



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA



UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Escuela Técnica Superior de Ingeniería del Diseño

Diseño, ensamblaje y programación del control cinemático  
de un brazo robótico de tres grados de libertad mediante  
cámaras.

Trabajo Fin de Grado

Grado en Ingeniería Mecánica

AUTOR/A: Leon Gilabert, Andreu

Tutor/a: González Sorribes, Antonio

CURSO ACADÉMICO: 2022/2023



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA



**DISEÑO, ENSAMBLAJE Y PROGRAMACIÓN DEL CONTROL  
CINEMÁTICO DE UN BRAZO ROBÓTICO DE TRES GRADOS DE  
LIBERTAD MEDIANTE CÁMARAS.**

TRABAJO DE FIN DE GRADO

**Autor:**

Andreu León Gilabert

**Supervisor:**

Antonio González Sorribes

Universidad Politécnica de Valencia  
Escuela Técnica Superior de Ingeniería del Diseño

## RESUMEN

El objetivo de este proyecto se desglosa en 2 partes. En primer lugar, el diseño y fabricación de un brazo robótico de 3 grados de libertad, y en segundo lugar, la programación del control cinemático para poder controlar la posición del órgano terminal del brazo robótico mediante el uso de cámaras. La programación se realizará en una plataforma Arduino, desde el cual, mediante la posición de un objeto de interés obtenida por triangulación con las cámaras, se controlarán los actuadores de los motores del robot para poder así situar el órgano terminal del robot en la posición del objeto detectado.

### Palabras clave:

- Robot manipulador
- Control cinemático
- Arduino
- Diseño y fabricación
- Localización espacial

## ÍNDICE

1.-INTRODUCCIÓN.....	12
1.1.- OBJETIVOS.....	13
1.2.- APLICACIONES DEL PROYECTO.....	13
1.3.- MARCO TEÓRICO.....	14
1.3.1.- LOCALIZACIÓN ESPACIAL.....	14
1.3.2.- CINEMÁTICA.....	15
1.3.2.1.- CINEMÁTICA DIRECTA.....	16
1.3.2.2.- CINEMÁTICA INVERSA.....	17
1.3.3.- CONTROL DE LA CÁMARA.....	19
1.3.3.1.- MODELO RGB.....	19
1.3.3.2.- LOCALIZACIÓN DEL OBJETO.....	20
1.3.4.- ARDUINO.....	20
2.- APLICACIONES SOFTWARE UTILIZADAS.....	21
2.1.- APLICACIONES DE DIBUJO Y SIMULACIÓN.....	21
2.1.1.- FUSION 360.....	21
2.1.2.- ROBOT SCENE.....	22
2.1.3.- TINKERCAD.....	22
2.1.4.- PAINT 3D.....	22
2.2.- APLICACIONES DE CÁLCULO.....	23
2.2.1.- MATLAB.....	23
2.3.- HERRAMIENTAS DE PROGRAMACIÓN.....	23
2.3.1.- ARDUINO IDE.....	23
2.3.2.- VISUAL STUDIO CODE.....	24
3.- DESARROLLO DEL MODELO.....	24
3.1.- ELECCIÓN DE LOS GRADOS DE LIBERTAD.....	24

3.2.- DISEÑO DE LAS PARTES DEL BRAZO ROBÓTICO.....	26
3.2.1.- DISEÑO DE LA BASE.....	27
3.2.2.- DISEÑO DEL HOMBRO.....	28
3.2.3.- DISEÑO DEL BRAZO 1.....	29
3.2.4.- DISEÑO DL BRAZO 2.....	30
3.3.- ANÁLISIS DE FUERZAS.....	31
3.3.1.- ANÁLISIS DE RESULTADOS.....	32
3.4.- DISPOSICIÓN DE LA CÁMARA.....	33
3.5.- RELACIÓN ENTRE LOS SISTEMAS DE REFERENCIA.....	35
4.- CONTROL CINEMÁTICO DEL MODELO.....	36
4.1.- POSICION EN REPOSO.....	36
4.2.- CINEMÁTICA DIRECTA.....	37
4.3.- CINEMÁTICA INVERSA.....	39
4.4.- VALIDACIÓN DE LA CINEMÁTICA.....	41
5.- COMPONENTES ELECTRÓNICOS.....	43
5.1.- MOTORES PASO A PASO NEMA 17.....	43
5.2.- PLACA ARDUINO UNO.....	44
5.3.- MODULOS L298n.....	44
5.4.- PLACA DE PRUEBAS.....	45
5.5.- CABLES PUENTE.....	46
5.6.- ALIMENTACIÓN EXTERNA.....	46
5.7.- CONEXIONADO ENTRE LOS COMPONENTES.....	47
6.- PROGRAMACIÓN.....	48
6.1.- ARDUINO.....	48
6.2.- PYTHON.....	51
7.- CONCLUSIONES.....	54
8.- OBJETIVOS DE DESARROLLO SOSTENIBLES.....	55
9.- BIBLIOGRAFÍA.....	56

