simplified Schematic</b> .....	<b>1</b>	9.1 Application Information.....	<b>18</b>
<b>5 Revision History</b> .....	<b>2</b>	9.2 Typical Application .....	<b>18</b>
<b>6 Pin Configuration and Functions</b> .....	<b>3</b>	<b>10 Power Supply Recommendations</b> .....	<b>21</b>
<b>7 Specifications</b> .....	<b>4</b>	10.1 Bulk Capacitance .....	<b>21</b>
7.1 Absolute Maximum Ratings .....	<b>4</b>	10.2 Power Supply and Logic Sequencing .....	<b>21</b>
7.2 Handling Ratings.....	<b>4</b>	<b>11 Layout</b> .....	<b>22</b>
7.3 Recommended Operating Conditions.....	<b>4</b>	11.1 Layout Guidelines .....	<b>22</b>
7.4 Thermal Information .....	<b>5</b>	11.2 Layout Example .....	<b>22</b>
7.5 Electrical Characteristics.....	<b>6</b>	11.3 Thermal Protection.....	<b>22</b>
7.6 Timing Requirements.....	<b>7</b>	<b>12 Device and Documentation Support</b> .....	<b>24</b>
7.7 Typical Characteristics .....	<b>8</b>	12.1 Trademarks .....	<b>24</b>
<b>8 Detailed Description</b> .....	<b>9</b>	12.2 Electrostatic Discharge Caution.....	<b>24</b>
8.1 Overview .....	<b>9</b>	12.3 Glossary .....	<b>24</b>
8.2 Functional Block Diagram .....	<b>10</b>	<b>13 Mechanical, Packaging, and Orderable Information</b> .....	<b>24</b>

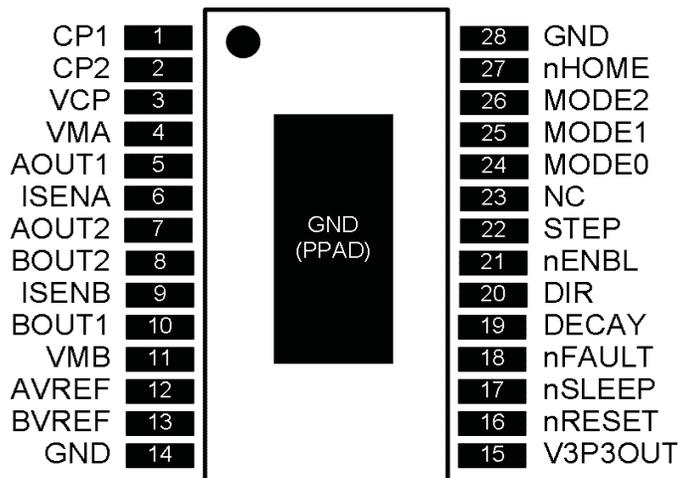
## 5 Revision History

### Changes from Revision E (August 2013) to Revision F

Page

• Added new sections and reordered data sheet to fit new TI flow .....	<b>1</b>
• Updated pin descriptions .....	<b>3</b>
• Added power supply ramp rate and updated ISENSE pin voltage in <i>Absolute Maximum Ratings</i> .....	<b>4</b>
• Updated $V_{IL}$ voltage minimum and typical in <i>Electrical Characteristics</i> .....	<b>6</b>
• Updated $I_{IN}$ and $t_{DEG}$ in <i>Electrical Characteristics</i> .....	<b>6</b>

## 6 Pin Configuration and Functions


**Pin Functions**

PIN		I/O <sup>(1)</sup>	DESCRIPTION	EXTERNAL COMPONENTS OR CONNECTIONS
NAME	NO.			
<b>POWER AND GROUND</b>				
CP1	1	I/O	Charge pump flying capacitor	Connect a 0.01- $\mu$ F 50-V capacitor between CP1 and CP2.
CP2	2	I/O	Charge pump flying capacitor	
GND	14, 28	—	Device ground	
VCP	3	I/O	High-side gate drive voltage	Connect a 0.1- $\mu$ F 16-V ceramic capacitor and a 1-M $\Omega$ resistor to VM.
VMA	4	—	Bridge A power supply	Connect to motor supply (8.2 to 45 V). Both pins must be connected to the same supply, bypassed with a 0.1- $\mu$ F capacitor to GND, and connected to appropriate bulk capacitance.
VMB	11	—	Bridge B power supply	
V3P3OUT	15	O	3.3-V regulator output	Bypass to GND with a 0.47- $\mu$ F 6.3-V ceramic capacitor. Can be used to supply VREF.
<b>CONTROL</b>				
AVREF	12	I	Bridge A current set reference input	Reference voltage for winding current set. Normally AVREF and BVREF are connected to the same voltage. Can be connected to V3P3OUT.
BVREF	13	I	Bridge B current set reference input	
DECAY	19	I	Decay mode	Low = slow decay, open = mixed decay, high = fast decay. Internal pulldown and pullup.
DIR	20	I	Direction input	Level sets the direction of stepping. Internal pulldown.
MODE0	24	I	Microstep mode 0	MODE0 through MODE2 set the step mode - full, 1/2, 1/4, 1/8/1/16, or 1/32 step. Internal pulldown.
MODE1	25	I	Microstep mode 1	
MODE2	26	I	Microstep mode 2	
NC	23	—	No connect	Leave this pin unconnected.
nENBL	21	I	Enable input	Logic high to disable device outputs and indexer operation, logic low to enable. Internal pulldown.
nRESET	16	I	Reset input	Active-low reset input initializes the indexer logic and disables the H-bridge outputs. Internal pulldown.
nSLEEP	17	I	Sleep mode input	Logic high to enable device, logic low to enter low-power sleep mode. Internal pulldown.
STEP	22	I	Step input	Rising edge causes the indexer to move one step. Internal pulldown.
<b>STATUS</b>				
nFAULT	18	OD	Fault	Logic low when in fault condition (overtemp, overcurrent)

(1) Directions: I = input, O = output, OD = open-drain output, IO = input/output

**Pin Functions (continued)**

PIN		I/O <sup>(1)</sup>	DESCRIPTION	EXTERNAL COMPONENTS OR CONNECTIONS
NAME	NO.			
nHOME	27	OD	Home position	Logic low when at home state of step table
<b>OUTPUT</b>				
AOUT1	5	O	Bridge A output 1	Connect to bipolar stepper motor winding A. Positive current is AOUT1 → AOUT2
AOUT2	7	O	Bridge A output 2	
BOUT1	10	O	Bridge B output 1	Connect to bipolar stepper motor winding B. Positive current is BOUT1 → BOUT2
BOUT2	8	O	Bridge B output 2	
ISENA	6	I/O	Bridge A ground / Isense	Connect to current sense resistor for bridge A.
ISENB	9	I/O	Bridge B ground / Isense	Connect to current sense resistor for bridge B.

## 7 Specifications

### 7.1 Absolute Maximum Ratings<sup>(1)(2)</sup>

		MIN	MAX	UNIT
V <sub>(VMx)</sub>	Power supply voltage	-0.3	47	V
	Power supply ramp rate		1	V/μs
	Digital pin voltage	-0.5	7	V
V <sub>(xVREF)</sub>	Input voltage	-0.3	4	V
	ISENSEx pin voltage <sup>(3)</sup>	-0.8	0.8	V
	Peak motor drive output current, t < 1 μs		Internally limited	A
	Continuous motor drive output current <sup>(4)</sup>	0	2.5	A
	Continuous total power dissipation	See <a href="#">Thermal Information</a>		
T <sub>J</sub>	Operating junction temperature range	-40	150	°C

- (1) Stresses beyond those listed under *Absolute Maximum Ratings* 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 under *Recommended Operating Conditions* is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.
- (2) All voltage values are with respect to network ground terminal.
- (3) Transients of ±1 V for less than 25 ns are acceptable
- (4) Power dissipation and thermal limits must be observed.

### 7.2 Handling Ratings

		MIN	MAX	UNIT
T <sub>stg</sub>	Storage temperature range	-60	150	°C
V <sub>(ESD)</sub>	Electrostatic discharge	Human body model (HBM), per ANSI/ESDA/JEDEC JS-001, all pins <sup>(1)</sup>		V
		Charged device model (CDM), per JEDEC specification JESD22-C101, all pins <sup>(2)</sup>		
		-2000	2000	
		-500	500	

- (1) JEDEC document JEP155 states that 500-V HBM allows safe manufacturing with a standard ESD control process.
- (2) JEDEC document JEP157 states that 250-V CDM allows safe manufacturing with a standard ESD control process.

### 7.3 Recommended Operating Conditions

		MIN	NOM	MAX	UNIT
V <sub>(VMx)</sub>	Motor power supply voltage range <sup>(1)</sup>	8.2		45	V
V <sub>(VREF)</sub>	VREF input voltage <sup>(2)</sup>	1		3.5	V
I <sub>V3P3</sub>	V3P3OUT load current	0		1	mA

- (1) All V<sub>M</sub> pins must be connected to the same supply voltage.
- (2) Operational at VREF between 0 to 1 V, but accuracy is degraded.

## 7.4 Thermal Information

THERMAL METRIC <sup>(1)</sup>		DRV8825	UNIT
		PWP	
		28 PINS	
$R_{\theta JA}$	Junction-to-ambient thermal resistance <sup>(2)</sup>	31.6	°C/W
$R_{\theta JC(top)}$	Junction-to-case (top) thermal resistance <sup>(3)</sup>	15.9	
$R_{\theta JB}$	Junction-to-board thermal resistance <sup>(4)</sup>	5.6	
$\psi_{JT}$	Junction-to-top characterization parameter <sup>(5)</sup>	0.2	
$\psi_{JB}$	Junction-to-board characterization parameter <sup>(6)</sup>	5.5	
$R_{\theta JC(bot)}$	Junction-to-case (bottom) thermal resistance <sup>(7)</sup>	1.4	

(1) For more information about traditional and new thermal metrics, see the *IC Package Thermal Metrics* application report, [SPRA953](#).

(2) The junction-to-ambient thermal resistance under natural convection is obtained in a simulation on a JEDEC-standard, high-K board, as specified in JESD51-7, in an environment described in JESD51-2a.

(3) The junction-to-case (top) thermal resistance is obtained by simulating a cold plate test on the package top. No specific JEDEC-standard test exists, but a close description can be found in the ANSI SEMI standard G30-88.

(4) The junction-to-board thermal resistance is obtained by simulating in an environment with a ring cold plate fixture to control the PCB temperature, as described in JESD51-8.

(5) The junction-to-top characterization parameter,  $\psi_{JT}$ , estimates the junction temperature of a device in a real system and is extracted from the simulation data for obtaining  $\theta_{JA}$ , using a procedure described in JESD51-2a (sections 6 and 7).

(6) The junction-to-board characterization parameter,  $\psi_{JB}$ , estimates the junction temperature of a device in a real system and is extracted from the simulation data for obtaining  $\theta_{JA}$ , using a procedure described in JESD51-2a (sections 6 and 7).

(7) The junction-to-case (bottom) thermal resistance is obtained by simulating a cold plate test on the exposed (power) pad. No specific JEDEC standard test exists, but a close description can be found in the ANSI SEMI standard G30-88.

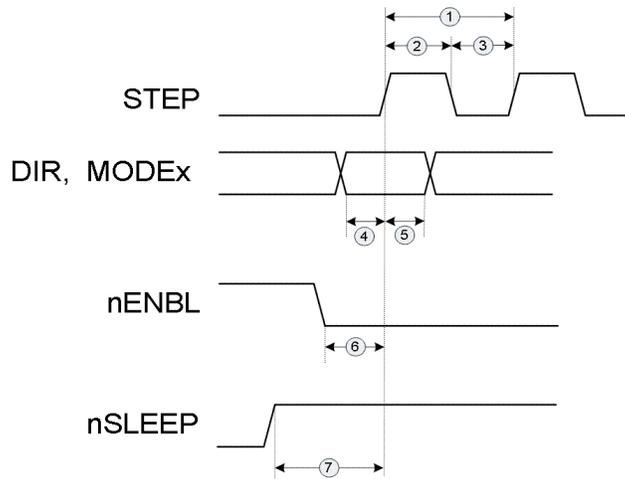
## 7.5 Electrical Characteristics

 over operating free-air temperature range of  $-40^{\circ}\text{C}$  to  $85^{\circ}\text{C}$  (unless otherwise noted)

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
<b>POWER SUPPLIES</b>						
$I_{VM}$	VM operating supply current	$V_{(VMx)} = 24\text{ V}$		5	8	mA
$I_{VMQ}$	VM sleep mode supply current	$V_{(VMx)} = 24\text{ V}$		10	20	$\mu\text{A}$
<b>V3P3OUT REGULATOR</b>						
$V_{3P3}$	V3P3OUT voltage	$\text{IOUT} = 0\text{ to }1\text{ mA}$	3.2	3.3	3.4	V
<b>LOGIC-LEVEL INPUTS</b>						
$V_{IL}$	Input low voltage		0		0.7	V
$V_{IH}$	Input high voltage		2.2		5.25	V
$V_{HYS}$	Input hysteresis		0.3	0.45	0.6	V
$I_{IL}$	Input low current	$V_{IN} = 0$	-20		20	$\mu\text{A}$
$I_{IH}$	Input high current	$V_{IN} = 3.3\text{ V}$			100	$\mu\text{A}$
$R_{PD}$	Internal pulldown resistance			100		k $\Omega$
<b>nHOME, nFAULT OUTPUTS (OPEN-DRAIN OUTPUTS)</b>						
$V_{OL}$	Output low voltage	$I_O = 5\text{ mA}$			0.5	V
$I_{OH}$	Output high leakage current	$V_O = 3.3\text{ V}$			1	$\mu\text{A}$
<b>DECAY INPUT</b>						
$V_{IL}$	Input low threshold voltage	For slow decay mode			0.8	V
$V_{IH}$	Input high threshold voltage	For fast decay mode	2			V
$I_{IN}$	Input current		-40		40	$\mu\text{A}$
$R_{PU}$	Internal pullup resistance (to 3.3 V)			130		k $\Omega$
$R_{PD}$	Internal pulldown resistance			80		k $\Omega$
<b>H-BRIDGE FETS</b>						
$R_{DS(ON)}$	HS FET on resistance	$V_{(VMx)} = 24\text{ V}, I_O = 1\text{ A}, T_J = 25^{\circ}\text{C}$		0.2		$\Omega$
		$V_{(VMx)} = 24\text{ V}, I_O = 1\text{ A}, T_J = 85^{\circ}\text{C}$		0.25	0.32	
	LS FET on resistance	$V_{(VMx)} = 24\text{ V}, I_O = 1\text{ A}, T_J = 25^{\circ}\text{C}$		0.2		
		$V_{(VMx)} = 24\text{ V}, I_O = 1\text{ A}, T_J = 85^{\circ}\text{C}$		0.25	0.32	
$I_{OFF}$	Off-state leakage current		-20		20	$\mu\text{A}$
<b>MOTOR DRIVER</b>						
$f_{PWM}$	Internal current control PWM frequency			30		kHz
$t_{BLANK}$	Current sense blanking time			4		$\mu\text{s}$
$t_R$	Rise time		30		200	ns
$t_F$	Fall time		30		200	ns
<b>PROTECTION CIRCUITS</b>						
$V_{UVLO}$	VM undervoltage lockout voltage	$V_{(VMx)}$ rising		7.8	8.2	V
$I_{OCP}$	Overcurrent protection trip level		3			A
$t_{DEG}$	Overcurrent deglitch time			3		$\mu\text{s}$
$t_{TSD}$	Thermal shutdown temperature	Die temperature	150	160	180	$^{\circ}\text{C}$
<b>CURRENT CONTROL</b>						
$I_{REF}$	xVREF input current	$V_{(xVREF)} = 3.3\text{ V}$	-3		3	$\mu\text{A}$
$V_{TRIP}$	xISENSE trip voltage	$V_{(xVREF)} = 3.3\text{ V}, 100\%$ current setting	635	660	685	mV
$\Delta I_{TRIP}$	Current trip accuracy (relative to programmed value)	$V_{(xVREF)} = 3.3\text{ V}, 5\%$ current setting	-25%		25%	
		$V_{(xVREF)} = 3.3\text{ V}, 10\%$ to 34% current setting	-15%		15%	
		$V_{(xVREF)} = 3.3\text{ V}, 38\%$ to 67% current setting	-10%		10%	
		$V_{(xVREF)} = 3.3\text{ V}, 71\%$ to 100% current setting	-5%		5%	
$A_{ISENSE}$	Current sense amplifier gain	Reference only		5		V/V

## 7.6 Timing Requirements

			MIN	MAX	UNIT
1	$f_{STEP}$	Step frequency		250	kHz
2	$t_{WH(STEP)}$	Pulse duration, STEP high	1.9		$\mu$ s
3	$t_{WL(STEP)}$	Pulse duration, STEP low	1.9		$\mu$ s
4	$t_{SU(STEP)}$	Setup time, command before STEP rising	650		ns
5	$t_{H(STEP)}$	Hold time, command after STEP rising	650		ns
6	$t_{ENBL}$	Enable time, nENBL active to STEP	650		ns
7	$t_{WAKE}$	Wakeup time, nSLEEP inactive high to STEP input accepted		1.7	ms



**Figure 1. Timing Diagram**

### 7.7 Typical Characteristics

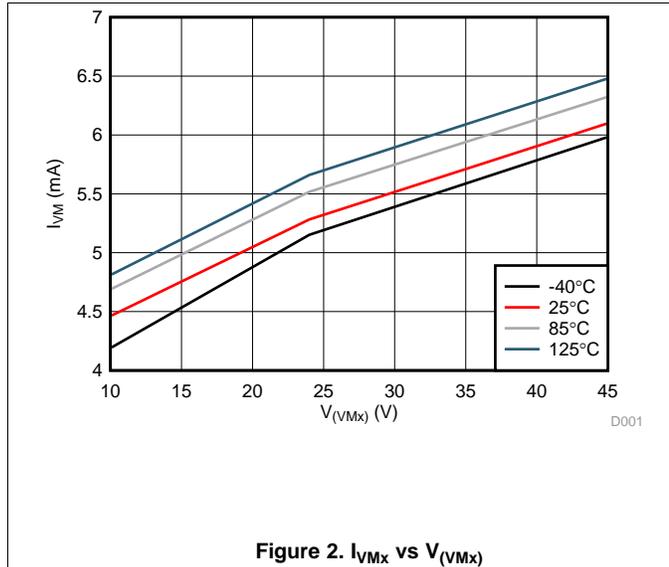


Figure 2.  $I_{VMx}$  vs  $V_{VMx}$

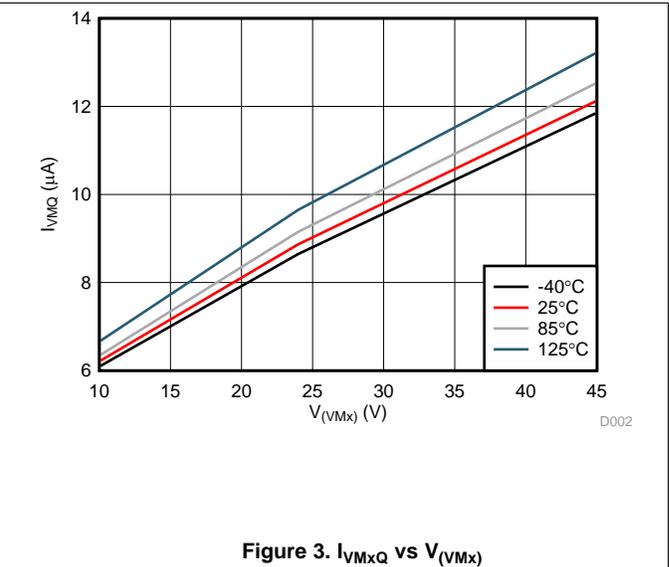


Figure 3.  $I_{VMxQ}$  vs  $V_{VMx}$

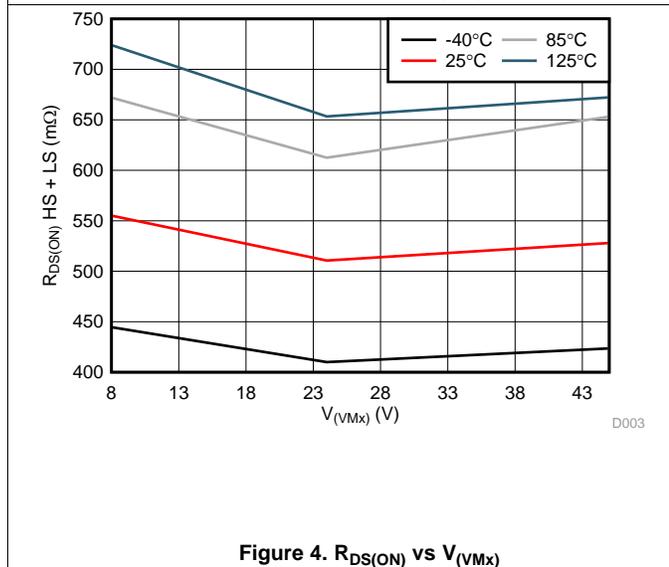


Figure 4.  $R_{DS(ON)}$  vs  $V_{VMx}$

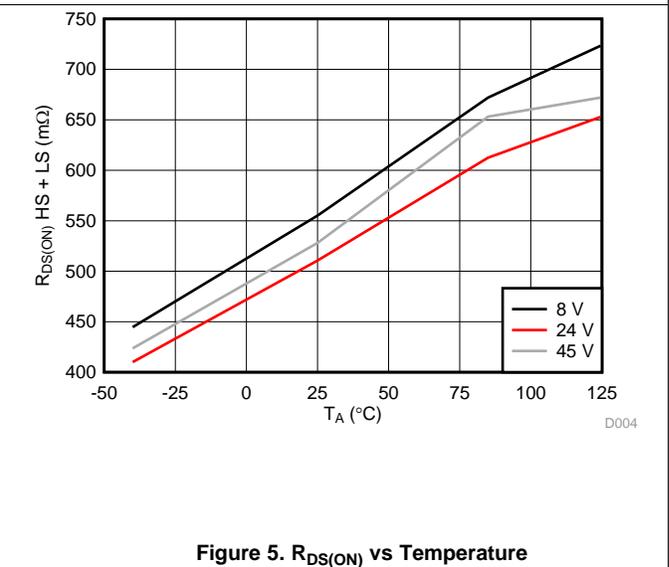


Figure 5.  $R_{DS(ON)}$  vs Temperature

## 8 Detailed Description

### 8.1 Overview

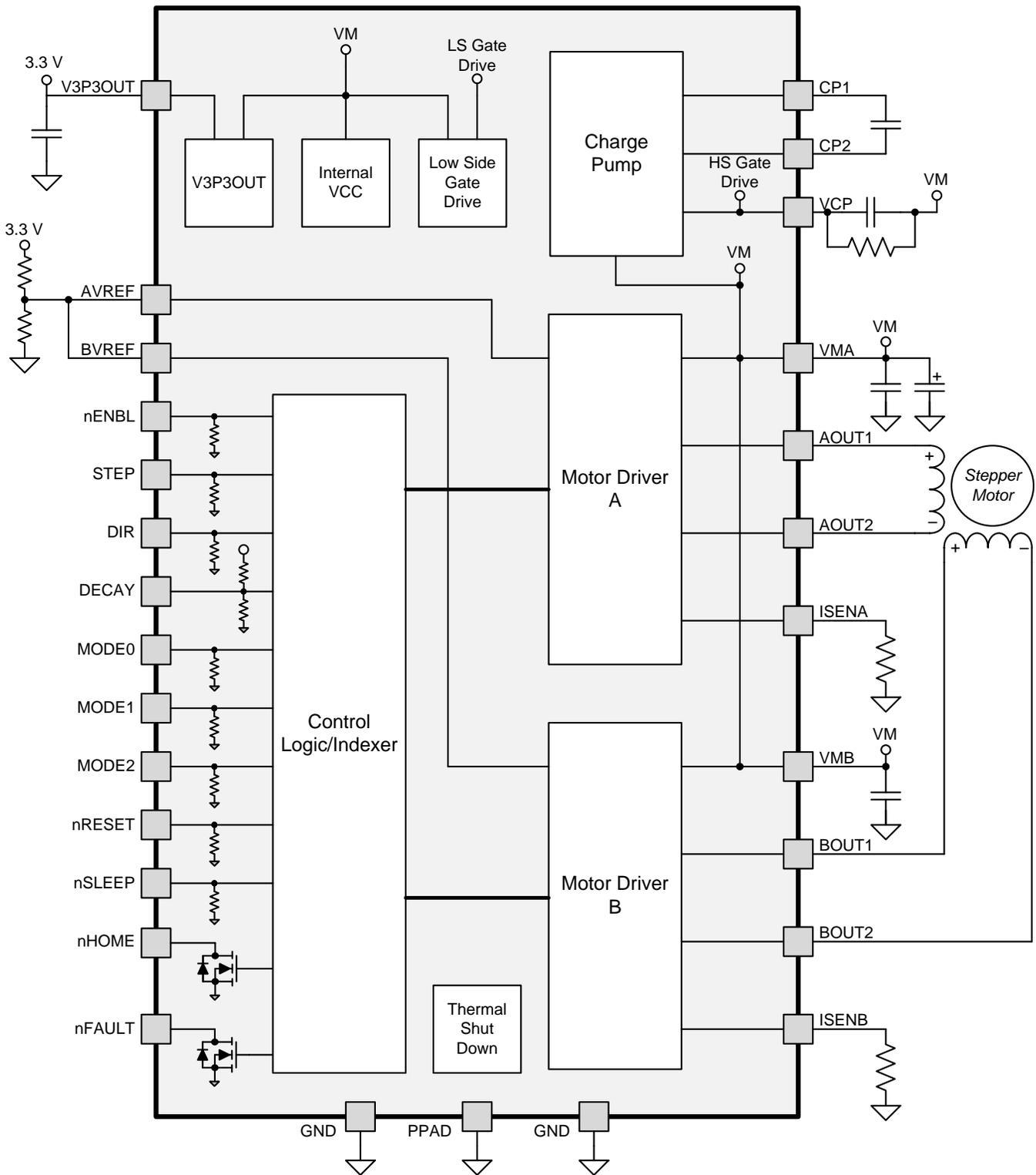
The DRV8825 is an integrated motor driver solution for bipolar stepper motors. The device integrates two NMOS H-bridges, current sense, regulation circuitry, and a microstepping indexer. The DRV8825 can be powered with a supply voltage between 8.2 and 45 V and is capable of providing an output current up to 2.5 A full-scale.

A simple STEP/DIR interface allows for easy interfacing to the controller circuit. The internal indexer is able to execute high-accuracy microstepping without requiring the processor to control the current level.

The current regulation is highly configurable, with three decay modes of operation. Depending on the application requirements, the user can select fast, slow, and mixed decay.