## **PRESUPUESTO**

1.- MATERIALES.....	58
2.- MANO DE OBRA.....	58
3.- PROCEDIMIENTOS.....	58
4.- ENSAMBLAJE DEL SOPORTE.....	59
5.- ENSAMBLAJE DE LA PARTE SUPERIOR.....	59
6.- ENSAMBLAJE DE LOS BRAZOS.....	60
7.- ENSAMBLAJE DEL SOPORTE DE LA CÁMARA.....	60
8.- COSTE TOTAL.....	61

## **PLIEGO DE CONDICIONES**

1.- OBJETO.....	63
2.- CONDICIONES DE LOS MATERIALES.....	63
2.1.- ENSAMBLAJE DEL SOPORTE.....	63
2.1.1.- BASE.....	63
2.1.2.- HOMBRO.....	63
2.1.3.- VARILLA.....	64
2.1.4.- VARILLA ROSCADA.....	64
2.1.5.- CONTROL DE CALIDAD.....	64
2.2.- ENSAMBLAJE DE LA PARTE SUPERIOR.....	64
2.2.1.- PARTE SUPERIOR.....	65
2.2.2.- CONTROL DE CALIDAD.....	65
2.3.- ENSAMBLAJE DE LOS BRAZOS.....	65
2.3.1.- BRAZO 1.....	65
2.3.2.- BRAZO 2.....	66
2.3.3.- VARILLA ROSCADA.....	66
2.3.4.- CONTROL DE CALIDAD.....	66
2.4.- ENSAMBLAJE DEL SOPORTE DE LA CÁMARA.....	66
2.4.1.- PLANCHA BASE.....	66
2.4.2.- PLANCHAS LATERALES.....	67
2.4.3.- PLANCHA DE SOPORTE.....	67
2.4.5.- CONTROL DE CALIDAD.....	67
3.- CONDICIONES DE EJECUCIÓN.....	68
3.1.- DESCRIPCIÓN.....	68
3.2.- CONTROL DE CALIDAD.....	70
4.- PRUEBA DE SERVICIO.....	71

## PLANOS

1.- ENSAMBLAJE DEL ROBOT.....	73
1.1.- ENSAMBLAJE DEL SOPORTE.....	74
1.1.1.- BASE.....	75
1.1.2.- HOMBRO.....	76
1.2.- ENSAMBLAJE DE LA PARTE SUPERIOR.....	77
1.2.1.- PARTE SUPERIOR.....	78
1.3.- ENSAMBLAJE DE LOS BRAZOS.....	79
1.3.1.- BRAZO 1.....	80
1.3.2.- BRAZO 2.....	81
2.- ENSAMBLAJE DEL SOPORTE DE LA CÁMARA.....	82
2.1.- PLANCHA BASE.....	83
2.2.- PLANCHAS LATERALES.....	84
2.3.- PLANCHA DE SOPORTE.....	85