A low-power sleep mode is included which allows the system to save power when not driving the motor.

## 8.2 Functional Block Diagram



### 8.3 Feature Description

#### 8.3.1 PWM Motor Drivers

The DRV8825 contains two H-bridge motor drivers with current-control PWM circuitry. Figure 6 shows a block diagram of the motor control circuitry.

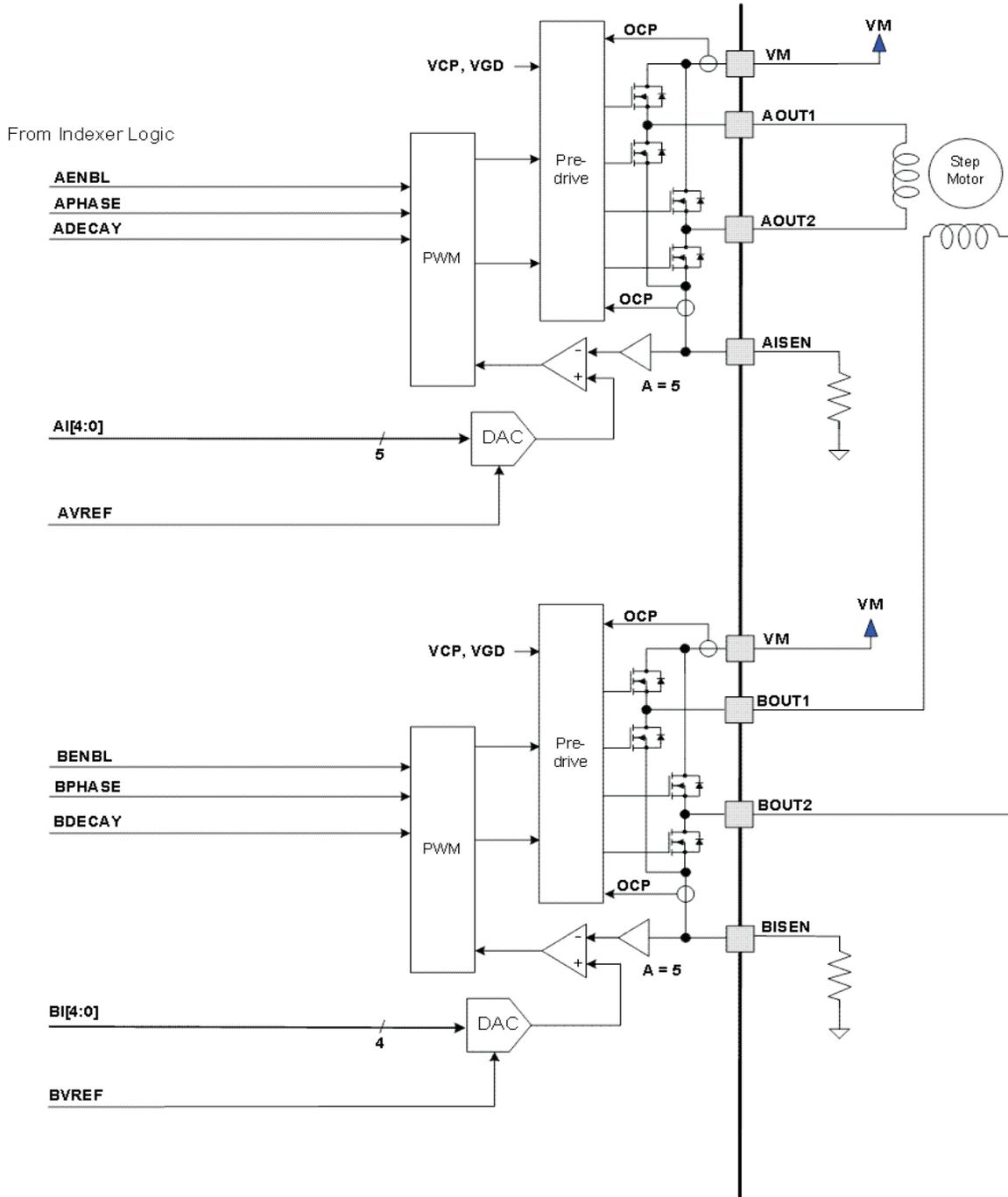


Figure 6. Motor Control Circuitry

Note that there are multiple VM motor power supply pins. All VM pins must be connected together to the motor supply voltage.

## Feature Description (continued)

### 8.3.2 Current Regulation

The current through the motor windings is regulated by a fixed-frequency PWM current regulation, or current chopping. When an H-bridge is enabled, current rises through the winding at a rate dependent on the DC voltage and inductance of the winding. Once the current hits the current chopping threshold, the bridge disables the current until the beginning of the next PWM cycle.

In stepping motors, current regulation is used to vary the current in the two windings in a semi-sinusoidal fashion to provide smooth motion.

The PWM chopping current is set by a comparator which compares the voltage across a current sense resistor connected to the xISEN pins, multiplied by a factor of 5, with a reference voltage. The reference voltage is input from the xVREF pins.

The full-scale (100%) chopping current is calculated in [Equation 1](#).

$$I_{\text{CHOP}} = \frac{V_{(\text{xREF})}}{5 \times R_{\text{ISENSE}}} \quad (1)$$

Example:

If a 0.25-Ω sense resistor is used and the VREFx pin is 2.5 V, the full-scale (100%) chopping current will be  $2.5 \text{ V} / (5 \times 0.25 \text{ } \Omega) = 2 \text{ A}$ .

The reference voltage is scaled by an internal DAC that allows fractional stepping of a bipolar stepper motor, as described in the microstepping indexer section below.

### 8.3.3 Decay Mode

During PWM current chopping, the H-bridge is enabled to drive current through the motor winding until the PWM current chopping threshold is reached. This is shown in [Figure 7](#) as case 1. The current flow direction shown indicates positive current flow.

Once the chopping current threshold is reached, the H-bridge can operate in two different states, fast decay or slow decay.

In fast decay mode, once the PWM chopping current level has been reached, the H-bridge reverses state to allow winding current to flow in a reverse direction. As the winding current approaches 0, the bridge is disabled to prevent any reverse current flow. Fast decay mode is shown in [Figure 7](#) as case 2.

In slow decay mode, winding current is recirculated by enabling both of the low-side FETs in the bridge. This is shown in [Figure 7](#) as case 3.

Feature Description (continued)

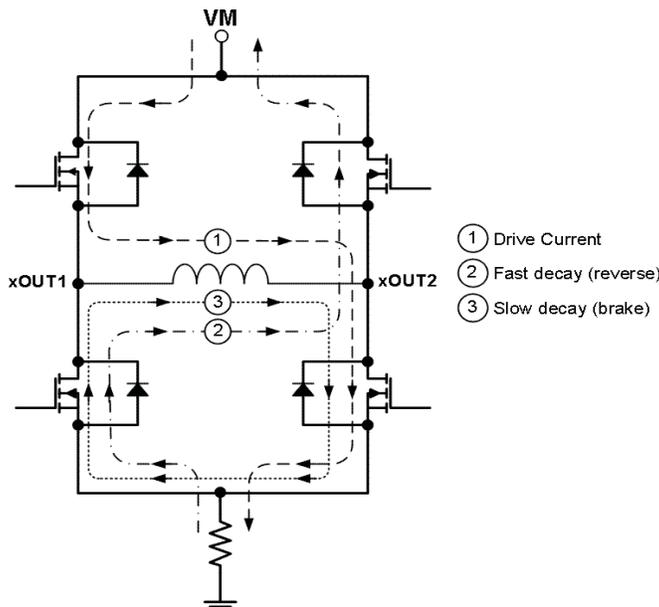


Figure 7. Decay Mode

The DRV8825 supports fast decay, slow decay and a mixed decay mode. Slow, fast, or mixed decay mode is selected by the state of the DECAY pin; logic low selects slow decay, open selects mixed decay operation, and logic high sets fast decay mode. The DECAY pin has both an internal pullup resistor of approximately 130 kΩ and an internal pulldown resistor of approximately 80 kΩ. This sets the mixed decay mode if the pin is left open or undriven.

Mixed decay mode begins as fast decay, but at a fixed period of time (75% of the PWM cycle) switches to slow decay mode for the remainder of the fixed PWM period. This occurs only if the current through the winding is decreasing (per the indexer step table); if the current is increasing, then slow decay is used.

8.3.4 Blanking Time

After the current is enabled in an H-bridge, the voltage on the xISEN pin is ignored for a fixed period of time before enabling the current sense circuitry. This blanking time is fixed at 3.75 μs. Note that the blanking time also sets the minimum on time of the PWM.

8.3.5 Microstepping Indexer

Built-in indexer logic in the DRV8825 allows a number of different stepping configurations. The MODE0 through MODE2 pins are used to configure the stepping format as shown in Table 1.

Table 1. Stepping Format

MODE2	MODE1	MODE0	STEP MODE
0	0	0	Full step (2-phase excitation) with 71% current
0	0	1	1/2 step (1-2 phase excitation)
0	1	0	1/4 step (W1-2 phase excitation)
0	1	1	8 microsteps/step
1	0	0	16 microsteps/step
1	0	1	32 microsteps/step
1	1	0	32 microsteps/step
1	1	1	32 microsteps/step

**Table 2** shows the relative current and step directions for different settings of MODEx. At each rising edge of the STEP input, the indexer travels to the next state in the table. The direction is shown with the DIR pin high; if the DIR pin is low the sequence is reversed. Positive current is defined as xOUT1 = positive with respect to xOUT2.

Note that if the step mode is changed while stepping, the indexer will advance to the next valid state for the new MODEx setting at the rising edge of STEP.

The home state is 45°. This state is entered at power-up or application of nRESET. This is shown in **Table 2** by the shaded cells. The logic inputs DIR, STEP, nRESET, and MODEx have internal pulldown resistors of 100 kΩ.

**Table 2. Relative Current and Step Directions**

1/32 STEP	1/16 STEP	1/8 STEP	1/4 STEP	1/2 STEP	FULL STEP 70%	WINDING CURRENT A	WINDING CURRENT B	ELECTRICAL ANGLE
1	1	1	1	1		100%	0%	0
2						100%	5%	3
3	2					100%	10%	6
4						99%	15%	8
5	3	2				98%	20%	11
6						97%	24%	14
7	4					96%	29%	17
8						94%	34%	20
9	5	3	2			92%	38%	23
10						90%	43%	25
11	6					88%	47%	28
12						86%	51%	31
13	7	4				83%	56%	34
14						80%	60%	37
15	8					77%	63%	39
16						74%	67%	42
17	9	5	3	2	1	71%	71%	45
18						67%	74%	48
19	10					63%	77%	51
20						60%	80%	53
21	11	6				56%	83%	56
22						51%	86%	59
23	12					47%	88%	62
24						43%	90%	65
25	13	7	4			38%	92%	68
26						34%	94%	70
27	14					29%	96%	73
28						24%	97%	76
29	15	8				20%	98%	79
30						15%	99%	82
31	16					10%	100%	84
32						5%	100%	87
33	17	9	5	3		0%	100%	90
34						-5%	100%	93
35	18					-10%	100%	96
36						-15%	99%	98
37	19	10				-20%	98%	101
38						-24%	97%	104
39	20					-29%	96%	107

**Table 2. Relative Current and Step Directions (continued)**

1/32 STEP	1/16 STEP	1/8 STEP	1/4 STEP	1/2 STEP	FULL STEP 70%	WINDING CURRENT A	WINDING CURRENT B	ELECTRICAL ANGLE
40						-34%	94%	110
41	21	11	6			-38%	92%	113
42						-43%	90%	115
43	22					-47%	88%	118
44						-51%	86%	121
45	23	12				-56%	83%	124
46						-60%	80%	127
47	24					-63%	77%	129
48						-67%	74%	132
49	25	13	7	4	2	-71%	71%	135
50						-74%	67%	138
51	26					-77%	63%	141
52						-80%	60%	143
53	27	14				-83%	56%	146
54						-86%	51%	149
55	28					-88%	47%	152
56						-90%	43%	155
57	29	15	8			-92%	38%	158
58						-94%	34%	160
59	30					-96%	29%	163
60						-97%	24%	166
61	31	16				-98%	20%	169
62						-99%	15%	172
63	32					-100%	10%	174
64						-100%	5%	177
65	33	17	9	5		-100%	0%	180
66						-100%	-5%	183
67	34					-100%	-10%	186
68						-99%	-15%	188
69	35	18				-98%	-20%	191
70						-97%	-24%	194
71	36					-96%	-29%	197
72						-94%	-34%	200
73	37	19	10			-92%	-38%	203
74						-90%	-43%	205
75	38					-88%	-47%	208
76						-86%	-51%	211
77	39	20				-83%	-56%	214
78						-80%	-60%	217
79	40					-77%	-63%	219
80						-74%	-67%	222
81	41	21	11	6	3	-71%	-71%	225
82						-67%	-74%	228
83	42					-63%	-77%	231
84						-60%	-80%	233
85	43	22				-56%	-83%	236
86						-51%	-86%	239

**Table 2. Relative Current and Step Directions (continued)**

1/32 STEP	1/16 STEP	1/8 STEP	1/4 STEP	1/2 STEP	FULL STEP 70%	WINDING CURRENT A	WINDING CURRENT B	ELECTRICAL ANGLE
87	44					-47%	-88%	242
88						-43%	-90%	245
89	45	23	12			-38%	-92%	248
90						-34%	-94%	250
91	46					-29%	-96%	253
92						-24%	-97%	256
93	47	24				-20%	-98%	259
94						-15%	-99%	262
95	48					-10%	-100%	264
96						-5%	-100%	267
97	49	25	13	7		0%	-100%	270
98						5%	-100%	273
99	50					10%	-100%	276
100						15%	-99%	278
101	51	26				20%	-98%	281
102						24%	-97%	284
103	52					29%	-96%	287
104						34%	-94%	290
105	53	27	14			38%	-92%	293
106						43%	-90%	295
107	54					47%	-88%	298
108						51%	-86%	301
109	55	28				56%	-83%	304
110						60%	-80%	307
111	56					63%	-77%	309
112						67%	-74%	312
113	57	29	15	8	4	71%	-71%	315
114						74%	-67%	318
115	58					77%	-63%	321
116						80%	-60%	323
117	59	30				83%	-56%	326
118						86%	-51%	329
119	60					88%	-47%	332
120						90%	-43%	335
121	61	31	16			92%	-38%	338
122						94%	-34%	340
123	62					96%	-29%	343
124						97%	-24%	346
125	63	32				98%	-20%	349
126						99%	-15%	352
127	64					100%	-10%	354
128						100%	-5%	357

### 8.3.6 nRESET, nENBL, and nSLEEP Operation

The nRESET pin, when driven active low, resets internal logic, and resets the step table to the home position. It also disables the H-bridge drivers. The STEP input is ignored while nRESET is active.

The nENBL pin is used to control the output drivers and enable/disable operation of the indexer. When nENBL is low, the output H-bridges are enabled, and rising edges on the STEP pin are recognized. When nENBL is high, the H-bridges are disabled, the outputs are in a high-impedance state, and the STEP input is ignored.

Driving nSLEEP low will put the device into a low power sleep state. In this state, the H-bridges are disabled, the gate drive charge pump is stopped, the V3P3OUT regulator is disabled, and all internal clocks are stopped. In this state all inputs are ignored until nSLEEP returns inactive high. When returning from sleep mode, some time (approximately 1 ms) needs to pass before applying a STEP input, to allow the internal circuitry to stabilize. Note that nRESET and nENBL have internal pulldown resistors of approximately 100 k $\Omega$ . The nSLEEP pin has an internal pulldown resistor of 1 M $\Omega$ . nSLEEP and nRESET signals need to be driven to logic high for device operation.

### 8.3.7 Protection Circuits

The DRV8825 is fully protected against undervoltage, overcurrent, and overtemperature events.

#### 8.3.7.1 Overcurrent Protection (OCP)

An analog current limit circuit on each FET limits the current through the FET by removing the gate drive. If this analog current limit persists for longer than the OCP time, all FETs in the H-bridge will be disabled and the nFAULT pin will be driven low. The device remains disabled until either nRESET pin is applied, or VM is removed and reapplied.

Overcurrent conditions on both high-side and low-side devices; that is, a short to ground, supply, or across the motor winding all result in an overcurrent shutdown. Note that overcurrent protection does not use the current sense circuitry used for PWM current control, and is independent of the I<sub>SENSE</sub> resistor value or xVREF voltage.

#### 8.3.7.2 Thermal Shutdown (TSD)

If the die temperature exceeds safe limits, all FETs in the H-bridge will be disabled and the nFAULT pin will be driven low. After the die temperature has fallen to a safe level, operation automatically resumes.

#### 8.3.7.3 Undervoltage Lockout (UVLO)

If at any time the voltage on the VM pins falls below the UVLO threshold voltage, all circuitry in the device will be disabled and internal logic will be reset. Operation will resume when V<sub>(VMx)</sub> rises above the UVLO threshold.

## 8.4 Device Functional Modes

### 8.4.1 STEP/DIR Interface

The STEP/DIR interface provides a simple method for advancing through the indexer table. For each rising edge on the STEP pin, the indexer travels to the next state in the table. The direction it moves in the table is determined by the input to the DIR pin. The signals applied to the STEP and DIR pins should not violate the timing diagram specified in [Figure 1](#).

### 8.4.2 Microstepping

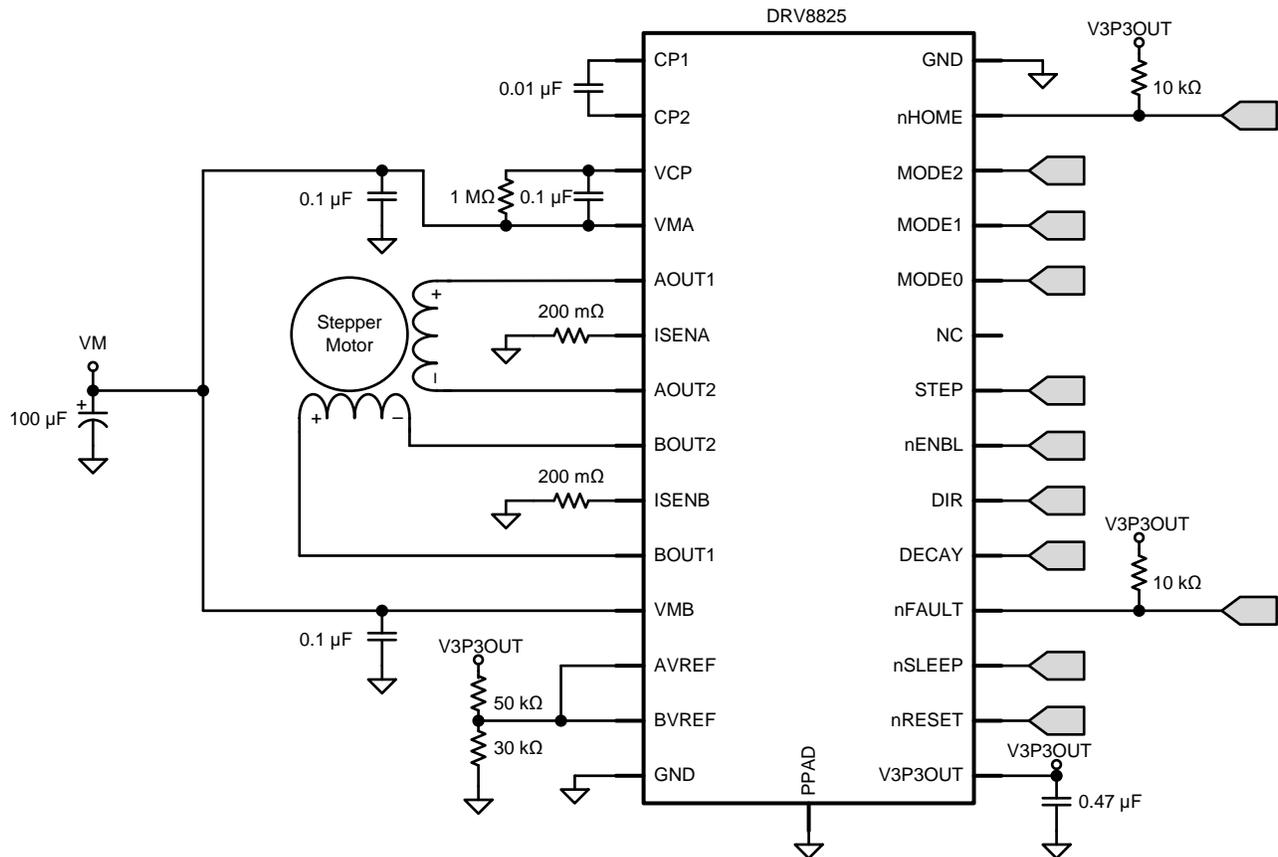
The microstepping indexer allows for a variety of stepping configurations. The state of the indexer is determined by the configuration of the three MODE pins (refer to [Table 1](#) for configuration options). The DRV8825 supports full step up to 1/32 microstepping.

## 9 Application and Implementation

### 9.1 Application Information

The DRV8825 is used in bipolar stepper control. The microstepping motor driver provides additional precision and a smooth rotation from the stepper motor. The following design is a common application of the DRV8825.

### 9.2 Typical Application



#### 9.2.1 Design Requirements

Design Parameter	Reference	Example Value
Supply Voltage	VM	24 V
Motor Winding Resistance	RL	3.9 Ω
Motor Winding Inductance	IL	2.9 mH
Motor Full Step Angle	$\theta_{step}$	1.8°/step
Target Microstepping Level	nm	8 $\mu$ steps per step
Target Motor Speed	v	120 rpm
Target Full-Scale Current	IFS	1.25 A

#### 9.2.2 Detailed Design Procedure

##### 9.2.2.1 Stepper Motor Speed

The first step in configuring the DRV8825 requires the desired motor speed and microstepping level. If the target application requires a constant speed, then a square wave with frequency  $f_{step}$  must be applied to the STEP pin.

If the target motor startup speed is too high, the motor will not spin. Make sure that the motor can support the target speed or implement an acceleration profile to bring the motor up to speed.

For a desired motor speed ( $v$ ), microstepping level ( $n_m$ ), and motor full step angle ( $\theta_{\text{step}}$ ),

$$f_{\text{step}} \left( \frac{\mu\text{steps}}{\text{second}} \right) = \frac{v \left( \frac{\text{rotations}}{\text{minute}} \right) \times 360 \left( \frac{\circ}{\text{rotation}} \right) \times n_m \left( \frac{\mu\text{steps}}{\text{step}} \right)}{60 \left( \frac{\text{seconds}}{\text{minute}} \right) \times \theta_{\text{step}} \left( \frac{\circ}{\text{step}} \right)} \quad (2)$$

$$f_{\text{step}} \left( \frac{\mu\text{steps}}{\text{second}} \right) = \frac{120 \left( \frac{\text{rotations}}{\text{minute}} \right) \times 360 \left( \frac{\circ}{\text{rotation}} \right) \times 8 \left( \frac{\mu\text{steps}}{\text{step}} \right)}{60 \left( \frac{\text{seconds}}{\text{minute}} \right) \times 1.8 \left( \frac{\circ}{\text{step}} \right)} \quad (3)$$

$\theta_{\text{step}}$  can be found in the stepper motor data sheet or written on the motor itself.

For the DRV8825, the microstepping level is set by the MODE pins and can be any of the settings in [Table 1](#). Higher microstepping will mean a smoother motor motion and less audible noise, but will increase switching losses and require a higher  $f_{\text{step}}$  to achieve the same motor speed.

### 9.2.2.2 Current Regulation

In a stepper motor, the set full-scale current ( $I_{\text{FS}}$ ) is the maximum current driven through either winding. This quantity depends on the  $x\text{VREF}$  analog voltage and the sense resistor value ( $R_{\text{SENSE}}$ ). During stepping,  $I_{\text{FS}}$  defines the current chopping threshold ( $I_{\text{TRIP}}$ ) for the maximum current step. The gain of DRV8825 is set for 5 V/V.

$$I_{\text{FS}} \text{ (A)} = \frac{x\text{VREF (V)}}{A_v \times R_{\text{SENSE}} \text{ (}\Omega\text{)}} = \frac{x\text{VREF (V)}}{5 \times R_{\text{SENSE}} \text{ (}\Omega\text{)}} \quad (4)$$

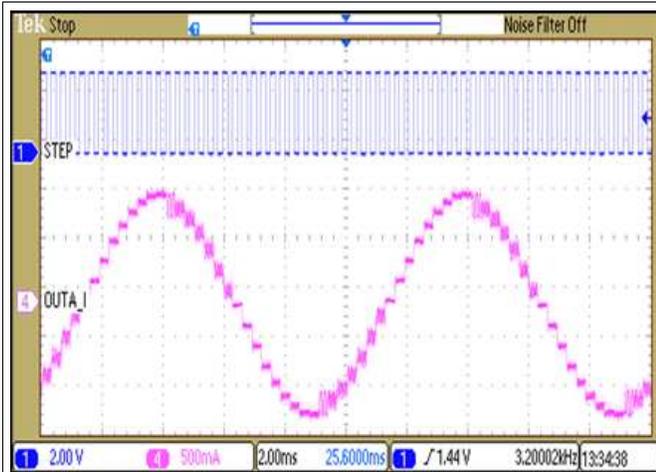
To achieve  $I_{\text{FS}} = 1.25 \text{ A}$  with  $R_{\text{SENSE}}$  of  $0.2 \Omega$ ,  $x\text{VREF}$  should be  $1.25 \text{ V}$ .

### 9.2.2.3 Decay Modes

The DRV8825 supports three different decay modes: slow decay, fast decay, and mixed decay. The current through the motor windings is regulated using a fixed-frequency PWM scheme. This means that after any drive phase, when a motor winding current has hit the current chopping threshold ( $I_{\text{TRIP}}$ ), the DRV8825 will place the winding in one of the three decay modes until the PWM cycle has expired. Afterward, a new drive phase starts.

The blanking time,  $t_{\text{BLANK}}$ , defines the minimum drive time for the current chopping.  $I_{\text{TRIP}}$  is ignored during  $t_{\text{BLANK}}$ , so the winding current may overshoot the trip level.