## **ANEXO MOTORES**

1.- HOJA TÉCNICA MOTOR NEMA 17 17HS08-1004S.....	86
2.- HOJA TÉCNICA MOTOR NEMA 17 17HS15-1504S-X1.....	87

## ANEXO DE FIGURAS

<b>Figura 1:</b> Representación de localización espacial en un brazo robótico.....	13
<b>Figura 2:</b> Vista cenital desde la cámara de la localización de un objeto respecto a la base del robot.....	14
<b>Figura 3:</b> Relación entre la cinemática directa e inversa. ....	14
<b>Figura 4:</b> Matriz de transformación homogénea individual.....	16
<b>Figura 5:</b> Vista cenital de la representación de los eslabones rotacionales del robot en configuración codo arriba y codo abajo.....	17
<b>Figura 6:</b> Tabla de colores RGB.....	19
<b>Figura 7:</b> Robot cartesiano.....	24
<b>Figura 8:</b> Robot cilíndrico.....	24
<b>Figura 9:</b> Robot polar.....	24
<b>Figura 10:</b> Robot angular.....	24
<b>Figura 11:</b> Boceto del robot.....	25
<b>Figura 12:</b> Diseño inicial de la base.....	26
<b>Figura 13:</b> Diseño de los anclajes de la base.....	26
<b>Figura 14:</b> Diseño final de la base.....	26
<b>Figura 15:</b> Vista en sección del conjunto de la base y el motor.....	26
<b>Figura 16:</b> Primer diseño del hombro.....	27
<b>Figura 17:</b> Diseño final del hombro.....	27
<b>Figura 18:</b> Parte superior de la estructura del robot.....	28
<b>Figura 19:</b> Vista isométrica superior del brazo 1.....	28
<b>Figura 20:</b> Vista isométrica inferior del brazo 1.....	29
<b>Figura 21:</b> Diseño del brazo 2.....	29
<b>Figura 22:</b> Restricciones fijas para elaborar el estudio.....	30
<b>Figura 23:</b> Distribución de las cargas externas.....	31
<b>Figura 24:</b> Resultados obtenidos del análisis de tensiones.....	31
<b>Figura 25:</b> Resultados obtenidos del análisis de deformaciones.....	32
<b>Figura 26:</b> Bocete del conjunto robot-soporte para la cámara.....	33

<b>Figura 27:</b> Boceto de los sistemas de referencia del robot y la cámara.....	34
<b>Figura 28:</b> Situación de los ejes del robot.....	36
<b>Figura 29:</b> Obtención del parámetro $r$ .....	38
<b>Figura 30:</b> Obtención del parámetro $q_2$ .....	39
<b>Figura 31:</b> Modelado de la base en RobotScene.....	40
<b>Figura 32:</b> Modelado del eslabón 1 en RobotScene.....	40
<b>Figura 33:</b> Modelado del eslabón 2 en RobotScene.....	40
<b>Figura 34:</b> Modelado del eslabón 3 en RobotScene.....	40
<b>Figura 35:</b> Ensamblaje del robot en RobotScene.....	41
<b>Figura 36:</b> Tabla de control del movimiento articular.....	41
<b>Figura 37:</b> Motor NEMA 17.....	42
<b>Figura 38:</b> Placa Arduino UNO.....	43
<b>Figura 39:</b> Modulo L298n.....	44
<b>Figura 40:</b> Placa de pruebas.....	44
<b>Figura 41:</b> Cables macho-macho.....	45
<b>Figura 42:</b> Cables macho-hembra.....	45
<b>Figura 43:</b> Fuente de alimentación externa.....	45
<b>Figura 44:</b> Conexionado entre el motor y el módulo l298n.....	47
<b>Figura 45:</b> Conexionado de todos los elementos.....	48

## ANEXO DE TABLAS

<b>Tabla 1:</b> Ejemplo de la tabla D-H.....	15
<b>Tabla 2:</b> Resultados de la cinemática inversa.....	17
<b>Tabla 3:</b> Resultados de la calibración de la altura de la cámara.....	32
<b>Tabla 4:</b> Valores articulares en la posición de reposo.....	35
<b>Tabla 5:</b> Parámetros Denavit-Hartenberg para el caso particular.....	37
<b>Tabla 6:</b> Resolución de la cinemática inversa.....	38
<b>Tabla 7:</b> Objetivos de desarrollo sostenibles.....	55
<b>Tabla 8:</b> Materiales utilizados.....	58
<b>Tabla 9:</b> Mano de obra.....	58
<b>Tabla 10:</b> Presupuesto del ensamblaje del soporte.....	59
<b>Tabla 11:</b> Presupuesto del ensamblaje de la parte superior.....	59
<b>Tabla 12:</b> Presupuesto del ensamblaje de los brazos.....	60
<b>Tabla 13:</b> Presupuesto del ensamblaje del soporte de la cámara.....	60
<b>Tabla 14:</b> Presupuesto del ensamblaje del proyecto.....	61
<b>Tabla 15:</b> Comprobaciones previas a las pruebas de servicio.....	71