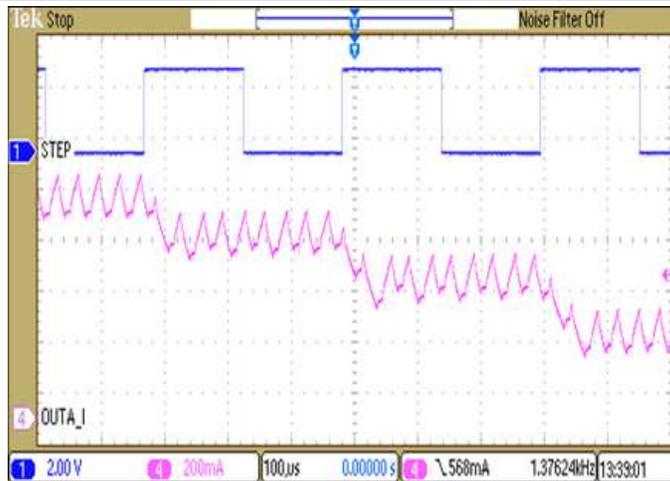
**9.2.3 Application Curves**



**Figure 8. Microstepping Current (Phase A) vs STEP Input, Mixed Decay**



**Figure 9. Microstepping Current (Phase A) vs STEP Input, Slow Decay on Increasing Steps**



**Figure 10. Microstepping Current (Phase A) vs STEP Input, Mixed Decay on Decreasing Steps**

## 10 Power Supply Recommendations

The DRV8825 is designed to operate from an input voltage supply (VMx) range between 8.2 and 45 V. Two 0.1- $\mu$ F ceramic capacitors rated for VMx must be placed as close as possible to the VMA and VMB pins respectively (one on each pin). In addition to the local decoupling caps, additional bulk capacitance is required and must be sized accordingly to the application requirements.

### 10.1 Bulk Capacitance

Bulk capacitance sizing is an important factor in motor drive system design. It is dependent on a variety of factors including:

- Type of power supply
- Acceptable supply voltage ripple
- Parasitic inductance in the power supply wiring
- Type of motor (brushed DC, brushless DC, stepper)
- Motor startup current
- Motor braking method

The inductance between the power supply and motor drive system will limit the rate current can change from the power supply. If the local bulk capacitance is too small, the system will respond to excessive current demands or dumps from the motor with a change in voltage. You should size the bulk capacitance to meet acceptable voltage ripple levels.

The data sheet generally provides a recommended value but system level testing is required to determine the appropriate sized bulk capacitor.

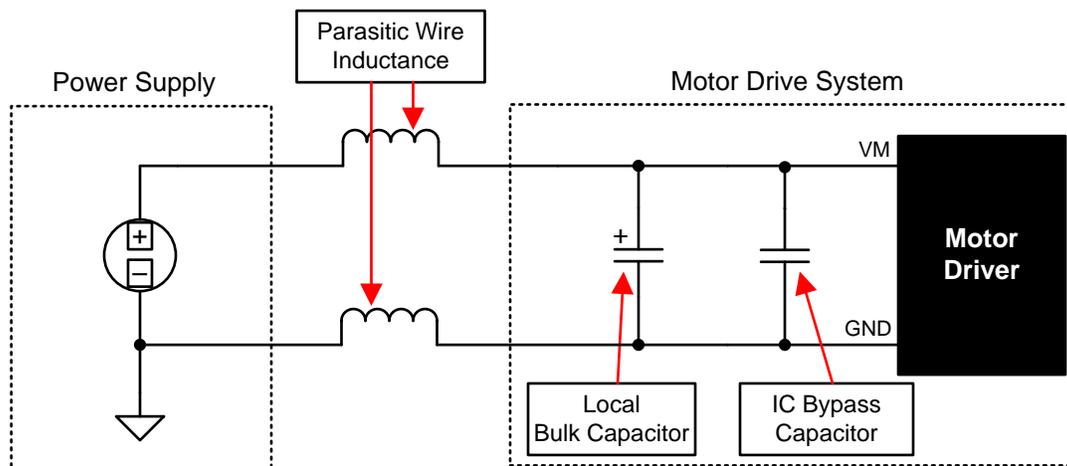


Figure 11. Setup of Motor Drive System With External Power Supply

### 10.2 Power Supply and Logic Sequencing

There is no specific sequence for powering-up the DRV8825. It is okay for digital input signals to be present before VMx is applied. After VMx is applied to the DRV8825, it begins operation based on the status of the control pins.

## 11 Layout

### 11.1 Layout Guidelines

The VMA and VMB pins should be bypassed to GND using low-ESR ceramic bypass capacitors with a recommended value of 0.1- $\mu\text{F}$  rated for VMx. This capacitor should be placed as close to the VMA and VMB pins as possible with a thick trace or ground plane connection to the device GND pin.

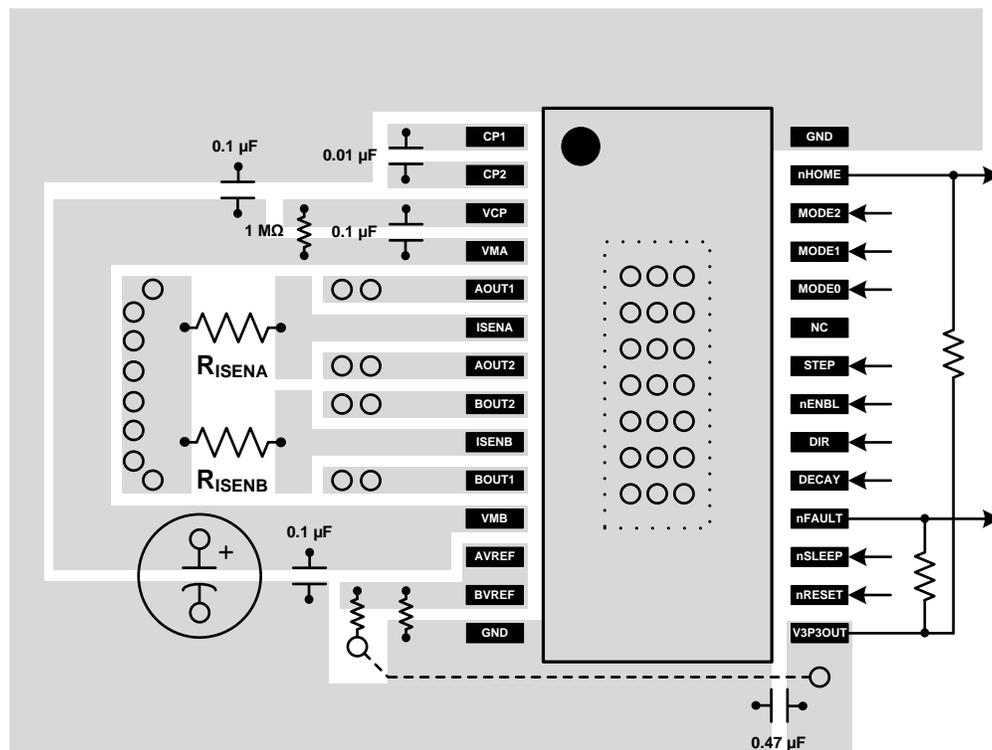
The VMA and VMB pins must be bypassed to ground using an appropriate bulk capacitor. This component may be an electrolytic and should be located close to the DRV8825.

A low-ESR ceramic capacitor must be placed in between the CPL and CPH pins. TI recommends a value of 0.01- $\mu\text{F}$  rated for VMx. Place this component as close to the pins as possible.

A low-ESR ceramic capacitor must be placed in between the VMA and VCP pins. TI recommends a value of 0.1- $\mu\text{F}$  rated for 16 V. Place this component as close to the pins as possible. Also, place a 1-M $\Omega$  resistor between VCP and VMA.

Bypass V3P3 to ground with a ceramic capacitor rated 6.3 V. Place this bypass capacitor as close to the pin as possible

### 11.2 Layout Example



### 11.3 Thermal Protection

The DRV8825 has thermal shutdown (TSD) as described above. If the die temperature exceeds approximately 150°C, the device will be disabled until the temperature drops to a safe level.

Any tendency of the device to enter TSD is an indication of either excessive power dissipation, insufficient heatsinking, or too high an ambient temperature.

#### 11.3.1 Power Dissipation

Power dissipation in the DRV8825 is dominated by the power dissipated in the output FET resistance, or  $R_{DS(ON)}$ . Average power dissipation when running a stepper motor can be roughly estimated by [Equation 5](#).

## Thermal Protection (continued)

$$P_{TOT} = 4 \times R_{DS(ON)} \times (I_{OUT(RMS)})^2 \quad (5)$$

where  $P_{TOT}$  is the total power dissipation,  $R_{DS(ON)}$  is the resistance of each FET, and  $I_{OUT(RMS)}$  is the RMS output current being applied to each winding.  $I_{OUT(RMS)}$  is equal to the approximately 0.7x the full-scale output current setting. The factor of 4 comes from the fact that there are two motor windings, and at any instant two FETs are conducting winding current for each winding (one high-side and one low-side).

The maximum amount of power that can be dissipated in the device is dependent on ambient temperature and heatsinking.

Note that  $R_{DS(ON)}$  increases with temperature, so as the device heats, the power dissipation increases. This must be taken into consideration when sizing the heatsink.

### 11.3.2 Heatsinking

The PowerPAD™ package uses an exposed pad to remove heat from the device. For proper operation, this pad must be thermally connected to copper on the PCB to dissipate heat. On a multi-layer PCB with a ground plane, this can be accomplished by adding a number of vias to connect the thermal pad to the ground plane. On PCBs without internal planes, copper area can be added on either side of the PCB to dissipate heat. If the copper area is on the opposite side of the PCB from the device, thermal vias are used to transfer the heat between top and bottom layers.

For details about how to design the PCB, refer to TI application report [SLMA002](#), "PowerPAD™ Thermally Enhanced Package" and TI application brief [SLMA004](#), *PowerPAD™ Made Easy*, available at [www.ti.com](http://www.ti.com).

In general, the more copper area that can be provided, the more power can be dissipated. It can be seen that the heatsink effectiveness increases rapidly to about 20 cm<sup>2</sup>, then levels off somewhat for larger areas.

## 12 Device and Documentation Support

### 12.1 Trademarks

PowerPAD is a trademark of Texas Instruments.

### 12.2 Electrostatic Discharge Caution



These devices have limited built-in ESD protection. The leads should be shorted together or the device placed in conductive foam during storage or handling to prevent electrostatic damage to the MOS gates.

### 12.3 Glossary

[SLYZ022](#) — *TI Glossary*.

This glossary lists and explains terms, acronyms, and definitions.

## 13 Mechanical, Packaging, and Orderable Information

The following pages include mechanical, packaging, and orderable information. This information is the most current data available for the designated devices. This data is subject to change without notice and revision of this document. For browser-based versions of this data sheet, refer to the left-hand navigation.

**PACKAGING INFORMATION**

Orderable Device	Status (1)	Package Type	Package Drawing	Pins	Package Qty	Eco Plan (2)	Lead/Ball Finish (6)	MSL Peak Temp (3)	Op Temp (°C)	Device Marking (4/5)	Samples
DRV8825PWP	ACTIVE	HTSSOP	PWP	28	50	Green (RoHS & no Sb/Br)	CU NIPDAU	Level-3-260C-168 HR	-40 to 85	DRV8825	<a href="#">Samples</a>
DRV8825PWPR	ACTIVE	HTSSOP	PWP	28	2000	Green (RoHS & no Sb/Br)	CU NIPDAU	Level-3-260C-168 HR	-40 to 85	DRV8825	<a href="#">Samples</a>

(1) The marketing status values are defined as follows:

**ACTIVE:** Product device recommended for new designs.

**LIFEBUY:** TI has announced that the device will be discontinued, and a lifetime-buy period is in effect.

**NRND:** Not recommended for new designs. Device is in production to support existing customers, but TI does not recommend using this part in a new design.

**PREVIEW:** Device has been announced but is not in production. Samples may or may not be available.

**OBsolete:** TI has discontinued the production of the device.

(2) Eco Plan - The planned eco-friendly classification: Pb-Free (RoHS), Pb-Free (RoHS Exempt), or Green (RoHS & no Sb/Br) - please check <http://www.ti.com/productcontent> for the latest availability information and additional product content details.

**TBD:** The Pb-Free/Green conversion plan has not been defined.

**Pb-Free (RoHS):** TI's terms "Lead-Free" or "Pb-Free" mean semiconductor products that are compatible with the current RoHS requirements for all 6 substances, including the requirement that lead not exceed 0.1% by weight in homogeneous materials. Where designed to be soldered at high temperatures, TI Pb-Free products are suitable for use in specified lead-free processes.

**Pb-Free (RoHS Exempt):** This component has a RoHS exemption for either 1) lead-based flip-chip solder bumps used between the die and package, or 2) lead-based die adhesive used between the die and leadframe. The component is otherwise considered Pb-Free (RoHS compatible) as defined above.

**Green (RoHS & no Sb/Br):** TI defines "Green" to mean Pb-Free (RoHS compatible), and free of Bromine (Br) and Antimony (Sb) based flame retardants (Br or Sb do not exceed 0.1% by weight in homogeneous material)

(3) MSL, Peak Temp. - The Moisture Sensitivity Level rating according to the JEDEC industry standard classifications, and peak solder temperature.

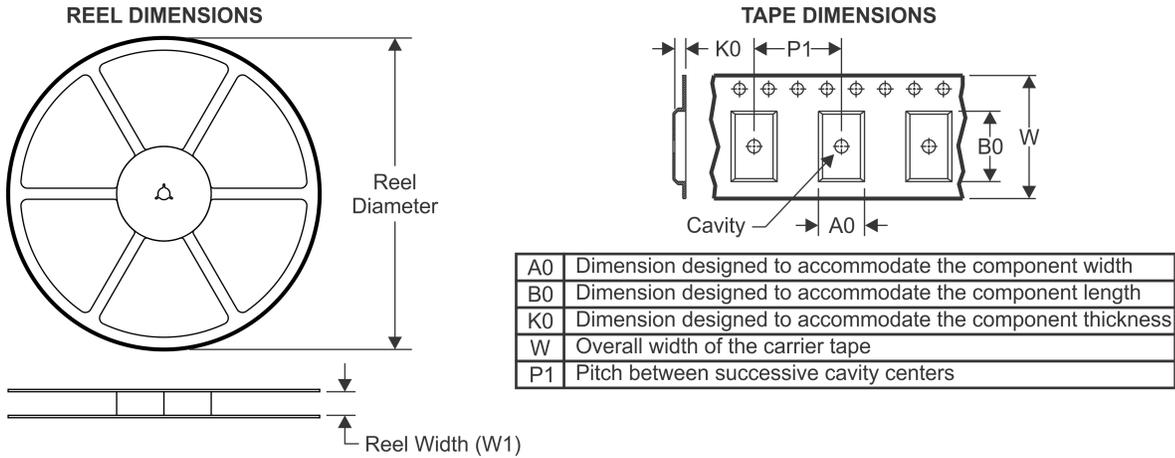
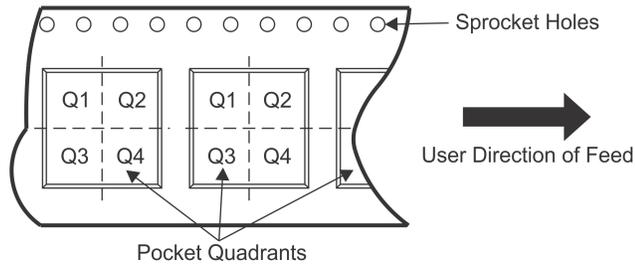
(4) There may be additional marking, which relates to the logo, the lot trace code information, or the environmental category on the device.

(5) Multiple Device Markings will be inside parentheses. Only one Device Marking contained in parentheses and separated by a "~" will appear on a device. If a line is indented then it is a continuation of the previous line and the two combined represent the entire Device Marking for that device.

(6) Lead/Ball Finish - Orderable Devices may have multiple material finish options. Finish options are separated by a vertical ruled line. Lead/Ball Finish values may wrap to two lines if the finish value exceeds the maximum column width.

**Important Information and Disclaimer:** The information provided on this page represents TI's knowledge and belief as of the date that it is provided. TI bases its knowledge and belief on information provided by third parties, and makes no representation or warranty as to the accuracy of such information. Efforts are underway to better integrate information from third parties. TI has taken and continues to take reasonable steps to provide representative and accurate information but may not have conducted destructive testing or chemical analysis on incoming materials and chemicals. TI and TI suppliers consider certain information to be proprietary, and thus CAS numbers and other limited information may not be available for release.

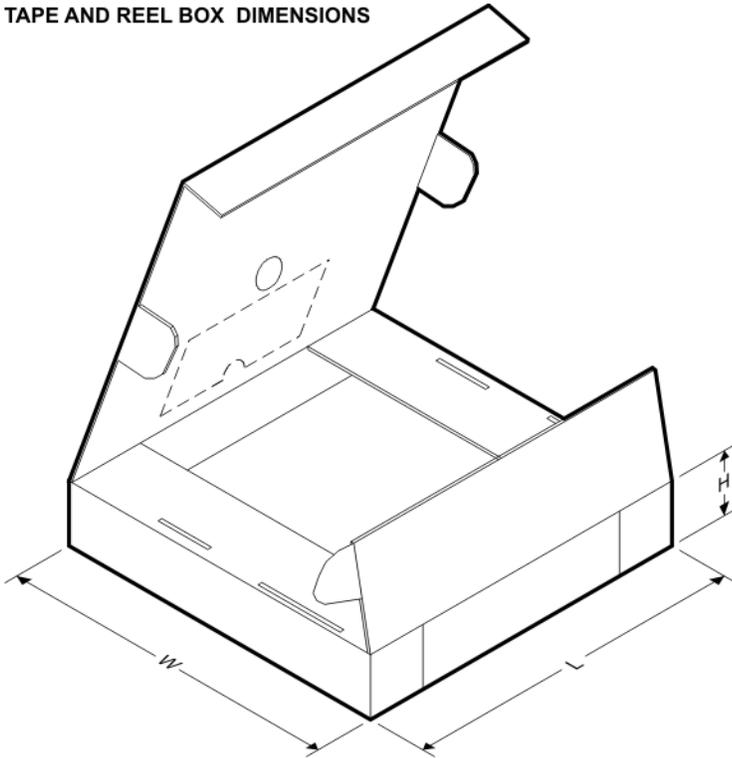
In no event shall TI's liability arising out of such information exceed the total purchase price of the TI part(s) at issue in this document sold by TI to Customer on an annual basis.

**TAPE AND REEL INFORMATION**

**QUADRANT ASSIGNMENTS FOR PIN 1 ORIENTATION IN TAPE**


\*All dimensions are nominal

Device	Package Type	Package Drawing	Pins	SPQ	Reel Diameter (mm)	Reel Width W1 (mm)	A0 (mm)	B0 (mm)	K0 (mm)	P1 (mm)	W (mm)	Pin1 Quadrant
DRV8825PWPR	HTSSOP	PWP	28	2000	330.0	16.4	6.9	10.2	1.8	12.0	16.0	Q1

TAPE AND REEL BOX DIMENSIONS



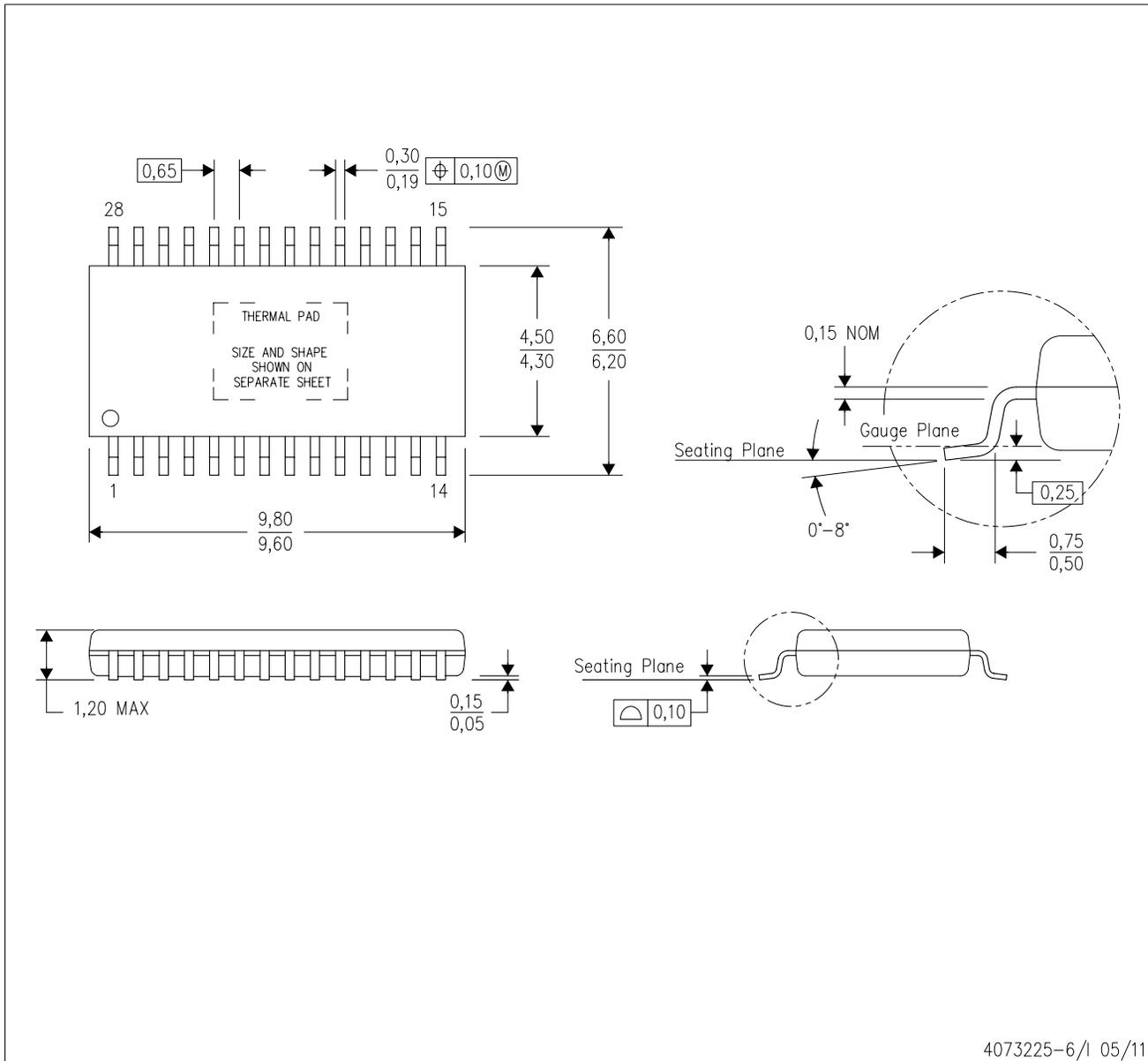
\*All dimensions are nominal

Device	Package Type	Package Drawing	Pins	SPQ	Length (mm)	Width (mm)	Height (mm)
DRV8825PWPR	HTSSOP	PWP	28	2000	367.0	367.0	38.0

# MECHANICAL DATA

PWP (R-PDSO-G28)

PowerPAD™ PLASTIC SMALL OUTLINE



- NOTES:
- All linear dimensions are in millimeters.
  - This drawing is subject to change without notice.
  - Body dimensions do not include mold flash or protrusions. Mold flash and protrusion shall not exceed 0.15 per side.
  - This package is designed to be soldered to a thermal pad on the board. Refer to Technical Brief, PowerPad Thermally Enhanced Package, Texas Instruments Literature No. SLMA002 for information regarding recommended board layout. This document is available at [www.ti.com](http://www.ti.com) <<http://www.ti.com>>.
  - See the additional figure in the Product Data Sheet for details regarding the exposed thermal pad features and dimensions.
  - Falls within JEDEC MO-153

PowerPAD is a trademark of Texas Instruments.

## THERMAL PAD MECHANICAL DATA

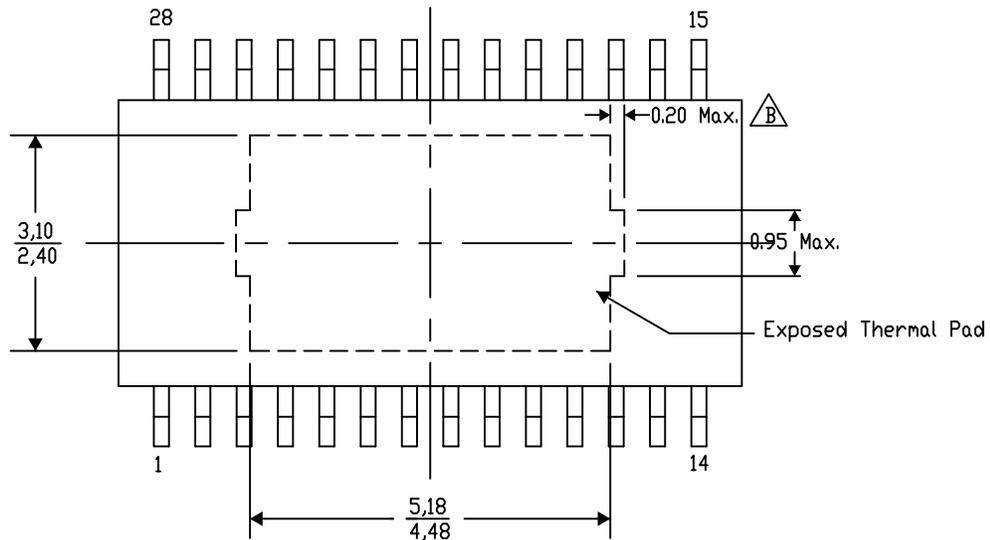
### PWP (R-PDSO-G28) PowerPAD™ SMALL PLASTIC OUTLINE

#### THERMAL INFORMATION

This PowerPAD™ package incorporates an exposed thermal pad that is designed to be attached to a printed circuit board (PCB). The thermal pad must be soldered directly to the PCB. After soldering, the PCB can be used as a heatsink. In addition, through the use of thermal vias, the thermal pad can be attached directly to the appropriate copper plane shown in the electrical schematic for the device, or alternatively, can be attached to a special heatsink structure designed into the PCB. This design optimizes the heat transfer from the integrated circuit (IC).

For additional information on the PowerPAD package and how to take advantage of its heat dissipating abilities, refer to Technical Brief, PowerPAD Thermally Enhanced Package, Texas Instruments Literature No. SLMA002 and Application Brief, PowerPAD Made Easy, Texas Instruments Literature No. SLMA004. Both documents are available at [www.ti.com](http://www.ti.com).

The exposed thermal pad dimensions for this package are shown in the following illustration.



Exposed Thermal Pad Dimensions

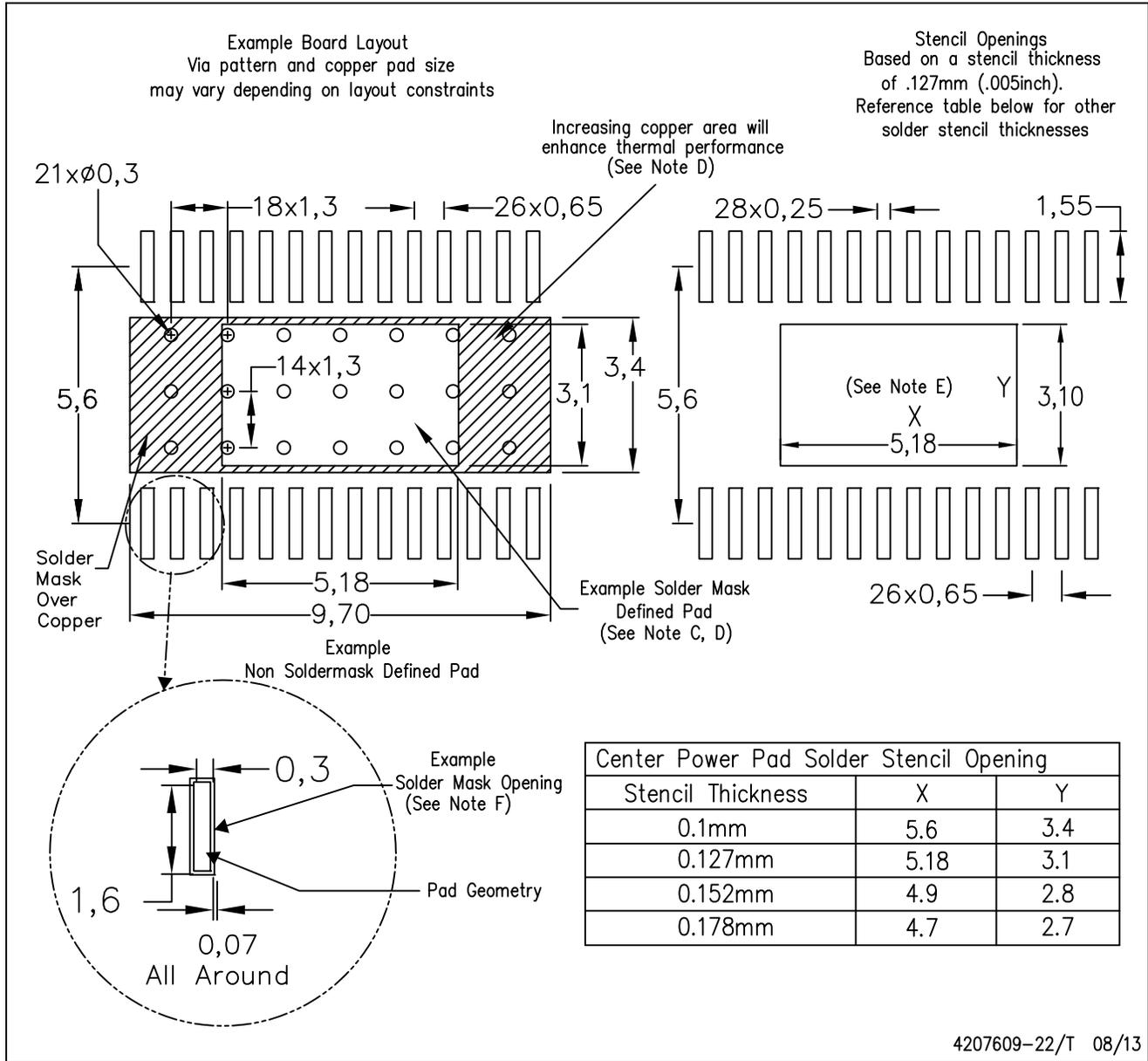
4206332-38/AH 11/13

NOTE: A. All linear dimensions are in millimeters  
B. Exposed tie strap features may not be present.

PowerPAD is a trademark of Texas Instruments

PWP (R-PDSO-G28)

PowerPAD™ PLASTIC SMALL OUTLINE



- NOTES:
- All linear dimensions are in millimeters.
  - This drawing is subject to change without notice.
  - Customers should place a note on the circuit board fabrication drawing not to alter the center solder mask defined pad.
  - This package is designed to be soldered to a thermal pad on the board. Refer to Technical Brief, PowerPad Thermally Enhanced Package, Texas Instruments Literature No. SLMA002, SLMA004, and also the Product Data Sheets.
  - For specific thermal information, via requirements, and recommended board layout. These documents are available at [www.ti.com](http://www.ti.com) <<http://www.ti.com>>. Publication IPC-7351 is recommended for alternate designs. Laser cutting apertures with trapezoidal walls and also rounding corners will offer better paste release. Customers should contact their board assembly site for stencil design recommendations. Example stencil design based on a 50% volumetric metal load solder paste. Refer to IPC-7525 for other stencil
  - Customers should contact their board fabrication site for solder mask tolerances between and around signal pads.

## IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, enhancements, improvements and other changes to its semiconductor products and services per JESD46, latest issue, and to discontinue any product or service per JESD48, latest issue. Buyers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All semiconductor products (also referred to herein as "components") are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its components to the specifications applicable at the time of sale, in accordance with the warranty in TI's terms and conditions of sale of semiconductor products. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by applicable law, testing of all parameters of each component is not necessarily performed.

TI assumes no liability for applications assistance or the design of Buyers' products. Buyers are responsible for their products and applications using TI components. To minimize the risks associated with Buyers' products and applications, Buyers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right relating to any combination, machine, or process in which TI components or services are used. Information published by TI regarding third-party products or services does not constitute a license to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of significant portions of TI information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

Resale of TI components or services with statements different from or beyond the parameters stated by TI for that component or service voids all express and any implied warranties for the associated TI component or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Buyer acknowledges and agrees that it is solely responsible for compliance with all legal, regulatory and safety-related requirements concerning its products, and any use of TI components in its applications, notwithstanding any applications-related information or support that may be provided by TI. Buyer represents and agrees that it has all the necessary expertise to create and implement safeguards which anticipate dangerous consequences of failures, monitor failures and their consequences, lessen the likelihood of failures that might cause harm and take appropriate remedial actions. Buyer will fully indemnify TI and its representatives against any damages arising out of the use of any TI components in safety-critical applications.

In some cases, TI components may be promoted specifically to facilitate safety-related applications. With such components, TI's goal is to help enable customers to design and create their own end-product solutions that meet applicable functional safety standards and requirements. Nonetheless, such components are subject to these terms.

No TI components are authorized for use in FDA Class III (or similar life-critical medical equipment) unless authorized officers of the parties have executed a special agreement specifically governing such use.

Only those TI components which TI has specifically designated as military grade or "enhanced plastic" are designed and intended for use in military/aerospace applications or environments. Buyer acknowledges and agrees that any military or aerospace use of TI components which have **not** been so designated is solely at the Buyer's risk, and that Buyer is solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI has specifically designated certain components as meeting ISO/TS16949 requirements, mainly for automotive use. In any case of use of non-designated products, TI will not be responsible for any failure to meet ISO/TS16949.

### Products

Audio	<a href="http://www.ti.com/audio">www.ti.com/audio</a>
Amplifiers	<a href="http://amplifier.ti.com">amplifier.ti.com</a>
Data Converters	<a href="http://dataconverter.ti.com">dataconverter.ti.com</a>
DLP® Products	<a href="http://www.dlp.com">www.dlp.com</a>
DSP	<a href="http://dsp.ti.com">dsp.ti.com</a>
Clocks and Timers	<a href="http://www.ti.com/clocks">www.ti.com/clocks</a>
Interface	<a href="http://interface.ti.com">interface.ti.com</a>
Logic	<a href="http://logic.ti.com">logic.ti.com</a>
Power Mgmt	<a href="http://power.ti.com">power.ti.com</a>
Microcontrollers	<a href="http://microcontroller.ti.com">microcontroller.ti.com</a>
RFID	<a href="http://www.ti-rfid.com">www.ti-rfid.com</a>
OMAP Applications Processors	<a href="http://www.ti.com/omap">www.ti.com/omap</a>
Wireless Connectivity	<a href="http://www.ti.com/wirelessconnectivity">www.ti.com/wirelessconnectivity</a>

### Applications

Automotive and Transportation	<a href="http://www.ti.com/automotive">www.ti.com/automotive</a>
Communications and Telecom	<a href="http://www.ti.com/communications">www.ti.com/communications</a>
Computers and Peripherals	<a href="http://www.ti.com/computers">www.ti.com/computers</a>
Consumer Electronics	<a href="http://www.ti.com/consumer-apps">www.ti.com/consumer-apps</a>
Energy and Lighting	<a href="http://www.ti.com/energy">www.ti.com/energy</a>
Industrial	<a href="http://www.ti.com/industrial">www.ti.com/industrial</a>
Medical	<a href="http://www.ti.com/medical">www.ti.com/medical</a>
Security	<a href="http://www.ti.com/security">www.ti.com/security</a>
Space, Avionics and Defense	<a href="http://www.ti.com/space-avionics-defense">www.ti.com/space-avionics-defense</a>
Video and Imaging	<a href="http://www.ti.com/video">www.ti.com/video</a>

### TI E2E Community

[e2e.ti.com](http://e2e.ti.com)



## Features

- Bushing mount
- Optional AR pin feature
- Plastic or metal shaft and bushings
- Wirewound
- Solder lugs or PC pins
- Sealable (Full body seal)
- Designed for use in HMI applications

■ RoHS compliant\*

## 3590 - Precision Potentiometer

### Electrical Characteristics<sup>1</sup>

Standard Resistance Range.....	200 to 100 K ohms
Total Resistance Tolerance.....	±5 %
Independent Linearity.....	±0.25 %
Effective Electrical Angle.....	3600 ° +10 °, -0 °
Absolute Minimum Resistance.....	1 ohm or 0.1 % maximum (whichever is greater)
Noise.....	100 ohms ENR maximum
Dielectric Withstanding Voltage (MIL-STD-202, Method 301) Sea Level.....	1,500 VAC minimum
Power Rating (Voltage Limited By Power Dissipation or 450 VAC, Whichever is Less)	
+40 °C.....	2 watts
+125 °C.....	0 watt
Insulation Resistance (500 VDC).....	1,000 megohms minimum
Resolution.....	See recommended part numbers

### Environmental Characteristics<sup>1</sup>

Operating Temperature Range.....	-40 °C to +125 °C
Storage Temperature Range.....	-55 °C to +125 °C
Temperature Coefficient Over Storage Temperature Range <sup>2</sup> .....	±50 ppm/°C maximum/unit
Vibration.....	15 G
Wiper Bounce.....	0.1 millisecond maximum
Shock.....	50 G
Wiper Bounce.....	0.1 millisecond maximum
Load Life.....	1,000 hours, 2 watts
Total Resistance Shift.....	±2 % maximum
Rotational Life (No Load).....	1,000,000 shaft revolutions
Total Resistance Shift.....	±5 % maximum
Moisture Resistance (MIL-STD-202, Method 103, Condition B)	
Total Resistance Shift.....	±2 % maximum
IP Rating	
Sealed Versions (-3, -4, -7, and -8).....	IP 65
Unsealed Versions (-1 -2, -5, and -6).....	IP 40
Moisture Sensitivity Level.....	1
ESD Classification (HBM).....	N/A

### Mechanical Characteristics<sup>1</sup>

Stop Strength.....	.45 N-cm (64 oz.-in.) minimum
Mechanical Angle.....	3600 ° +10 °, -0 °
Torque	
Starting & Running.....	0.85 N-cm (1.2 oz.-in.) maximum (unsealed) 1.41 N-cm (2.0 oz.-in.) maximum (sealed)
Mounting.....	55-80 N-cm (5-7 lb.-in.) (plastic) 90-113 N-cm (8-10 in.-lb.) (metal)
Shaft Runout.....	0.13 mm (0.005 in.) T.I.R.
Lateral Runout.....	0.20 mm (0.008 in.) T.I.R.
Shaft End Play.....	0.25 mm (0.010 in.) T.I.R.
Shaft Radial Play.....	0.13 mm (0.005 in.) T.I.R.
Pilot Diameter Runout.....	0.08 mm (0.003 in.) T.I.R.
Backlash.....	1.0 ° maximum
Weight.....	Approximately 19 G
Terminals.....	Solder lugs or PC pins
Soldering Condition	
Manual Soldering.....	.96.5Sn/3.0Ag/0.5Cu solid wire or no-clean rosin cored wire; 370 °C (700 °F) max. for 3 seconds
Wave Soldering.....	.96.5Sn/3.0Ag/0.5Cu solder with no-clean flux; 260 °C (500 °F) max. for 5 seconds
Wash processes.....	Not recommended
Marking.....	Manufacturer's name and part number, resistance value and tolerance, linearity tolerance, wiring diagram, and date code.
Ganging (Multiple Section Potentiometers).....	1 cup maximum
Hardware.....	One lockwasher and one mounting nut is shipped with each potentiometer.

NOTE: For Anti-rotation pin add 91 after configuration dash number. Example: -2 becomes -291 to add AR pin.

<sup>1</sup> At room ambient: +25 °C nominal and 50 % relative humidity nominal, except as noted.

<sup>2</sup> Consult manufacturer for complete specification details for resistances below 1k ohms.

\*RoHS Directive 2002/95/EC Jan. 27, 2003 including annex and RoHS Recast 2011/65/EU June 8, 2011.

Specifications are subject to change without notice.

Users should verify actual device performance in their specific applications.

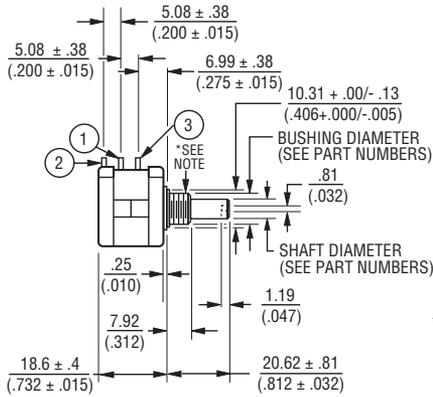
The products described herein and this document are subject to specific disclaimers as set forth on the last page of this document, and at [www.bourns.com/legal/disclaimer.pdf](http://www.bourns.com/legal/disclaimer.pdf).

# 3590 - Precision Potentiometer

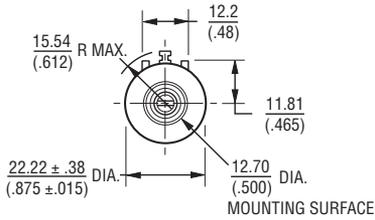
**BOURNS®**

## Product Dimensions

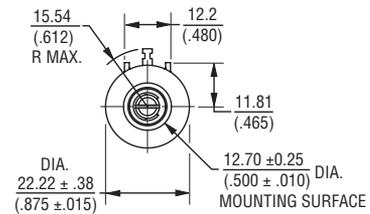
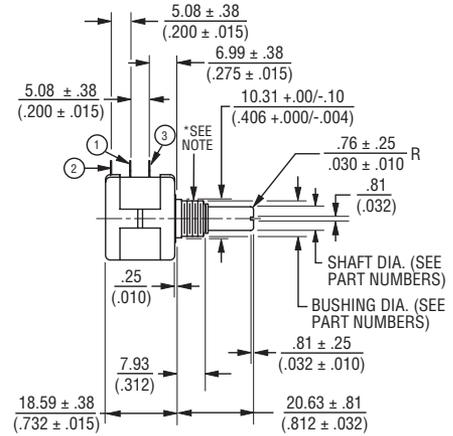
### -1, -3, -5, -7 Configurations



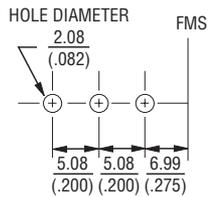
\*BUSHING THREAD SPECIFICATIONS:  
 -1, -2, -3, -4.....3/8-32 UNF  
 -5, -6, -7, -8.....M9 x 0.75-7H



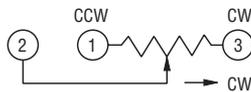
### -2, -4, -6, -8 Configurations



## Recommended PCB Layout



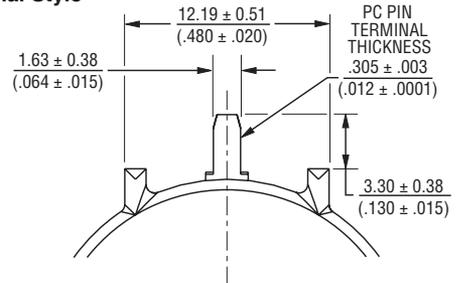
## Schematic



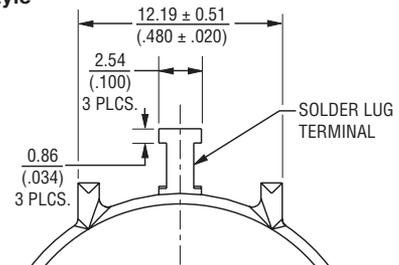
TOLERANCES: EXCEPT WHERE NOTED  
 DECIMALS: XX ± .508 (.02), .XXX ± .127 (.005)  
 FRACTIONS: ±1/64  
 DIMENSIONS: MM (IN.)

## Terminal Styles

### "P" Terminal Style



### "S" Terminal Style



# 3590 - Precision Potentiometer

**BOURNS®**

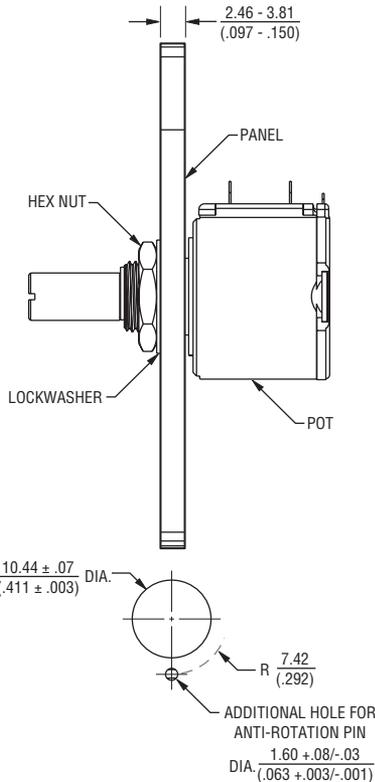
## Recommended Part Numbers

(Printed Circuit)	(Solder Lug)	(Solder Lug)	Resistance (Ω)	Resolution (%)
3590P-2-102L	<b>3590S-2-102L</b>	3590S-1-102L	<b>1,000</b>	<b>.029</b>
3590P-2-202L	<b>3590S-2-202L</b>	3590S-1-202L	<b>2,000</b>	<b>.023</b>
3590P-2-502L	<b>3590S-2-502L</b>	3590S-1-502L	<b>5,000</b>	<b>.025</b>
3590P-2-103L	<b>3590S-2-103L</b>	3590S-1-103L	<b>10,000</b>	<b>.020</b>
3590P-2-203L	<b>3590S-2-203L</b>	3590S-1-203L	<b>20,000</b>	<b>.019</b>
3590P-2-503L	<b>3590S-2-503L</b>	3590S-1-503L	<b>50,000</b>	<b>.013</b>
3590P-2-104L	<b>3590S-2-104L</b>	3590S-1-104L	<b>100,000</b>	<b>.009</b>

**BOLDFACE LISTINGS ARE IN STOCK AND READILY AVAILABLE THROUGH DISTRIBUTION. FOR OTHER OPTIONS CONSULT FACTORY.**  
 ROHS IDENTIFIER:  
 L = COMPLIANT

## Panel Thickness Dimensions

(For Bushing Mount Only)



Anti-rotation pin hole is shown at six o'clock position for reference only. The actual location is determined by the customer's application. Refer to the front view of the potentiometer to see the location of the optional A/R pin.

Panel thickness and hole diameters are recommended for best fit. However, customers may adjust the dimensions to suit their specific application.

## Shaft & Bushing Configurations

(Bushing - DxL, Shaft - D):

- (-1) Plastic Bushing (3/8 " x 5/16 ") and Shaft (.2480 + .001, - .002)
- (-2) **Metal Bushing (3/8 " x 5/16 ") and Shaft (.2497 + .0000, - .0009)**
- (-3) Sealed, Plastic Bushing (3/8 " x 5/16 ") and Shaft (.2480 + .001, - .002)
- (-4) Sealed, Metal Bushing (3/8 " x 5/16 ") and Shaft (.2497 + .0000, - .0009)
- (-5) Metric, Plastic Bushing (9 mm x 7.94 mm) and Shaft (6 mm + 0, - .076 mm)
- (-6) Metric, Metal Bushing (9 mm x 7.94 mm) and Shaft (6 mm + 0, - .023 mm)
- (-7) Metric, Sealed, Plastic Bushing (9 mm x 7.94 mm) and Shaft (6 mm + 0, - .076 mm)
- (-8) Metric, Sealed, Metal Bushing (9 mm x 7.94 mm) and Shaft (6 mm + 0, - .023 mm)

REV. 11/17

Specifications are subject to change without notice.

Users should verify actual device performance in their specific applications.

The products described herein and this document are subject to specific disclaimers as set forth on the last page of this document, and at [www.bourns.com/legal/disclaimer.pdf](http://www.bourns.com/legal/disclaimer.pdf).

This legal disclaimer applies to purchasers and users of Bourns® products manufactured by or on behalf of Bourns, Inc. and its affiliates (collectively, "Bourns").

Unless otherwise expressly indicated in writing, Bourns® products and data sheets relating thereto are subject to change without notice. Users should check for and obtain the latest relevant information before placing orders and should verify that such information is current and complete.

The characteristics and parameters of a Bourns® product set forth in its data sheet are based on laboratory conditions, and statements regarding the suitability of products for certain types of applications are based on Bourns' knowledge of typical requirements in generic applications. The characteristics and parameters of a Bourns® product in a user application may vary from the data sheet characteristics and parameters due to a combination of the Bourns® product with other components in the user's application or due to the environment of the user application itself. Such characteristics and parameters also can and do vary in different applications and actual performance may vary over time. Users should always verify actual performance of the Bourns® product in their specific devices and applications, and make their own independent judgments about how much additional test margin to design in to compensate for differences between laboratory and real world conditions.

Unless Bourns has explicitly designated an individual Bourns® product as meeting the requirements of a particular industry standard (e.g., ISO/TS 16949) or a particular qualification (e.g., UL listed or recognized), Bourns is not responsible for any failure of an individual Bourns® product to meet requirements of such industry standard or such particular qualification. Users of Bourns® products are responsible for ensuring compliance with safety-related requirements and standards applicable to their applications.

Bourns® products are not recommended, authorized or intended for use in nuclear, lifesaving, life-critical or life-sustaining applications, nor in any other applications where failure or malfunction may result in personal injury, death, or severe property or environmental damage. Unless expressly and specifically approved in writing by two authorized Bourns representatives on a case-by-case basis, use of any Bourns® products in such unauthorized applications is at the user's sole risk. Life-critical applications include devices identified by the U.S. Food and Drug Administration as Class III devices and generally equivalent classifications outside of the United States.

Bourns® standard products that are designed and tested for use in automotive applications will be described on the applicable data sheets as compliant with the applicable AEC-Q standard. Unless expressly and specifically approved in writing by two authorized Bourns representatives on a case-by-case basis, use of any other Bourns® standard products in an automotive application is not recommended, authorized or intended and will be at the user's sole risk.

Bourns® standard products are not tested to comply with United States Federal Aviation Administration standards generally or any other generally equivalent governmental organization standard applicable to products designed or manufactured for use in aircraft or space applications. Bourns® standard products that are designed and tested for use in aircraft or space applications will be described on the applicable data sheets as compliant with the RTCA DO-160 standard. Unless expressly and specifically approved in writing by two authorized Bourns representatives on a case-by-case basis, use of any other Bourns® standard product in an aircraft or space application is not recommended, authorized or intended and will be at the user's sole risk.

The use and level of testing applicable to Bourns® custom products shall be negotiated on a case-by-case basis by Bourns and the user for which such Bourns® custom products are specially designed. Absent a written agreement between Bourns and the user regarding the use and level of such testing, the provisions above applicable to Bourns® standard products shall also apply to such Bourns® custom products.

Users shall not sell, transfer, export or re-export any Bourns® products or technology for use in activities which involve the design, development, production, use or stockpiling of nuclear, chemical or biological weapons or missiles, nor shall they use Bourns® products or technology in any facility which engages in activities relating to such devices. The foregoing restrictions apply to all uses and applications that violate national or international prohibitions, including embargos or international regulations. Further, Bourns® products, technology or technical data may not under any circumstance be exported or re-exported to countries subject to international sanctions or embargoes, and Bourns® products may not, without prior authorization from Bourns and/or the U.S. Government, be resold, transferred, or re-exported to any party not eligible to receive U.S. commodities, software, and technical data.

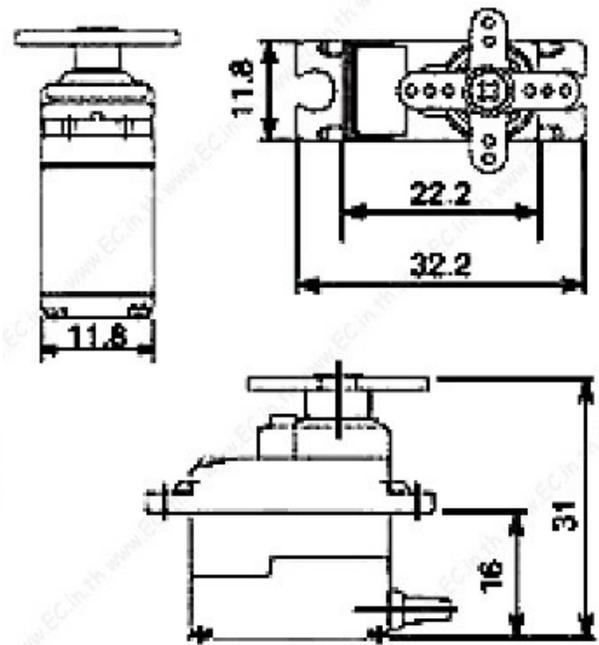
To the maximum extent permitted by applicable law, Bourns disclaims (i) any and all liability arising out of the application or use of any Bourns® standard product, (ii) any and all liability, including, without limitation, special, punitive, consequential or incidental damages, and (iii) any and all implied warranties, including implied warranties of fitness for particular purpose, non-infringement and merchantability.

*For your convenience, copies of this Legal Disclaimer Notice with German, Spanish, Japanese, Traditional Chinese and Simplified Chinese bilingual versions are available at:*

Web Page: <http://www.bourns.com/legal/disclaimers-terms-and-policies>

PDF: <http://www.bourns.com/docs/Legal/disclaimer.pdf>

# SG90 9 g Micro Servo

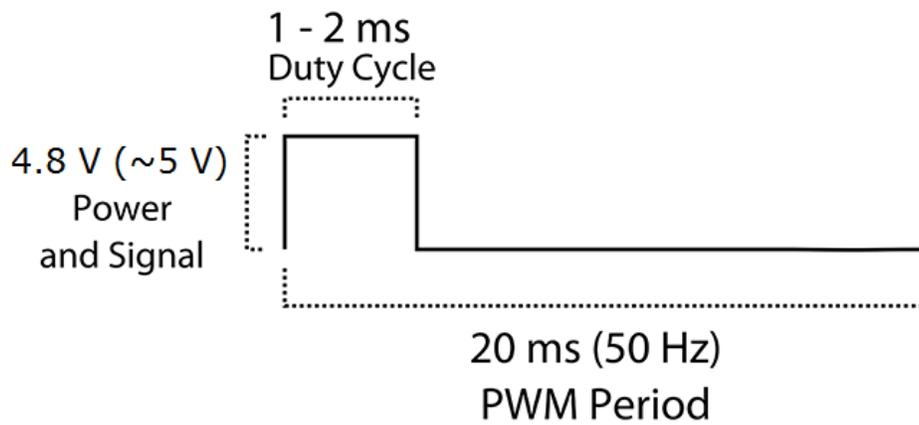
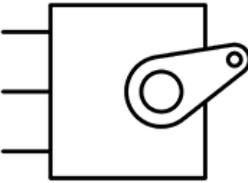


Tiny and lightweight with high output power. Servo can rotate approximately 180 degrees (90 in each direction), and works just like the standard kinds but *smaller*. You can use any servo code, hardware or library to control these servos. Good for beginners who want to make stuff move without building a motor controller with feedback & gear box, especially since it will fit in small places. It comes with a 3 horns (arms) and hardware.