## ANEXO DE CÓDIGOS

<b>Código 1:</b> Librerías de Arduino.....	48
<b>Código 2:</b> Declaración de las variables de Arduino.....	49
<b>Código 3:</b> Definición de los motores paso a paso.....	49
<b>Código 4:</b> Primera parte del Setup.....	49
<b>Código 5:</b> Segunda parte del Setup.....	50
<b>Código 6:</b> Primera parte del loop.....	50
<b>Código 7:</b> Segunda parte del loop.....	50
<b>Código 8:</b> Librerías de Python.....	51
<b>Código 9:</b> Declaración de las variables de Python.....	51
<b>Código 10:</b> Primera función.....	52
<b>Código 11:</b> Segunda función.....	52
<b>Código 12:</b> Conexión y puesta en marcha de la cámara.....	52
<b>Código 13:</b> Localización de los puntos mediante los colores definidos.....	52
<b>Código 14:</b> Obtención de la altura del objeto y configuración del robot.....	53
<b>Código 15:</b> Conexión con Arduino.....	53
<b>Código 16:</b> Cierre del programa.....	53

## 1.- INTRODUCCION

Durante el desarrollo de este proyecto se pretende llevar a cabo la elaboración de un prototipo de robot comercial que sea capaz de realizar sus movimientos de manera autónoma con la ayuda de una cámara que va a indicarle las posiciones en las cuales se encuentran los objetos de interés para el robot.

La principal motivación de realizar este proyecto se basa en conseguir, gracias a los conocimientos aportados en las diferentes asignaturas del grado como pueda ser, robótica o diseño asistido por ordenador, además de combinar la programación y electrónica, un brazo robótico totalmente funcional y listo para ser puesto en marcha en cadenas de producción convencionales debido al gran uso de estas máquinas en la actualidad.

Los orígenes de la robótica se remontan a los años 20, donde el escritor checo Karel Capek hizo uso por primera vez de la palabra *robot*, que traducido significa trabajo forzado, pero no fue hasta años más tarde, en 1948, cuando se empezó a utilizar el término robótica por el inventor estadounidense George Devol, creador del primer robot industrial con la finalidad de crear un aparato automático capaz de poder mover, transporta objetos o realizar tareas que para un humano resultarían casi imposibles, debido a las grandes cantidades de peso que debería mover y/o condiciones o escenarios en los que debería trabajar.[1]

Actualmente, según la Real Academia Española, conocemos como robótica el “conjunto de técnicas que aplican la informática y emplean aparatos que, en sustitución de personas, realizan trabajos por lo general en la industria.”

Tanto el avance de este campo, como de las nuevas tecnologías, han permitido un desarrollo de robots cada vez más complejos, con más aplicaciones, y de softwares ligados a ellos que permiten tanto la simulación de tareas o escenarios, como la programación de estos, permitiendo así, una implementación del campo de la robótica en la industria cada vez mayor. Además, gracias a este avance, se ha conseguido que los robots no formen parte únicamente de las industrias, sino que también se están desarrollando en otros sectores, como podría ser el doméstico, o el sector de la medicina.

### **1.1.- OBJETIVOS**

El objeto principal del presente proyecto consiste en conseguir elaborar un brazo robótico de 3 grados de libertad (o 3 articulaciones) mediante impresión 3D que gracias al uso de una cámara sea capaz de identificar la posición de un objeto en un espacio de trabajo para que, posteriormente, con la implementación de un microcontrolador Arduino, pueda situar su órgano terminal de manera automática sobre dicho objeto de interés. Para ello, se deberán de seguir los siguientes puntos:

- Diseño y modelado del robot y su espacio de trabajo.
- Adquisición de los componentes necesarios para el ensamblaje del robot, ya sea obtenidos de terceros o elaborados de mano propia.
- Montaje del robot según el diseño realizado.
- Montaje de la cámara según el diseño realizado.
- Programación, tanto del robot, siguiendo el modelo cinemático, como de la cámara.
- Comprobación del correcto funcionamiento según lo esperado.

### **1.2.- APLICACIONES DEL PROYECTO**

Debido a que la finalidad del trabajo se basa en la elaboración de un robot autónomo gracias a la cámara, este proyecto se presenta con la intención de elaborar una maqueta o base de otros proyectos de mayor estudio y complejidad sobre monitorización de robots, además de ofrecer un prototipo para aplicaciones o tareas más sencillas.