## Specifications

- Weight: 9 g
- Dimension: 22.2 x 11.8 x 31 mm approx.
- Stall torque: 1.8 kgf·cm
- Operating speed: 0.1 s/60 degree
- Operating voltage: 4.8 V (~5V)
- Dead band width: 10  $\mu$ s
- Temperature range: 0 °C – 55 °C

PWM=Orange (  $\square$  )  
Vcc = Red ( + )  
Ground=Brown ( - )



Position "0" (1.5 ms pulse) is middle, "90" (~2 ms pulse) is all the way to the right, "-90" (~1 ms pulse) is all the way to the left.

# Stepper Motor NEMA 17

This document describes mechanical and electrical specifications for PBC Linear stepper motors; including standard, hollow, and extended shaft variations.

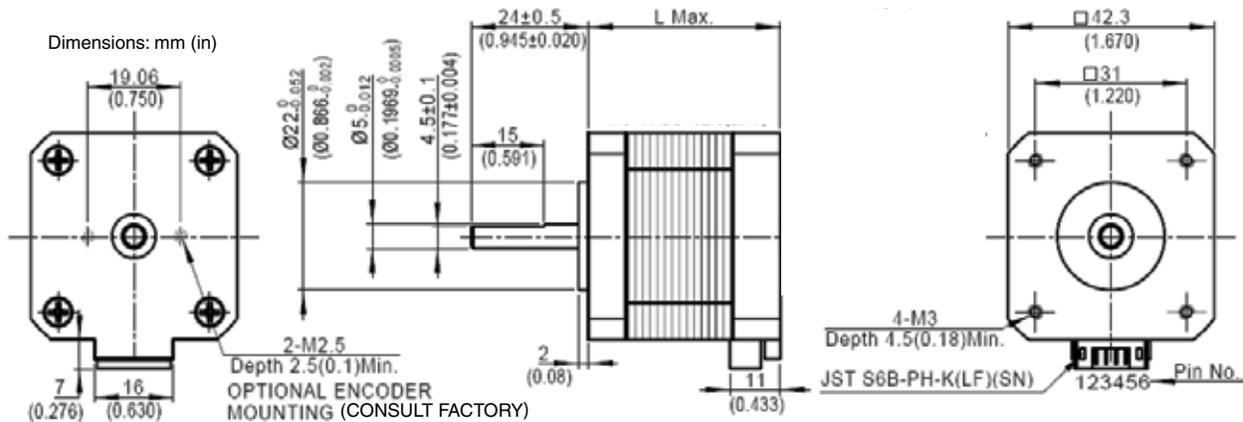


Phases	2
Steps/Revolution	200
Step Accuracy	±5%
Shaft Load	20,000 Hours at 1000 RPM
Axial	25 N (5.6 lbs.) Push 65 N (15 lbs.) Pull
Radial	29 N (6.5 lbs.) At Flat Center
IP Rating	40
Approvals	RoHS
Operating Temp	-20° C to +40° C
Insulation Class	B, 130° C
Insulation Resistance	100 MegOhms

Standard shaft motor shown.

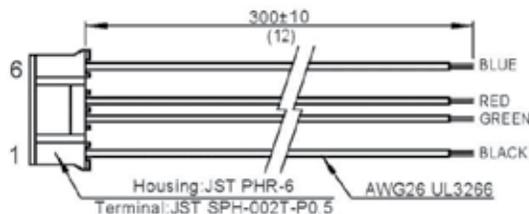
Description	Length	Mounted Rated Current	Mounted Holding Torque		Winding Ohms   mH		Detent Torque		Rotor Inertia		Motor Weight	
			Nm Typ.	oz-in Typ.	±10% @ 20°C Typ.	Typ.	mNm	oz-in	g cm2	oz-in2	kg	lbs
(Stack)	"L" Max	Amps										
Single	39.8 mm (1.57 in)	2	0.48	68	1.04	2.2	15	2.1	57	0.31	0.28	0.62
Double	48.3 mm (1.90 in)	2	0.63	89	1.3	2.9	25	3.5	82	0.45	0.36	0.79
Triple	62.8 mm (2.47 in)	2	0.83	120	1.49	3.8	30	4.2	123	0.67	0.6	1.3

\*All standard motors have plug connector. Consult factory for other options.



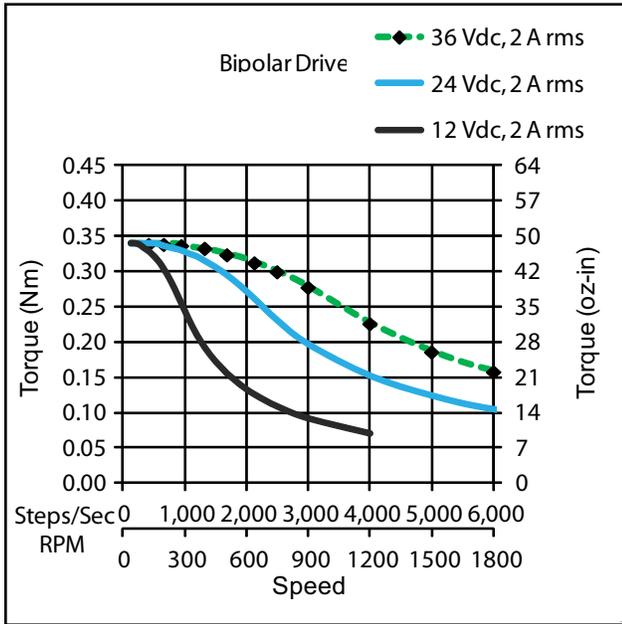
Standard shaft dimensions shown. All other dimensions apply to hollow and extended shaft options.

Dimensions: mm (in)  
4 Lead Connector, PBC Part#6200490  
(Consult factory for optional motor connectors)

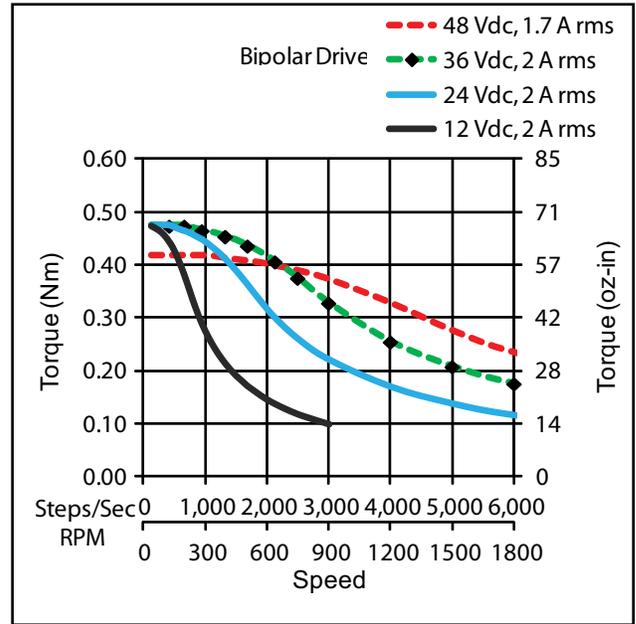


# NEMA 17 Stepper Motor

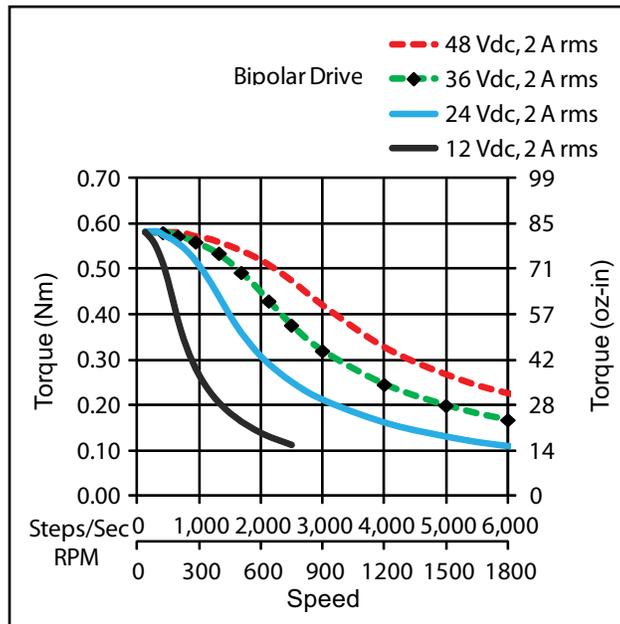
\*Performance curves apply to continuous duty cycles.  
Consult factory for intermittent cycles or other voltages.



Single Stack



Double Stack



Triple Stack

# Stepper Motor NEMA 23

This document describes mechanical and electrical specifications for PBC Linear stepper motors; including standard, hollow, and extended shaft variations.

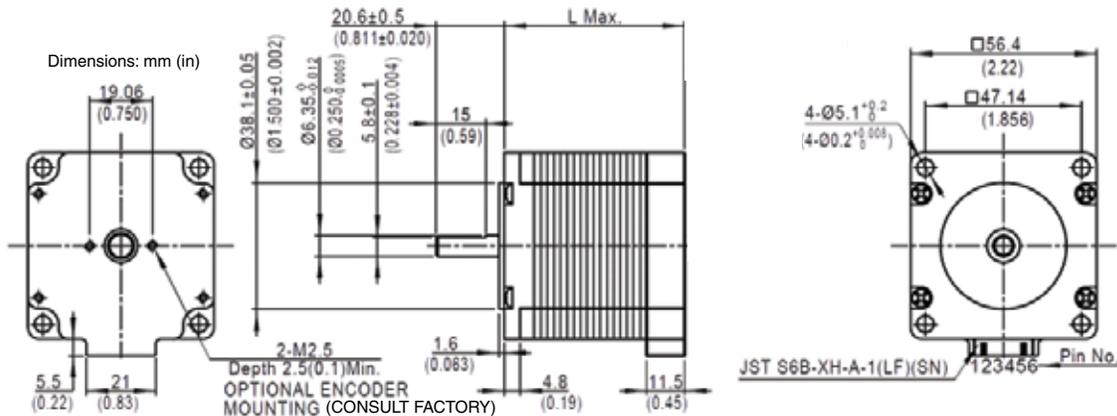


Standard shaft motor shown.

Phases	2
Steps/Revolution	200
Step Accuracy	±5%
Shaft Load	20,000 Hours at 1000 RPM
Axial	40 N (9 lbs.) Push 130 N (30 lbs.) Pull
Radial	70 N (15.5 lbs.) At Flat Center
IP Rating	40
Approvals	RoHS
Operating Temp	-20° C to +40° C
Insulation Class	B, 130° C
Insulation Resistance	100 MegOhms

Description	Length	Mounted Rated Current	Mounted Holding Torque		Winding Ohms mH		Detent Torque		Rotor Inertia		Motor Weight	
			Nm Typ.	oz-in Typ.	±10% @ 20°C	Typ.	mNm	oz-in	g cm2	oz-in2	kg	lbs
(Stack)	"L" Max	Amps										
Single	55.0 mm (2.17 in)	2.2	1.50	210	1.6	6.9	45	6.4	220	1.2	0.6	1.3
Double	77.0 mm (3.03 in)	3	2.30	330	1.1	4.5	75	11	390	2.1	1	2.2
Power Plus (Triple)	77.0 mm (3.03 in)	3	3.30	470	1.1	3.7	150	21	390	2.1	1.1	2.4

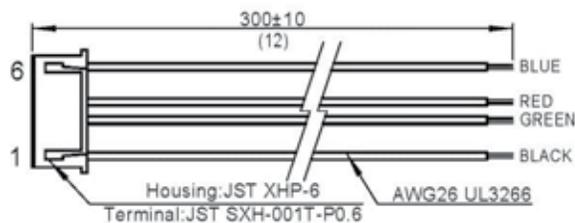
\*All standard motors have plug connector. Consult factory for other options.



Motor with leads: Lead wire is 22 AWG UL3266, 300 ±10 (12 ±.5) long

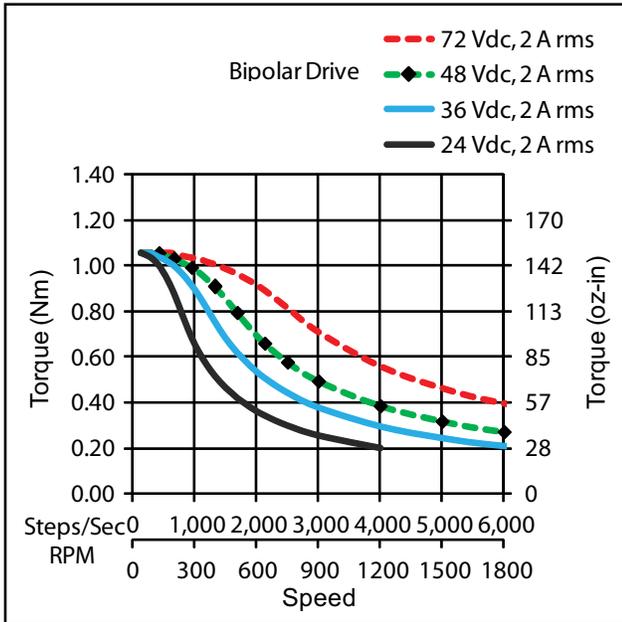
Standard shaft dimensions shown. All other dimensions apply to hollow and extended shaft options.

Dimensions: mm (in)  
4 Lead Connector, PBC Part#6200491  
(Consult factory for optional motor connectors)

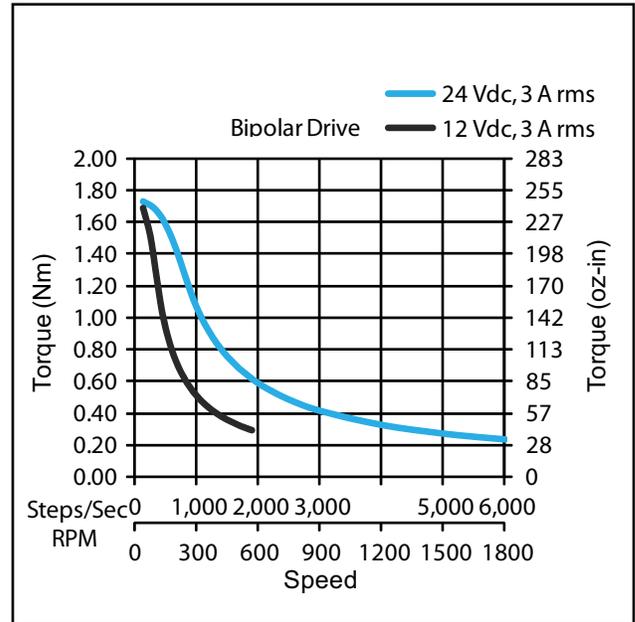


# NEMA 23 Stepper Motor

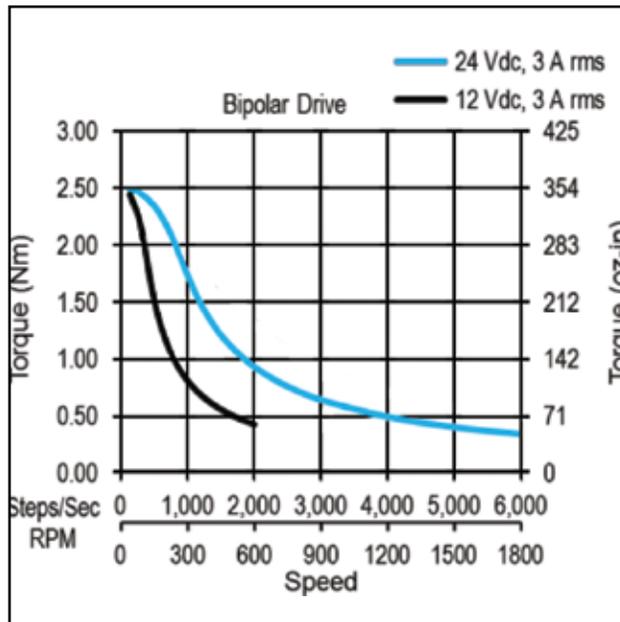
\*Performance curves apply to continuous duty cycles.  
Consult factory for intermittent cycles or other voltages.



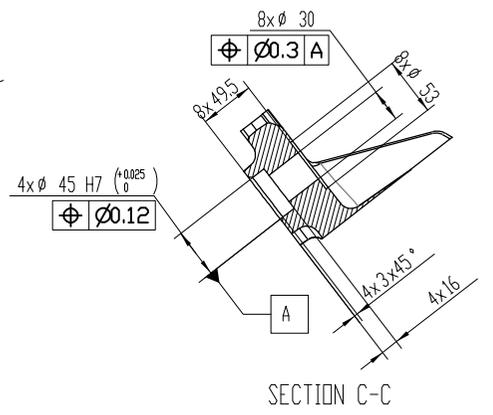
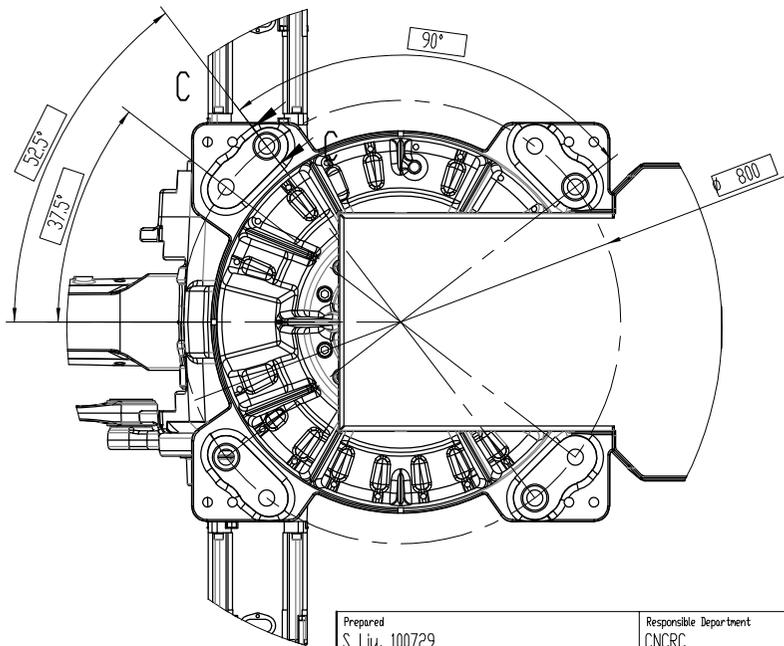
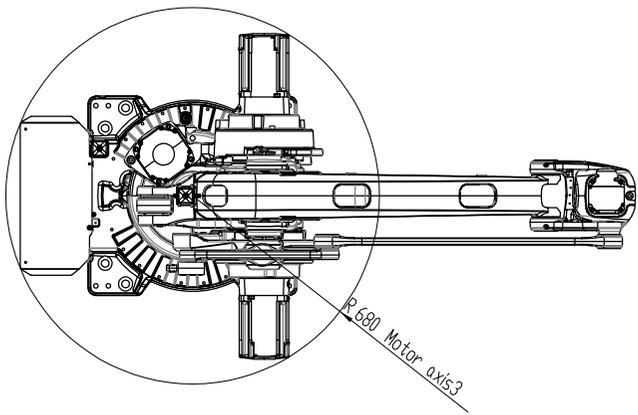
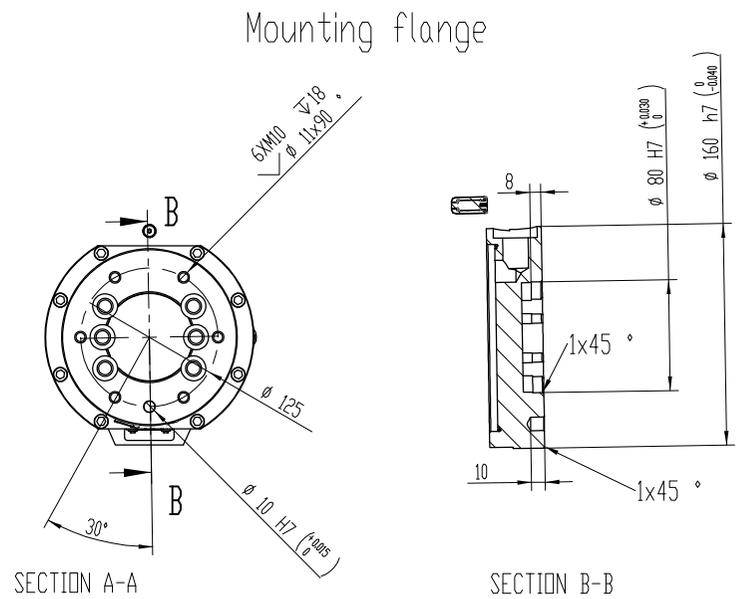
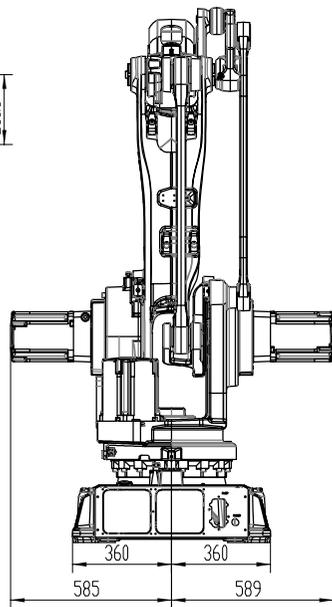
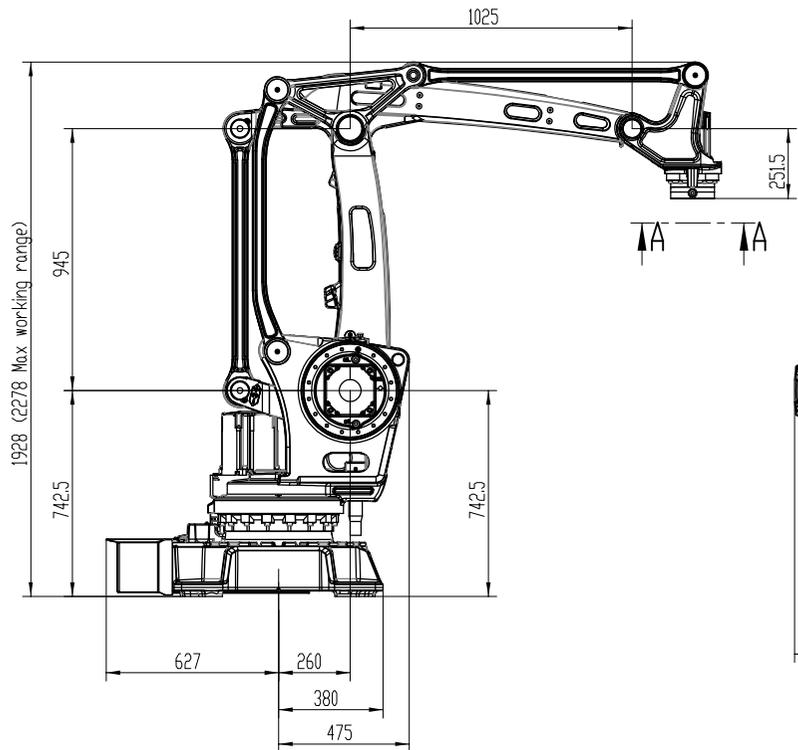
**Single Stack**



**Double Stack**



**Power Plus (Triple Stack)**



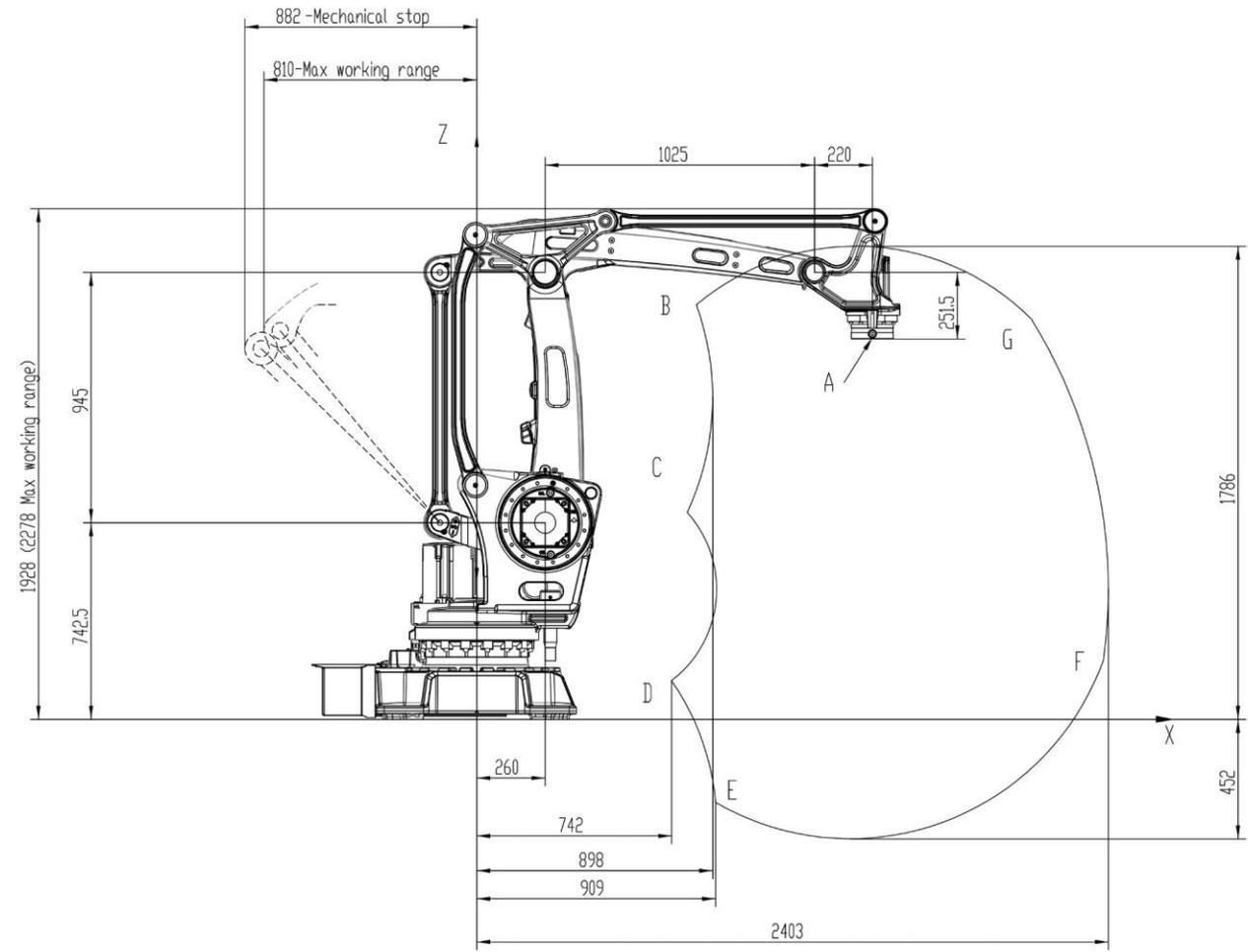
New	Latest Revision
-----	-----------------

Prepared S Liu, 100729	Responsible Department CNCRC	Description Main dimension IRB460	SWD	Drawing	Sheet 1
Approved P Mikaelsson, 100730	Take over department				
Status Approved	ECD number 500000042094	Document no. 3HAC036446-002			Rev. ind. 00



We reserve all rights in this document and in the information contained therein. Reproduction, use or disclosure to third parties without express authority is strictly forbidden. ©Copyright 2003 ABB

We reserve all rights in this document and in the information contained therein. Reproduction, use or modification of this document without the written authority of ABB is strictly forbidden. © Copyright 2002 ABB



Pos letter	Position (mm)		Angles(°)	
	X	Z	Axis2	Axis3
A	1505	1437	0	0
B	836	1565	-40	-20
C	802	782	-40	25
D	742	145	55	120
E	909	-314	85	120
F	2385	223	85	20
G	2111	1510	45	-20

Range of movement	
Axis1:	-165° to +165°
Axis2:	-40° to +85°
Axis3:	-20° to +120°
Axis2-3:	+25° to +155°
Axis6:	-300° to +300°

New    Latest Revision	Prepared S Liu, 100729	Responsible Department CNCRC	Description SWD Drawing		
	Approved P Mikaelsson, 100730	Take over department	Main dimension IRB460		Sheet 3
	Status Approved	ECD number 500000042094			No. of sh. 4
	<b>ABB</b>				Document no. 3HAC036446-002

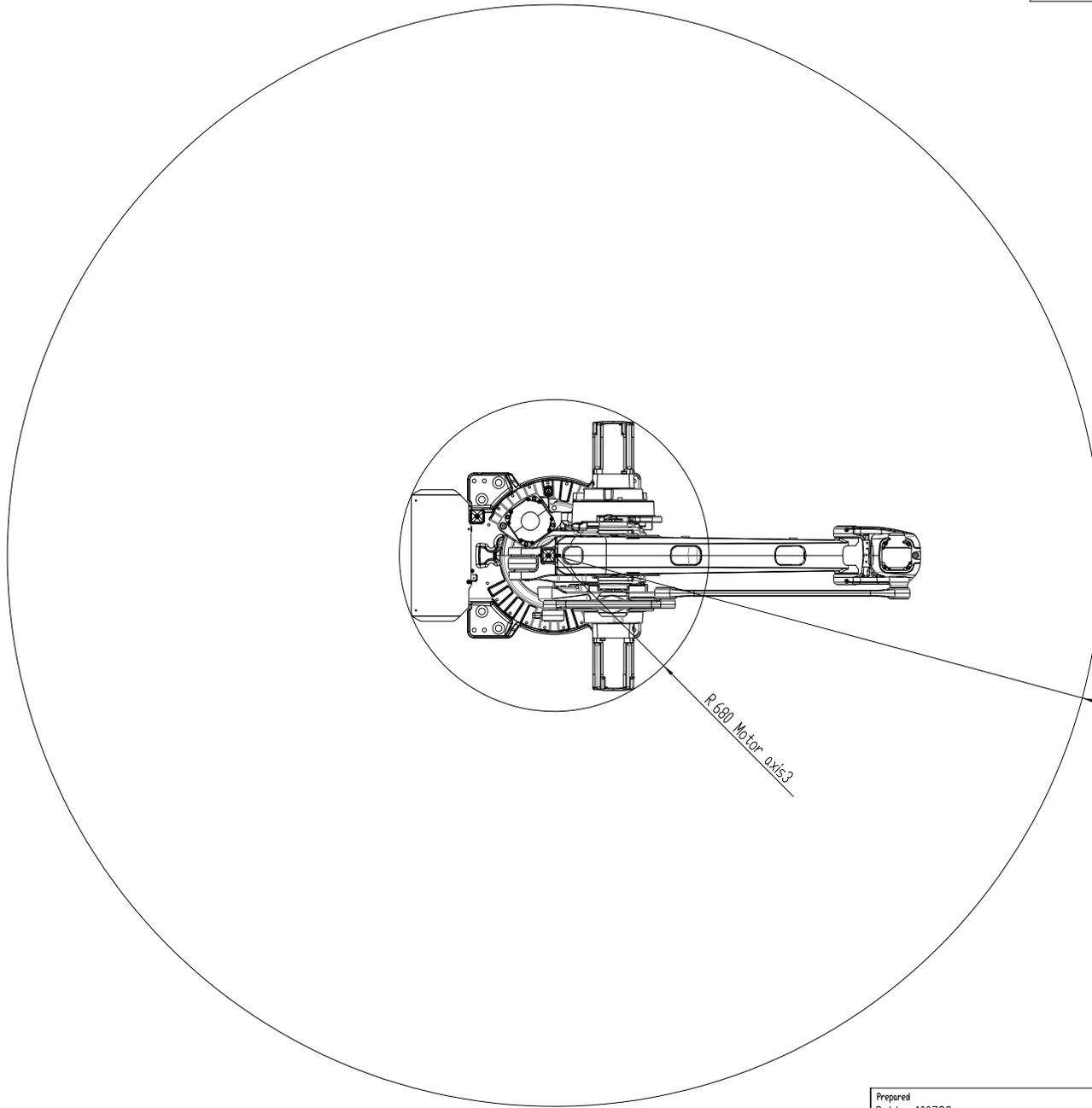
A

B

C

D

We reserve all rights in this document and in the information contained therein. Reproduction, use or disclosure to third parties without express authority is strictly forbidden. © Copyright 2001 ABB



New

Latest Revision

Prepared S Liu, 100729	Responsible Department CNCRC	Description SWD Drawing	
Approved P Mikaelsson, 100730	Take over department		
Status Approved	ECO number 500000042094	Main dimension IRB460	No. of sh 4
			Document no. 3HAC036446-002

1

2

3

4

5

6

8 7 6 5 4 3 2 1

F

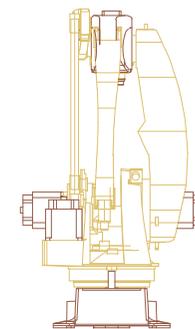
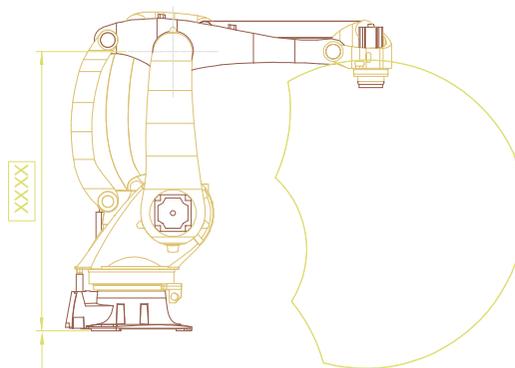
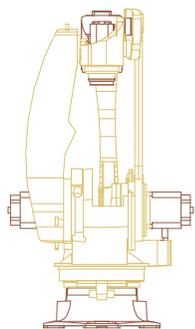
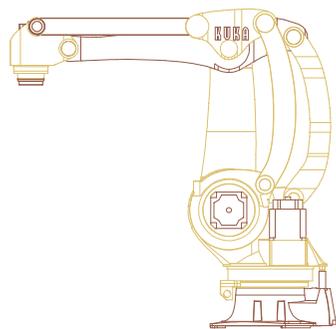
E

D

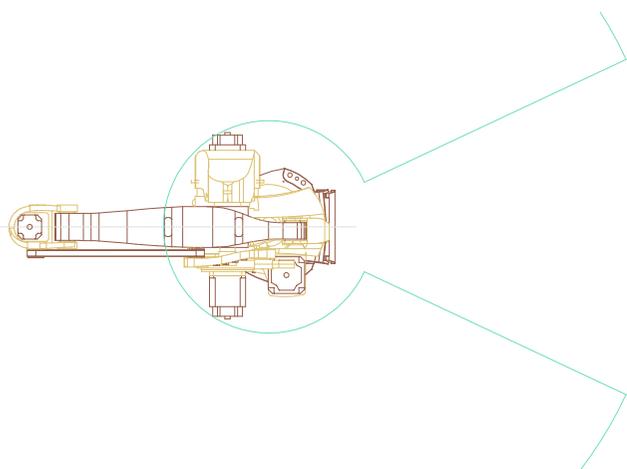
C

B

A



Bei Skalierung beachten  
 Originalmass: 1470 mm  
 Caution when scaling  
 original dimension: 1470 mm



8 7 6 5 4 3 2 1

Es ist anzuwenden, nach DIN EN ISO 15926-1 Copyright notice according to DIN EN ISO 15926-1	<b>KUKA</b>		Gewicht/Weight		Benennung/Description		
	Firma/Company KUKA Roboter GmbH		Maßstab/Scale 1:20		Dokumentation KR 40 PA documentation KR40 PA Symbole / symbols		
	Kundezeichnung Customer-Drawing		A-Teil Part		Zeichnungs-Nr. Drawing-No.		
	Datum Date		Name		Mat-Nr. Mat-No.		
Entwurf Drawn		A-Teil Part		Status		Blatt Sheet	
Bearbeitet Prepared		Dateiname Dateiname		Anzahl Rev. Nr.		von of	
Geprüft Checked		Dateiname Dateiname		Anzahl Rev. Nr.		von of	

F

E

D

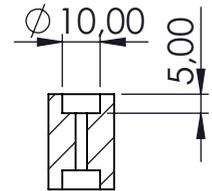
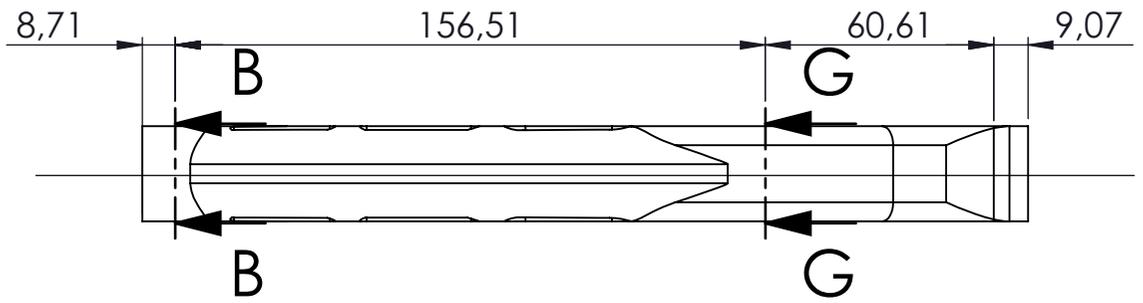
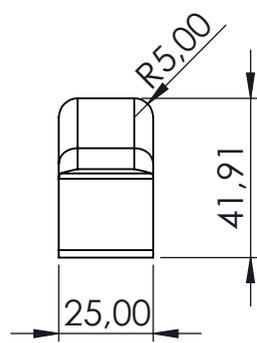
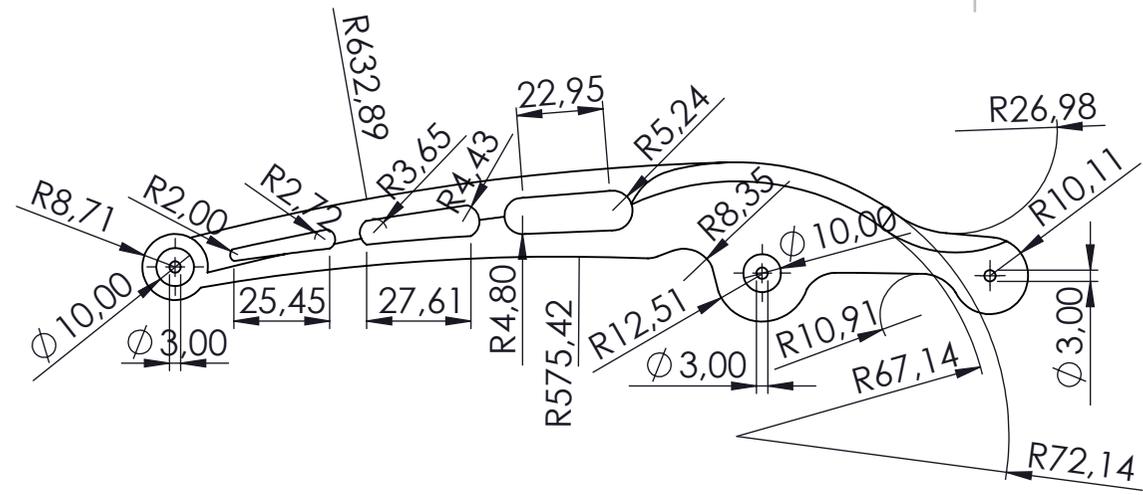
C

B

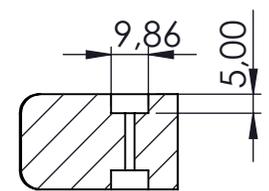
A

## II. PLANOS DEL PROYECTO

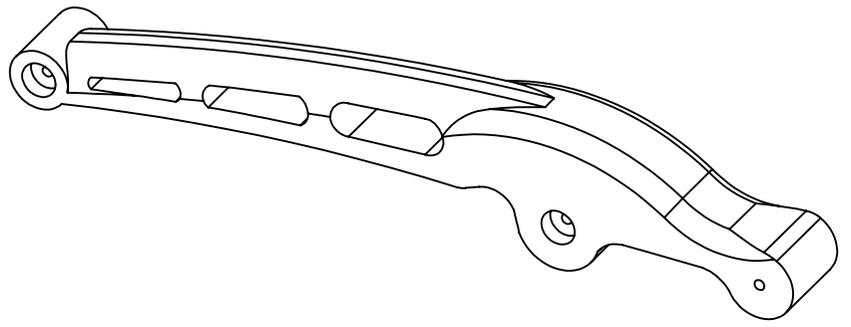
---



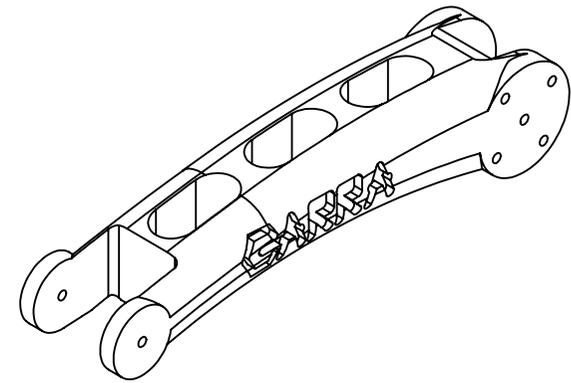
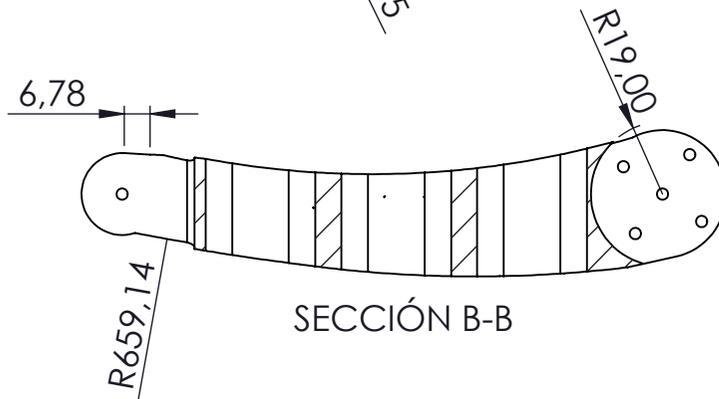
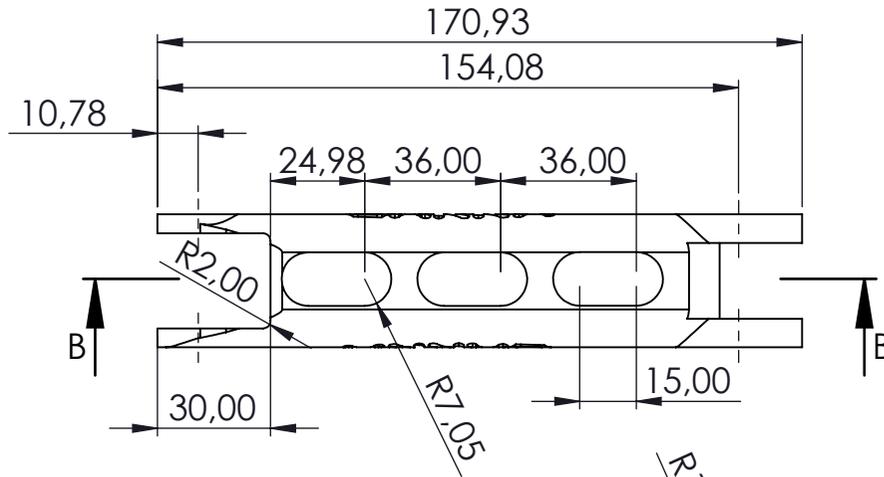
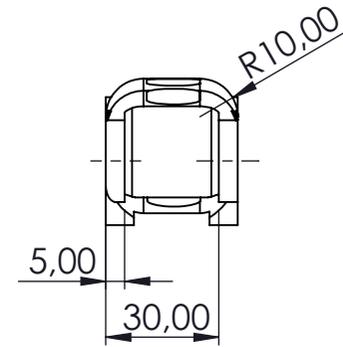
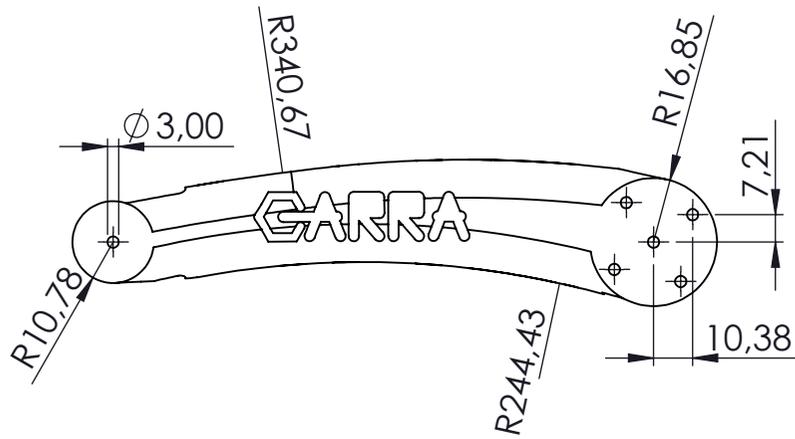
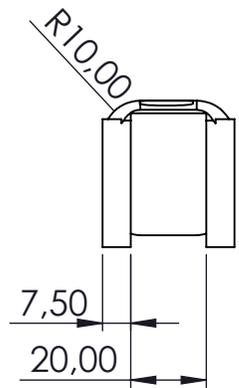
SECCIÓN B-B



SECCIÓN G-G

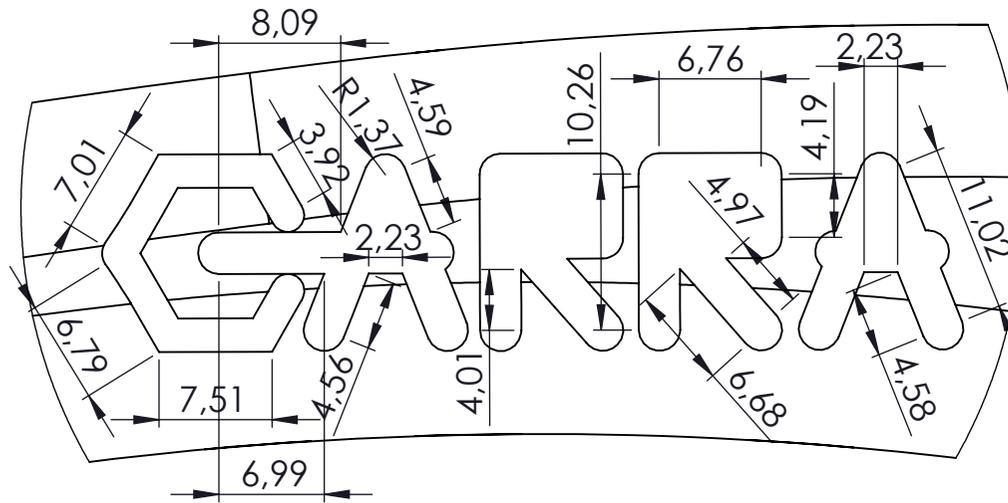
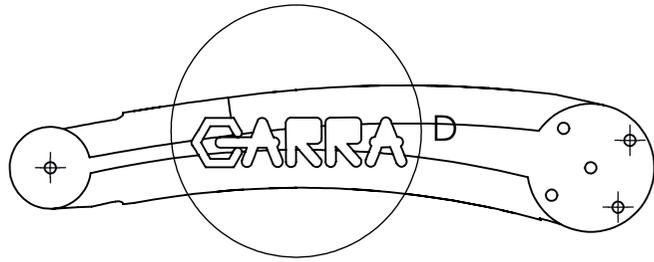


TRABAJO FIN DE GRADO INGENIERIA ELECTRONICA INDUSTRIAL Y AUTOMÁTICA		UNIVERSIDAD POLITÉCNICA DE VALENCIA ESCUELA TÉCNICA SUPERIOR DE INGENIERIA DEL DISEÑO	
AUTOR MARC GARCIA ABELLA		TÍTULO <b>ANTEBRAZO</b>	
FECHA 20-06-2018		N.º DE PLANO <b>01</b>	
MATERIAL Z-ULTRAT BLANCO		ESCALA 1:2	



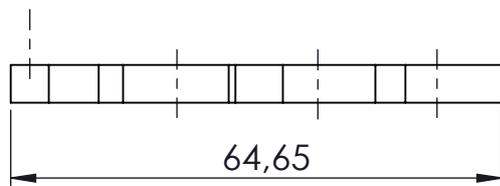
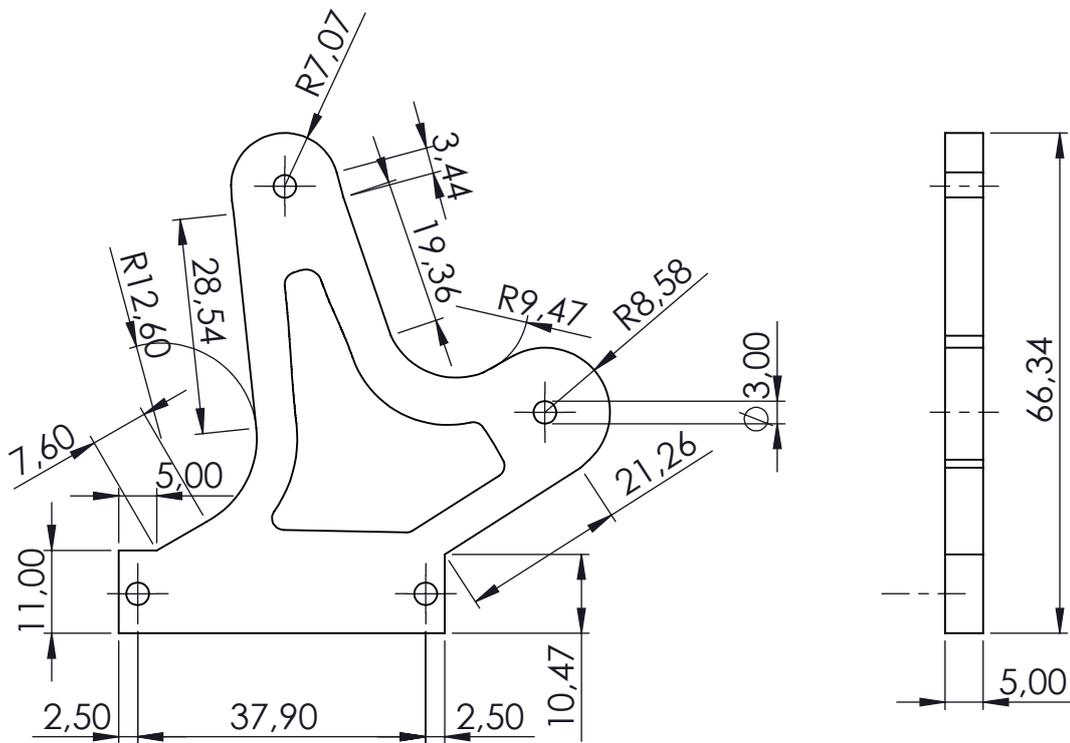
SECCIÓN B-B

TRABAJO FIN DE GRADO INGENIERIA ELECTRONICA INDUSTRIAL Y AUTOMÁTICA	UNIVERSIDAD POLITÉCNICA DE VALENCIA ESCUELA TÉCNICA SUPERIOR DE INGENIERIA DEL DISEÑO	
AUTOR MARC GARCIA ABELLA	TÍTULO <b>BRAZO</b>	
FECHA 20-06-2018		
MATERIAL Z-ULTRAT NEGRO	N.º DE PLANO <b>02</b>	ESCALA 1:2