Los robots monitorizados con cámaras presentan una utilidad mayor respecto a los convencionales, esto se debe a que esos robots son capaces de trabajar en espacios con obstáculos, o se pueden emplear, a su vez, para tareas con difícil accesibilidad para un humano. Además, estos robots están mejor preparados para realizar trabajos conjuntos junto a otros, ya que, gracias al uso de las cámaras, son capaces de identificar cuando han de empezar a realizar las tareas que se les haya asignado sin tener que esperar a recibir órdenes de otro robot.

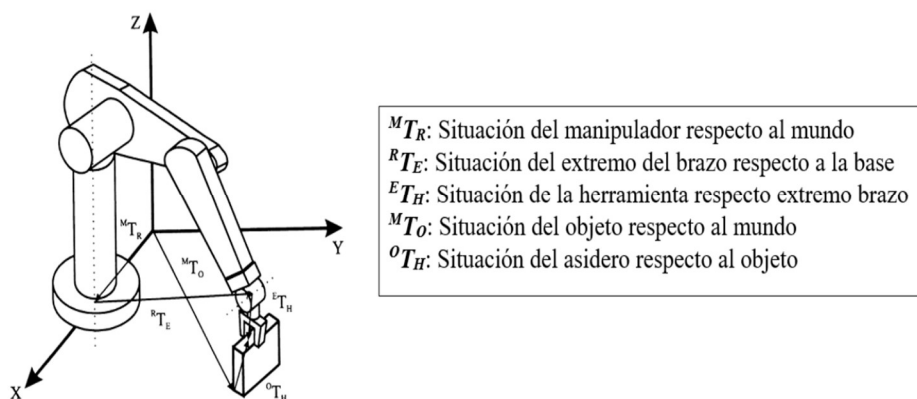
Por otra parte, se pretenderá diseñar un robot similar a los de la familia SCARA, este estará diseñado con la intención de que se pueda cambiar su órgano terminal dependiendo de las tareas que vaya a realizar, dándole al robot una mayor flexibilidad ante las circunstancias que se le puedan presentar.

### 1.3 MARCO TEÓRICO

Para poder llevar a cabo este proyecto es imprescindible aclarar en que se basa tanto, la cinemática del robot, que va a permitir a las diferentes articulaciones moverse correctamente como la localización espacial del robot respecto al objeto de interés mediante el uso de la cámara, y el funcionamiento de esta última para localizar dicho objeto. Es por ello que el objetivo de este punto es la explicación teórica detrás de los componentes del proyecto.

#### 1.3.1.- LOCALIZACIÓN ESPACIAL

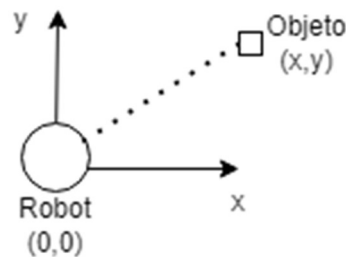
Debido a que el principal objetivo es conseguir que el órgano terminal del robot llegue a una posición indicada, en primer lugar, es necesario conocer la posición y orientación de dicho órgano terminal respecto un sistema de referencia, que en este caso va a ser la base del robot, como se puede apreciar en la **Figura 1**. Para ello, se deberán conocer las posiciones de los diferentes eslabones del robot, que, gracias a los cálculos matriciales y a la geometría del robot, se puede obtener de manera sencilla.



**Figura 1:** Representación de localización espacial en un brazo robótico.



Una vez conseguido localizar el órgano terminal del robot respecto su base como referencia, mediante la cámara, el siguiente paso será identificar el objeto con el que queremos trabajar, y situarlo respecto al mismo sistema de referencia que el órgano terminal como se muestra en la **Figura 2**, con la finalidad de que el robot no haya de hacer ninguna transformación en las coordenadas para determinar la posición del objeto respecto su referencia base.

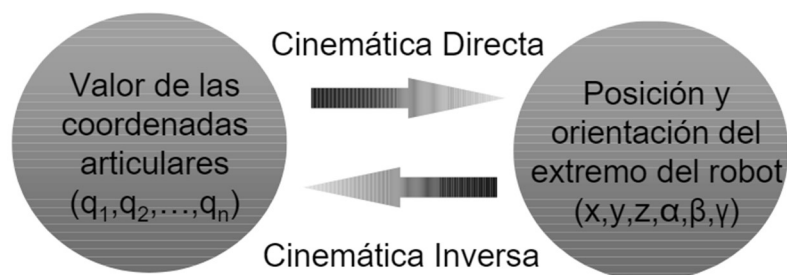


**Figura 2:** Vista cenital desde la cámara de la localización de un objeto respecto a la base del robot.

### 1.3.2.- CINEMÁTICA

La cinemática de los robots es fundamental a la hora de realizar un diseño y programación de un robot, ya que es la base de la obtención de las trayectorias de movimiento de las diferentes articulaciones de los robots.