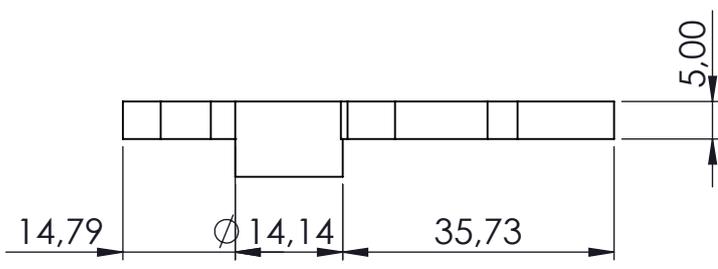
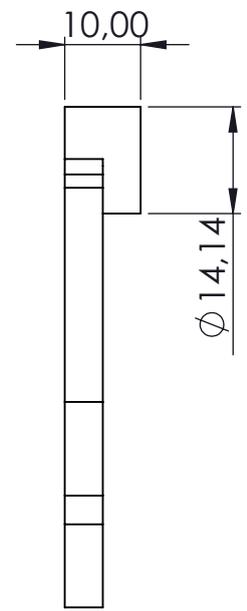
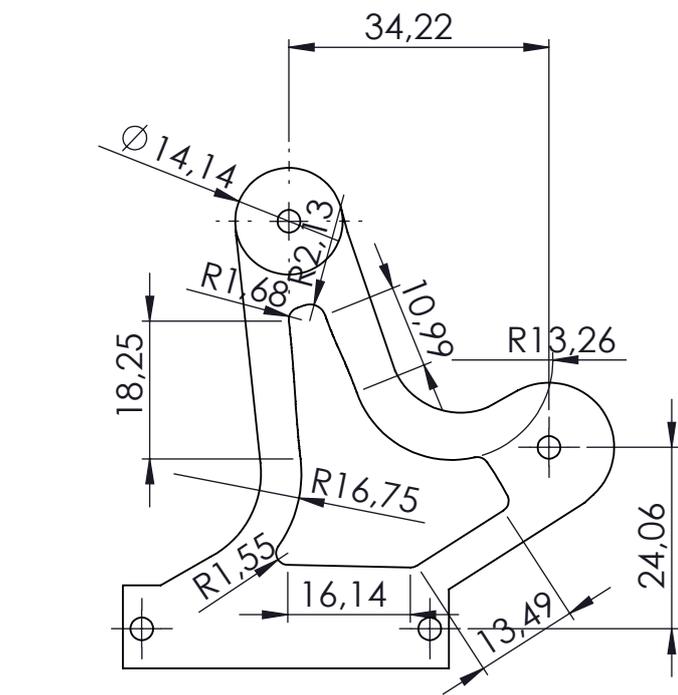
DETALLE D  
ESCALA 2 : 1

TRABAJO FIN DE GRADO INGENIERIA ELECTRONICA INDUSTRIAL Y AUTOMÁTICA		UNIVERSIDAD POLITÉCNICA DE VALENCIA ESCUELA TÉCNICA SUPERIOR DE INGENIERIA DEL DISEÑO	
AUTOR MARC GARCIA ABELLA		TÍTULO <b>LOGO GARRA</b>	
FECHA 20-06-2018		N.º DE PLANO <b>03</b>	
MATERIAL N/A		ESCALA 1:2	



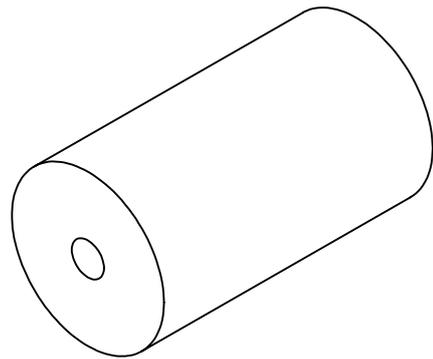
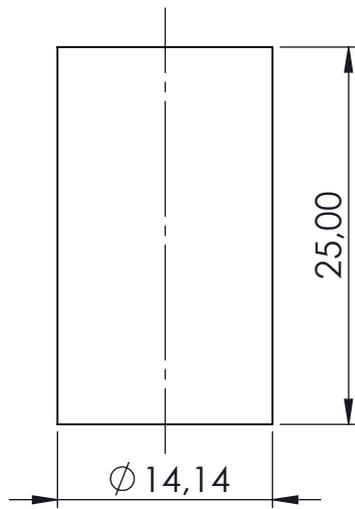
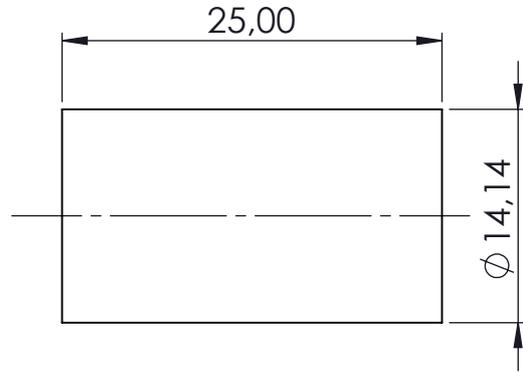
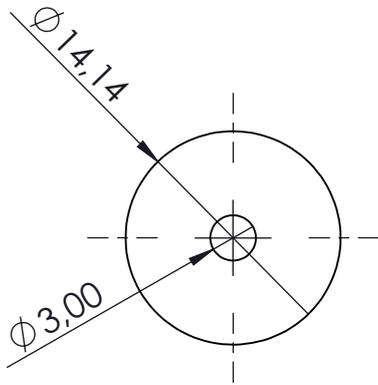
NOTA: PLANO LATERAL IZQUIERDO  
MUÑECA, ESTRUCTURA INTERIOR  
DETALLADA EN PLANO 5

TRABAJO FIN DE GRADO INGENIERIA ELECTRONICA INDUSTRIAL Y AUTOMÁTICA	UNIVERSIDAD POLITÉCNICA DE VALENCIA ESCUELA TÉCNICA SUPERIOR DE INGENIERIA DEL DISEÑO	
AUTOR MARC GARCIA ABELLA	TÍTULO MUÑECA LATERAL-L	
FECHA 20-06-2018		
MATERIAL PLA BQ NEGRO	N.º DE PLANO 04	ESCALA 1:1

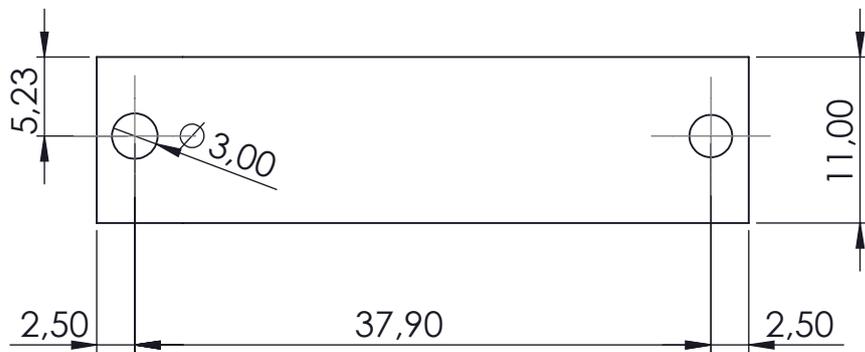
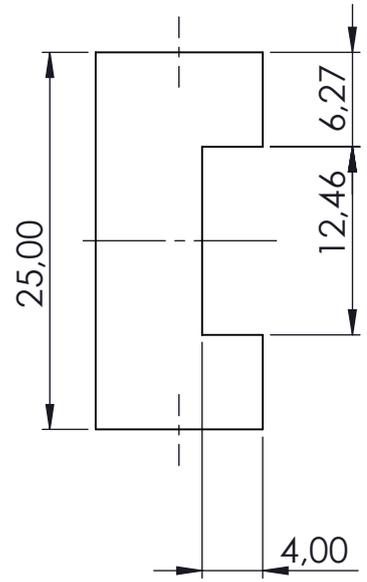
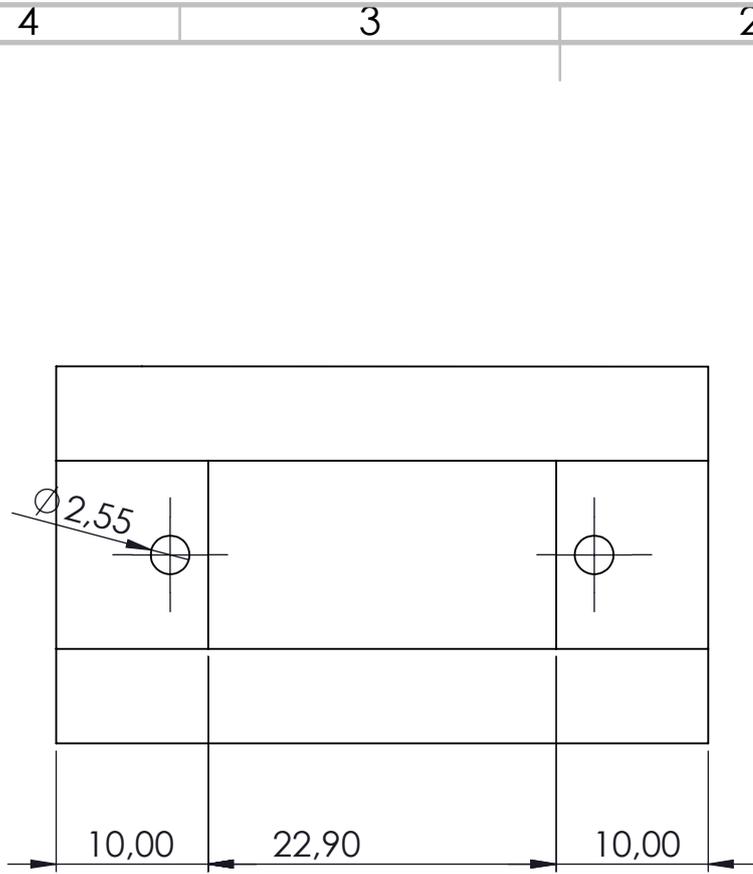


NOTA: PLANO LATERAL DERECHO  
MUÑECA, ESTRUCTURA EXTERIOR  
DETALLADA EN PLANO 4

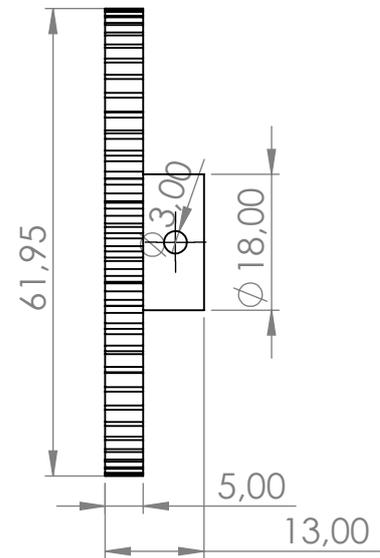
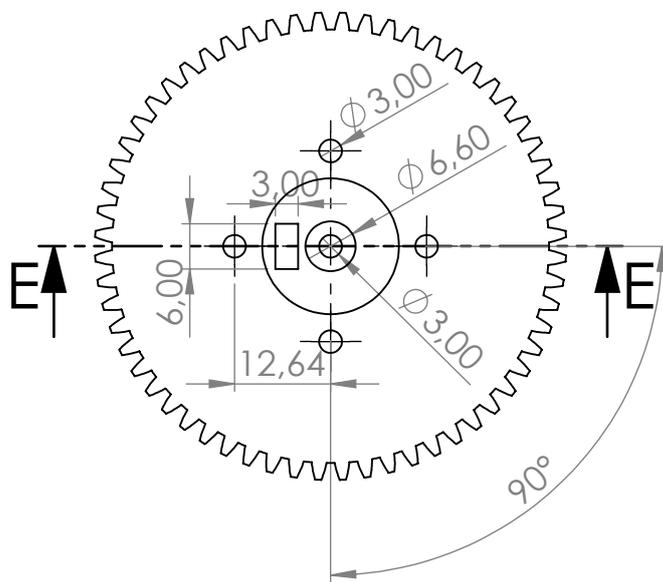
TRABAJO FIN DE GRADO INGENIERIA ELECTRONICA INDUSTRIAL Y AUTOMÁTICA	UNIVERSIDAD POLITÉCNICA DE VALENCIA ESCUELA TÉCNICA SUPERIOR DE INGENIERIA DEL DISEÑO	
AUTOR MARC GARCIA ABELLA	TÍTULO <b>MUÑECA LATERAL-R</b>	
FECHA 20-06-2018	N.º DE PLANO <b>05</b>	
MATERIAL PLA BQ NEGRO	ESCALA 1:1	



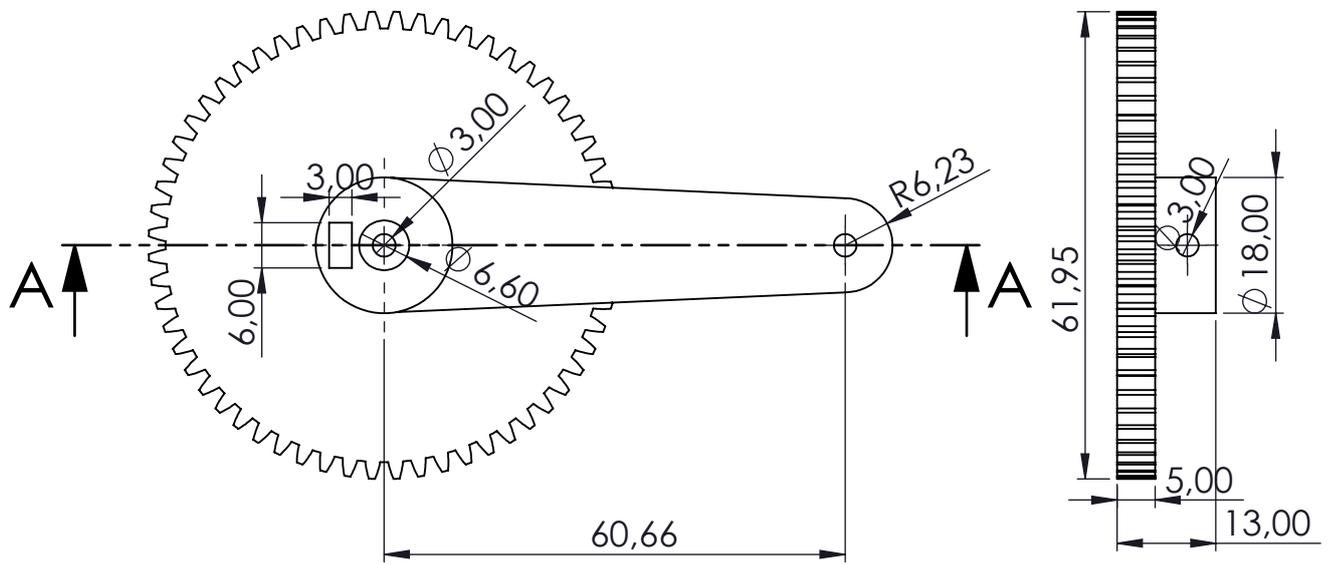
TRABAJO FIN DE GRADO INGENIERIA ELECTRONICA INDUSTRIAL Y AUTOMÁTICA	UNIVERSIDAD POLITÉCNICA DE VALENCIA ESCUELA TÉCNICA SUPERIOR DE INGENIERIA DEL DISEÑO	
AUTOR MARC GARCIA ABELLA	TÍTULO <b>SEPARAD. CILINDRO</b>	
FECHA 20-06-2018		
MATERIAL PLA BQ NEGRO	N.º DE PLANO <b>06</b>	ESCALA 2:1



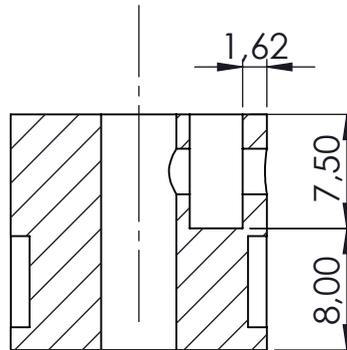
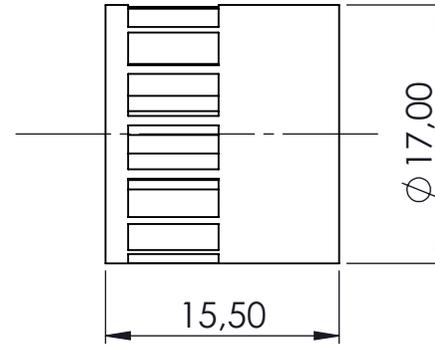
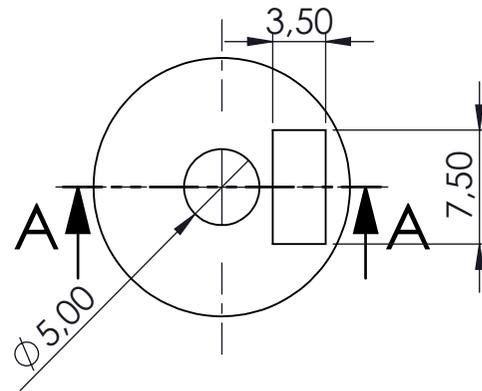
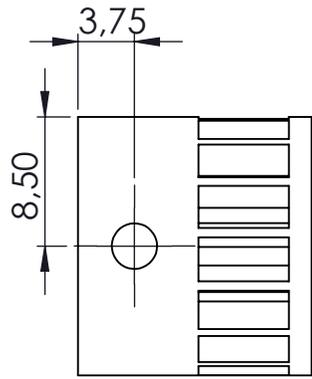
TRABAJO FIN DE GRADO INGENIERIA ELECTRONICA INDUSTRIAL Y AUTOMÁTICA	UNIVERSIDAD POLITÉCNICA DE VALENCIA ESCUELA TÉCNICA SUPERIOR DE INGENIERIA DEL DISEÑO	
AUTOR MARC GARCIA ABELLA	TÍTULO <b>ZOCALO SERVO</b>	
FECHA 20-06-2018		
MATERIAL PLA BQ NEGRO	N.º DE PLANO <b>07</b>	ESCALA 2:1



TRABAJO FIN DE GRADO INGENIERIA ELECTRONICA INDUSTRIAL Y AUTOMÁTICA		UNIVERSIDAD POLITÉCNICA DE VALENCIA ESCUELA TÉCNICA SUPERIOR DE INGENIERIA DEL DISEÑO	
AUTOR MARC GARCIA ABELLA		TÍTULO <b>CORONA M2</b>	
FECHA 20-06-2018		N.º DE PLANO <b>08</b>	
MATERIAL PLA BQ NEGRO		ESCALA 1:1	

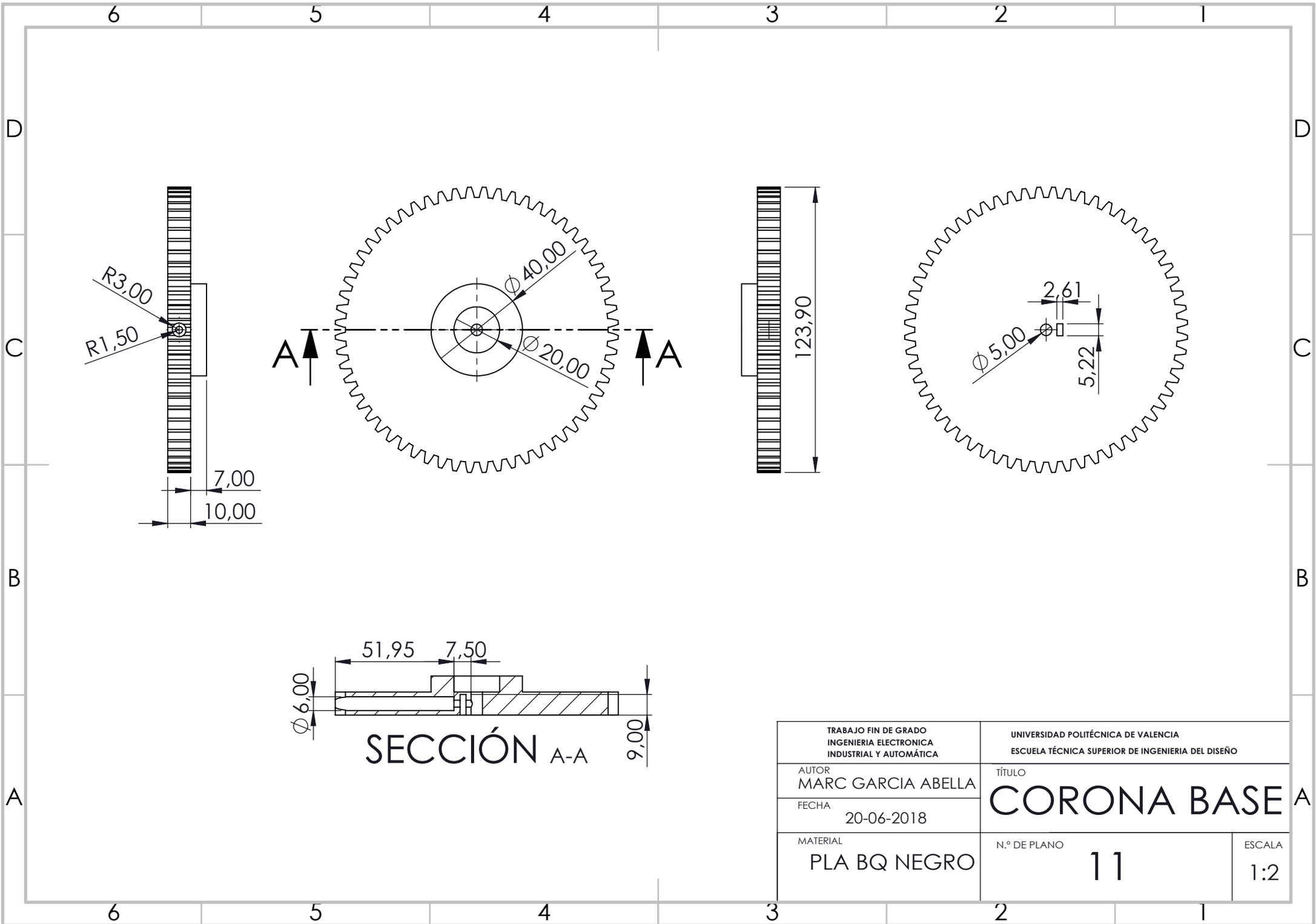


TRABAJO FIN DE GRADO INGENIERIA ELECTRONICA INDUSTRIAL Y AUTOMÁTICA	UNIVERSIDAD POLITÉCNICA DE VALENCIA ESCUELA TÉCNICA SUPERIOR DE INGENIERIA DEL DISEÑO	
AUTOR MARC GARCIA ABELLA	TÍTULO <b>CORONA M3</b>	
FECHA 20-06-2018	N.º DE PLANO <b>09</b>	
MATERIAL PLA BQ NEGRO	ESCALA 1:1	



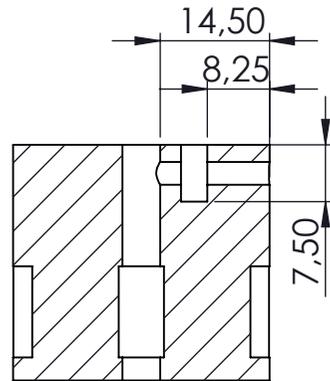
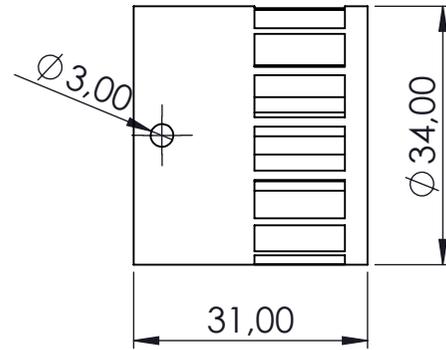
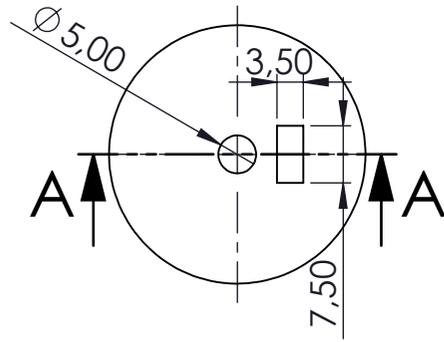
SECCIÓN A-A

TRABAJO FIN DE GRADO INGENIERIA ELECTRONICA INDUSTRIAL Y AUTOMÁTICA		UNIVERSIDAD POLITÉCNICA DE VALENCIA ESCUELA TÉCNICA SUPERIOR DE INGENIERIA DEL DISEÑO	
AUTOR MARC GARCIA ABELLA		TÍTULO <b>PIÑÓN M2-M3</b>	
FECHA 20-06-2018			
MATERIAL PLA BQ NEGRO		N.º DE PLANO <b>10</b>	ESCALA <b>2:1</b>



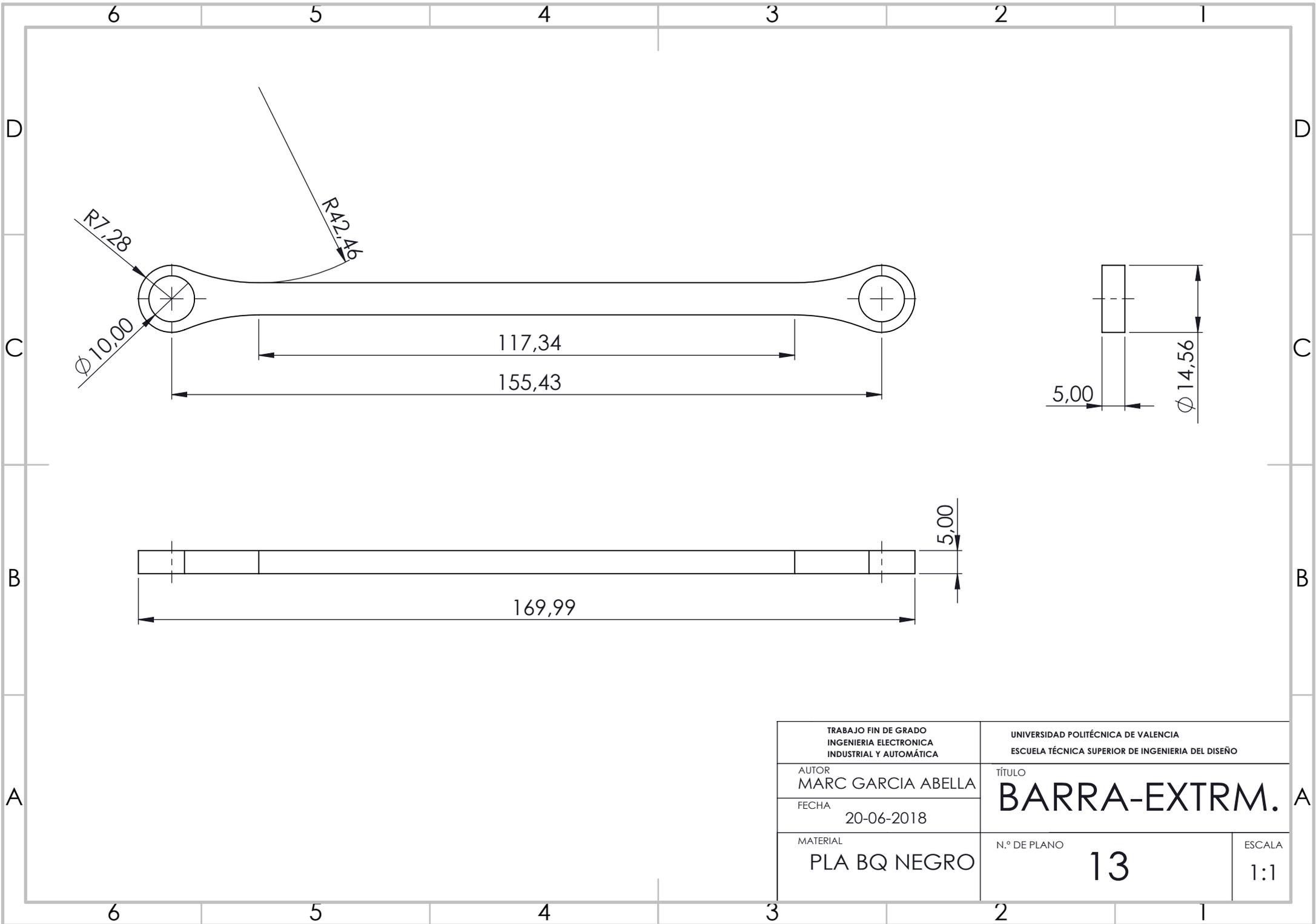
SECCIÓN A-A

TRABAJO FIN DE GRADO INGENIERIA ELECTRONICA INDUSTRIAL Y AUTOMÁTICA	UNIVERSIDAD POLITÉCNICA DE VALENCIA ESCUELA TÉCNICA SUPERIOR DE INGENIERIA DEL DISEÑO	
AUTOR MARC GARCIA ABELLA	TÍTULO <b>CORONA BASE</b>	
FECHA 20-06-2018	N.º DE PLANO <b>11</b>	ESCALA 1:2
MATERIAL PLA BQ NEGRO		

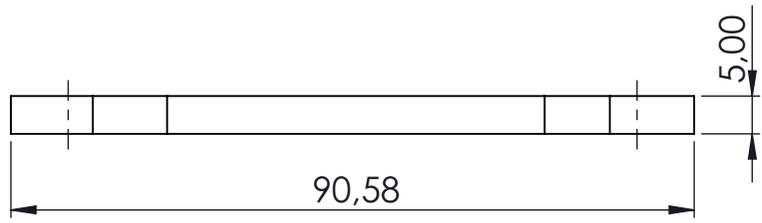
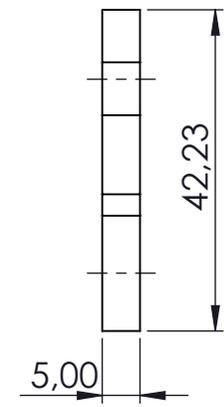
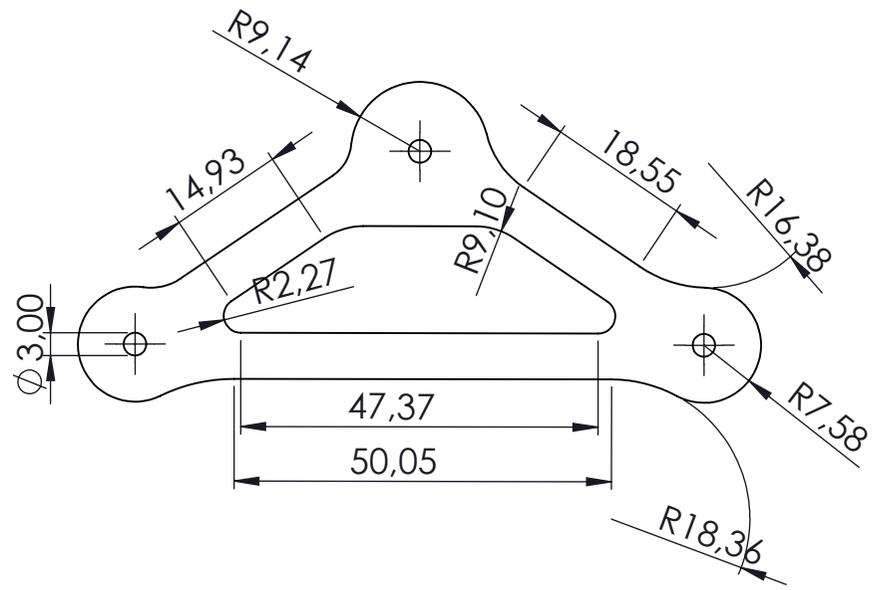


SECCIÓN A-A

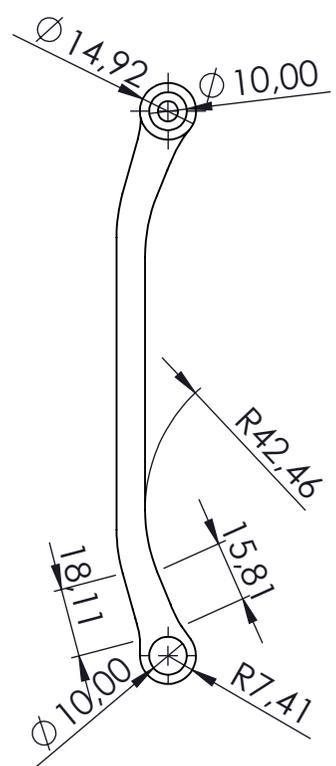
TRABAJO FIN DE GRADO INGENIERIA ELECTRONICA INDUSTRIAL Y AUTOMÁTICA		UNIVERSIDAD POLITÉCNICA DE VALENCIA ESCUELA TÉCNICA SUPERIOR DE INGENIERIA DEL DISEÑO	
AUTOR MARC GARCIA ABELLA		TÍTULO <b>PIÑÓN BASE</b>	
FECHA 20-06-2018			
MATERIAL PLA BQ NEGRO		N.º DE PLANO <b>12</b>	ESCALA 1:1



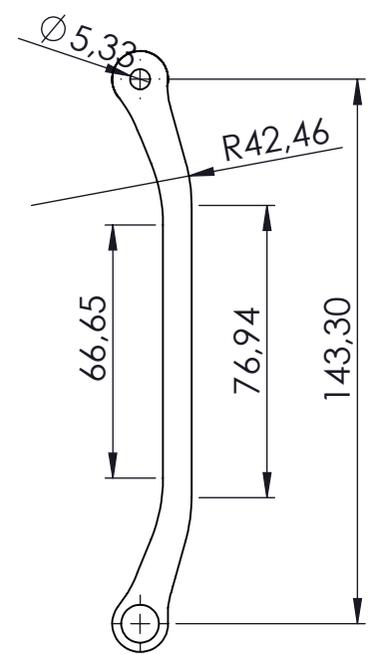
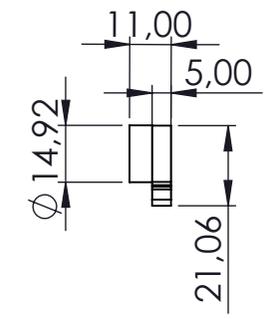
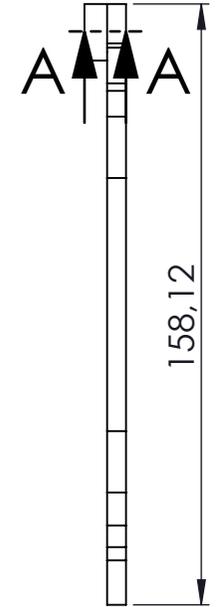
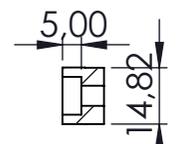
TRABAJO FIN DE GRADO INGENIERIA ELECTRONICA INDUSTRIAL Y AUTOMÁTICA		UNIVERSIDAD POLITÉCNICA DE VALENCIA ESCUELA TÉCNICA SUPERIOR DE INGENIERIA DEL DISEÑO	
AUTOR MARC GARCIA ABELLA		TÍTULO <b>BARRA-EXTRM.</b>	
FECHA 20-06-2018			
MATERIAL PLA BQ NEGRO		N.º DE PLANO <b>13</b>	ESCALA <b>1:1</b>



TRABAJO FIN DE GRADO INGENIERIA ELECTRONICA INDUSTRIAL Y AUTOMÁTICA		UNIVERSIDAD POLITÉCNICA DE VALENCIA ESCUELA TÉCNICA SUPERIOR DE INGENIERIA DEL DISEÑO	
AUTOR MARC GARCIA ABELLA		TÍTULO <b>ENLC-TRIANG.</b>	
FECHA 20-06-2018			
MATERIAL PLA BQ NEGRO		N.º DE PLANO <b>14</b>	ESCALA 1:1

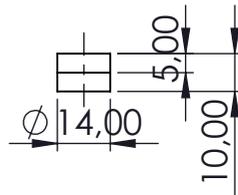
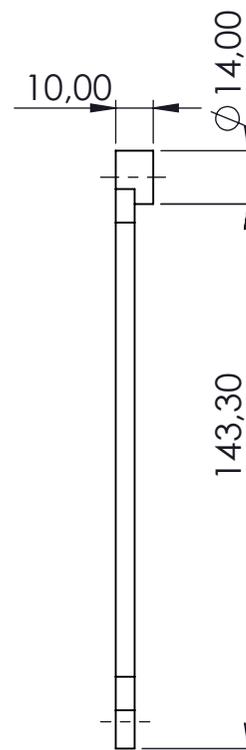
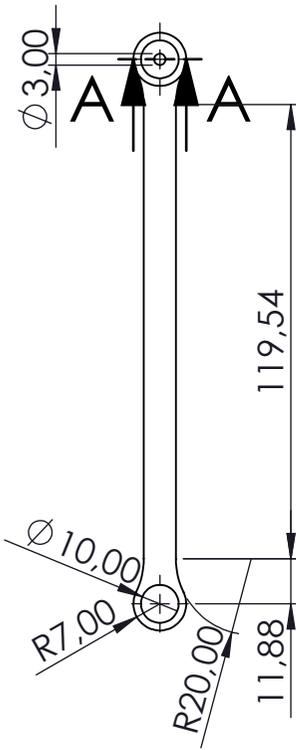
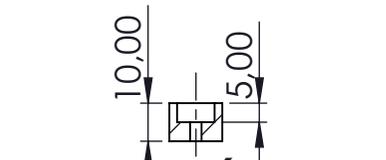


SECCIÓN A-A

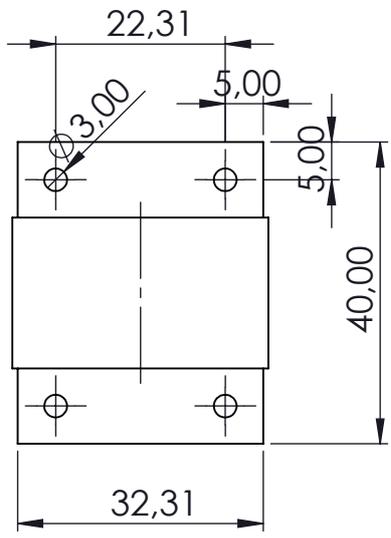
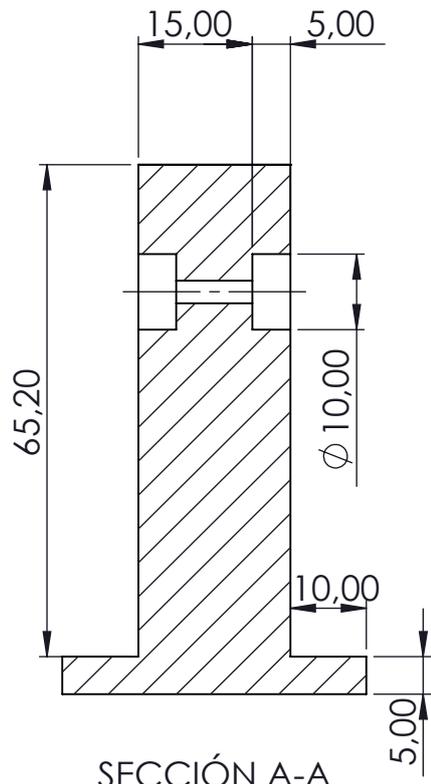
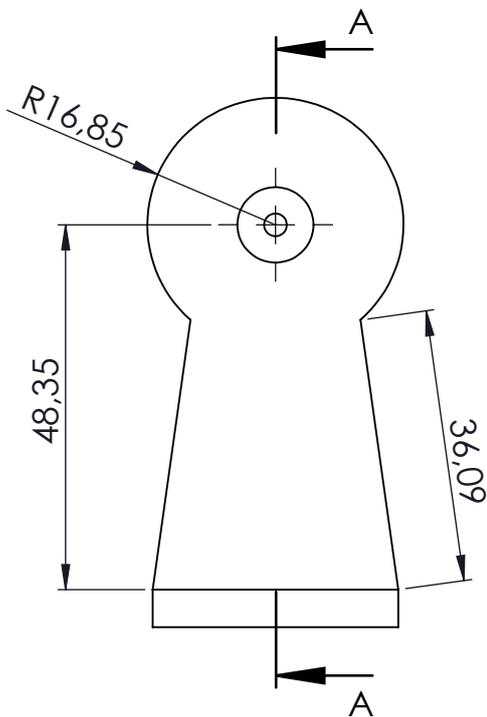


TRABAJO FIN DE GRADO INGENIERIA ELECTRONICA INDUSTRIAL Y AUTOMÁTICA		UNIVERSIDAD POLITÉCNICA DE VALENCIA ESCUELA TÉCNICA SUPERIOR DE INGENIERIA DEL DISEÑO	
AUTOR MARC GARCIA ABELLA		TÍTULO <b>BARRA-BASE</b>	
FECHA 20-06-2018		N.º DE PLANO <b>15</b>	
MATERIAL PLA BQ NEGRO		ESCALA 1:2	

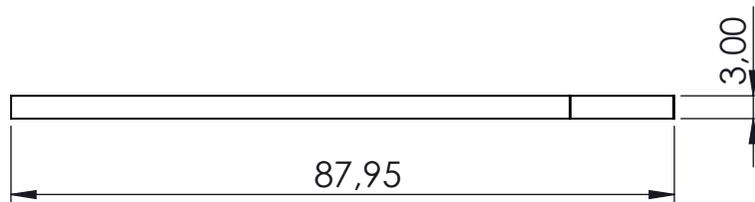
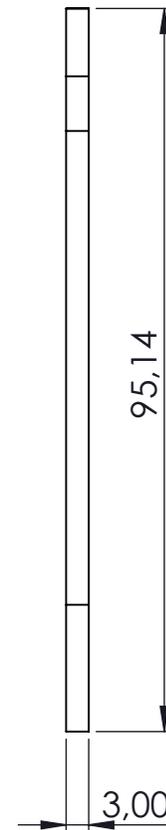
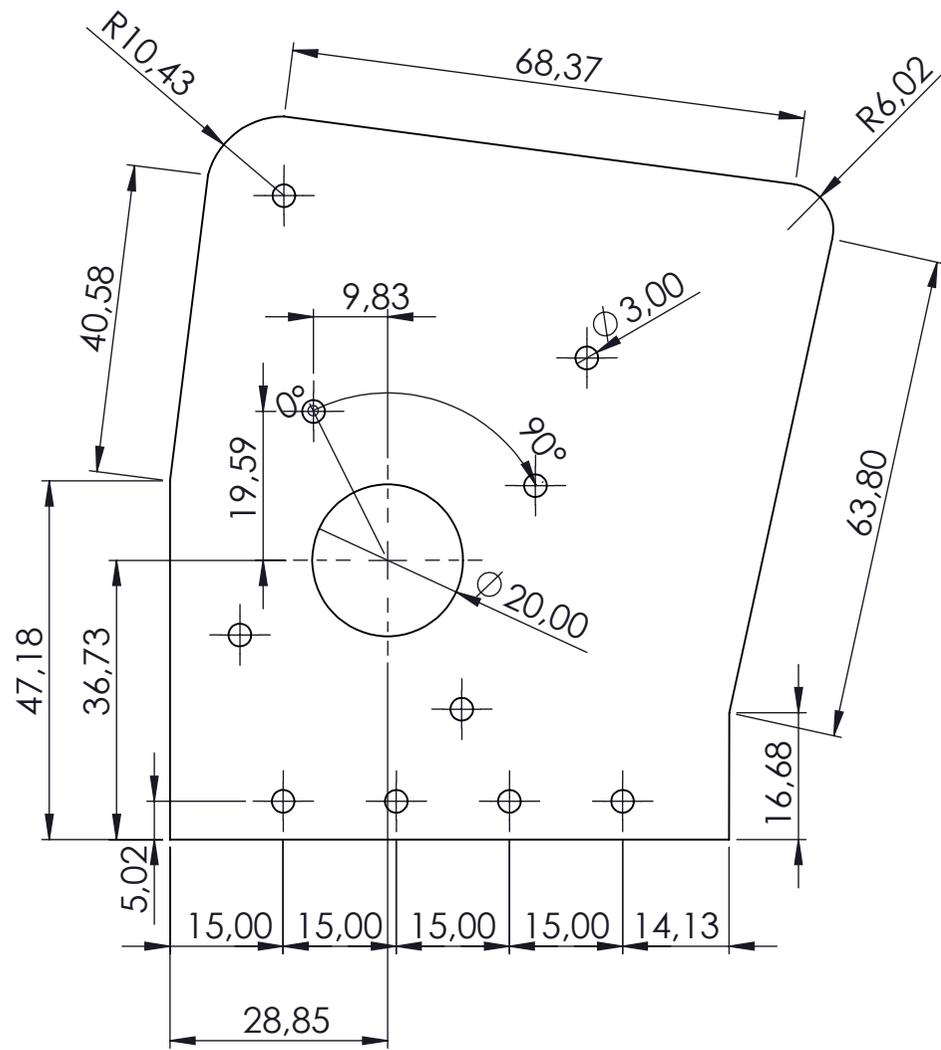
SECCIÓN A-A



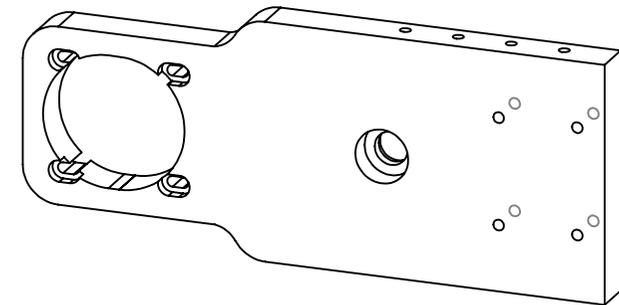
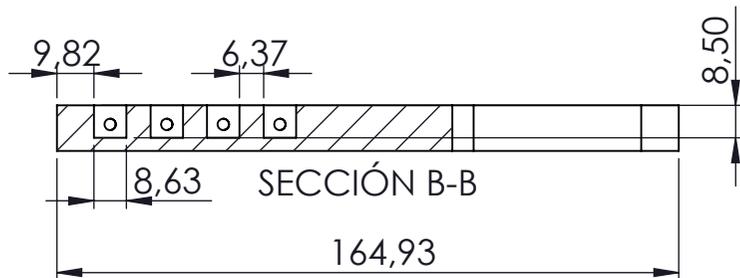
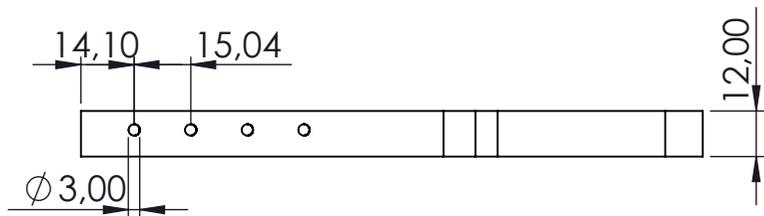
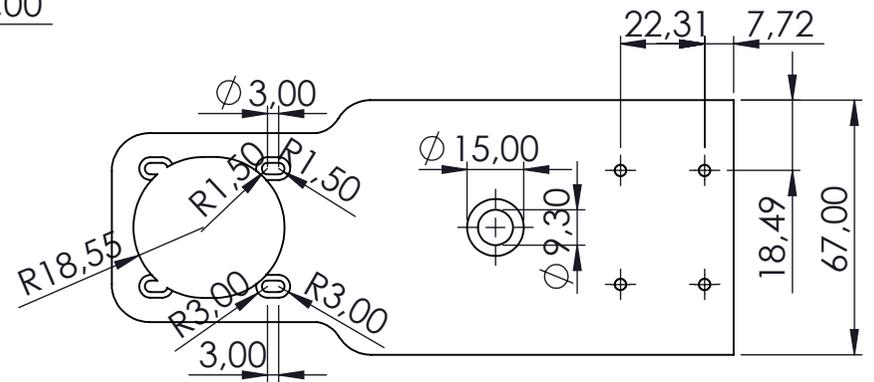
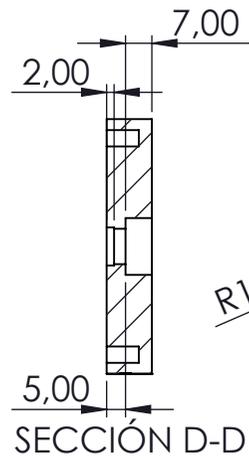
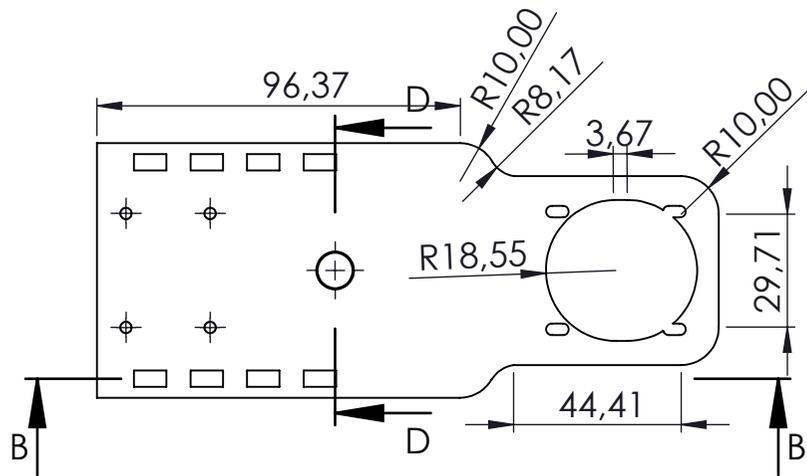
TRABAJO FIN DE GRADO INGENIERIA ELECTRONICA INDUSTRIAL Y AUTOMÁTICA		UNIVERSIDAD POLITÉCNICA DE VALENCIA ESCUELA TÉCNICA SUPERIOR DE INGENIERIA DEL DISEÑO	
AUTOR MARC GARCIA ABELLA		TÍTULO <b>ENLACE M3</b>	
FECHA 20-06-2018			
MATERIAL PLA BQ NEGRO		N.º DE PLANO <b>16</b>	ESCALA 1:2



TRABAJO FIN DE GRADO INGENIERIA ELECTRONICA INDUSTRIAL Y AUTOMÁTICA	UNIVERSIDAD POLITÉCNICA DE VALENCIA ESCUELA TÉCNICA SUPERIOR DE INGENIERIA DEL DISEÑO	
AUTOR MARC GARCIA ABELLA	TÍTULO <b>SOPORTE BRAZO</b>	
FECHA 20-06-2018		
MATERIAL PLA BQ NEGRO	N.º DE PLANO <b>17</b>	ESCALA 1:1



TRABAJO FIN DE GRADO INGENIERIA ELECTRONICA INDUSTRIAL Y AUTOMÁTICA	UNIVERSIDAD POLITÉCNICA DE VALENCIA ESCUELA TÉCNICA SUPERIOR DE INGENIERIA DEL DISEÑO	
AUTOR MARC GARCIA ABELLA	TÍTULO <b>LATERAL BASE</b>	
FECHA 20-06-2018		
MATERIAL PLA BQ NEGRO	N.º DE PLANO <b>18</b>	ESCALA 1:1



TRABAJO FIN DE GRADO INGENIERIA ELECTRONICA INDUSTRIAL Y AUTOMÁTICA		UNIVERSIDAD POLITÉCNICA DE VALENCIA ESCUELA TÉCNICA SUPERIOR DE INGENIERIA DEL DISEÑO	
AUTOR MARC GARCIA ABELLA		TÍTULO <b>BASE</b>	
FECHA 20-06-2018			
MATERIAL PLA BQ NEGRO		N.º DE PLANO <b>19</b>	ESCALA 1:2

# III. PRESUPUESTO Y ESTUDIO ECONÓMICO

---

# Presupuesto y estudio económico detallado

En el presente apartado se describe el estudio económico del desarrollo y prototipado de manera individual y para la producción en masa.

## Desarrollo prototipo

Se dividen los gastos en dos apartados, los que atañen a los honorarios del proyectista y el coste de la ejecución material.

### Honorarios del proyectista

En este punto se estiman las horas que se invierten en el proyecto en las distintas fases. Cada una de las fases requiere una inversión de tiempo y precio distinta. A continuación, en la tabla se muestra desglosado el resultado de los honorarios del proyectista:

Honorarios del proyectista, inversión en personal			
Concepto	Horas	Precio/Hora (€)	Importe
Análisis, estudio y viabilidad del proyecto	35	30	1.050,00 €
Diseño del sistema 2D y 3D	60	30	1.800,00 €
Construcción y montaje	10	10	100,00 €
Diseño e implementación del programa	40	25	1.000,00 €
Fase de pruebas	15	15	225,00 €
		<b>Total</b>	<b>4.175,00 €</b>

### Presupuesto de la ejecución material

En este apartado se tienen en cuenta no solo el material que conforma el prototipo sino todas las herramientas necesarias para la realización del prototipo. Podemos dividir principalmente en costes de desarrollo y costes por unidad de producto.

Costes de desarrollo	
Concepto	Importe
Licencia LabVIEW	800,00 €
Licencia SolidWorks	300,00 €
Licencia AutoCAD	400,00 €
Impresora 3D	400,00 €
Ordenador	300,00 €
<b>Total</b>	<b>2.200,00 €</b>

Costes por unidad de producto		
Concepto	Cantidad	Importe total
Nema 17	3	36,00 €
Potenciómetro 10k	2	3,20 €
Potenciómetro multivuelta	1	3,59 €
Servomotor 9g	1	1,76 €
Pololu DVR8825	3	6,00 €
CNC Shield	1	2,45 €
Arduino UNO	1	4,65 €
Alimentación 12V 1A	1	4,10 €
Cable USB	1	1,20 €
Tornillería M3	20	3,10 €
Tuercas M3	20	2,45 €
Cables	2m	0,45 €
Piezas 3D	20	4,42 €
	<b>Total</b>	<b>73,37 €</b>

## Producción en masa

A continuación, se expone el precio unitario para un volumen de producción de unas 500 piezas. Como la producción es en masa y la compra de material supone un volumen elevado, se estima una reducción del veinte por ciento en el coste material por unidad de producto.

**Coste material total con 20% de descuento: 58,70 €**

Se dividen los costes por los honorarios del proyectista y los costes de desarrollo entre las 500 unidades.

Se establece un beneficio del 18% y el impuesto de valores se establece en 21% por lo que el precio de venta al público queda reflejado de la siguiente forma:

PVP unitario por 500 uds. de producción	
Concepto	Importe
Material	58,70 €
Gastos generales	4,40 €
Honorarios del proyectista	8,35 €
Beneficio	12,86 €
IVA	17,70€
<b>Total</b>	<b>102,01 €</b>

**El precio de venta al público del robot industrial a escala Garra es de 102,01 €**