Entendemos como cinemática de un robot el estudio y análisis del movimiento de sus diferentes componentes sin tener en cuenta ni velocidades ni fuerzas que se producen en estos movimientos [2]. Dentro de la cinemática del robot, existen 2 tipos principales, cinemática directa e inversa, que se explicaran en el apartado 1.3.2.1 y 1.3.2.2 del presente documento y que se relacionan como podemos observar en la **Figura 3**.



**Figura 3:** Relación entre la cinemática directa e inversa.

### 1.3.2.1.- CINEMÁTICA DIRECTA

El control de la cinemática directa de un robot se basa en, conociendo la geometría del robot, así como los valores de las articulaciones que van tomando en un instante de tiempo, determinar tanto la orientación como la posición del órgano terminal del robot.

Para conseguir determinar estos valores, tanto de geometría como de articulaciones, se empleará el algoritmo Denavit-Hartenberg (D-H). Gracias a este algoritmo, mediante una tabla de 5 columnas y tantas filas como articulaciones tenga el robot, se podrán obtener los sistemas de referencia correspondientes a cada eslabón del robot definiéndose cada uno a partir del sistema de coordenadas del eslabón anterior. Para el paso de un eslabón al siguiente se realizará mediante las siguientes transformaciones:

- Rotación en un ángulo  $\Phi$  sobre el eje  $Z_{i-1}$
- Translación de una distancia  $d_i$  sobre el eje  $Z_{i-1}$
- Translación de una distancia  $a_i$  sobre el eje  $X_i$
- Rotación en un ángulo  $\alpha$  sobre el eje  $X_i$

Siendo  $i$  el eslabón sobre el que estamos trabajando y todos los ángulos girados siguiendo la regla de la mano derecha.[3]

Quedando tras estas transformaciones una tabla como la siguiente para un robot de 3 grados de libertad:

	$\Phi$ (°)	$d$ (mm)	$a$ (mm)	$\alpha$ (°)
<b>0-1</b>	$\Phi_1$	$d_1$	$a_1$	$\alpha_1$
<b>1-2</b>	$\Phi_2$	$d_2$	$a_2$	$\alpha_2$
<b>2-3</b>	$\Phi_3$	$d_3$	$a_3$	$\alpha_3$

**Tabla 1:** Ejemplo de la tabla D-H.

Para realizar este algoritmo correctamente, se ha de seguir la regla que se conforma por los siguientes pasos:

1. Numerar los eslabones, siendo el eslabón 0 la base del robot.
2. Numerar las articulaciones siguiendo los grados de libertad que tenga el robot.
3. Localizar los ejes en las diferentes articulaciones.
4. Identificar los ejes  $Z_i$  que corresponden, en caso de articulaciones rotacionales, al eje de giro de la articulación, y en caso de articulaciones prismáticas, al eje de desplazamiento de la articulación.
5. Situar el sistema de coordenadas 0, respecto al cual irán los demás referenciados.

6. Situar el resto de los sistemas, colocando el origen de cada uno de estos sistemas en el punto de intersección entre  $Z_i$  y  $Z_{i+1}$  en caso de ser perpendiculares, o en cualquier punto de la articulación en caso de ser paralelos
7. Identificar los ejes  $X_i$  en dirección perpendicular a  $Z_i$  y  $Z_{i-1}$ , en dirección de  $Z_{i-1}$  a  $Z_i$ .
8. Identificar los ejes  $Y_i$  que vienen determinados por la posición de los otros 2 ejes y la regla de la mano derecha.
9. Determinar la tabla D-H según se ha explicado anteriormente en este punto.
10. Determinar, mediante la tabla D-H las matrices de transformación homogéneas individuales para cada eslabón, de la siguiente manera:

$$A_i = \left( \begin{array}{ccc|c} \cos \theta_i & -\cos \alpha_i \sin \theta_i & \sin \alpha_i \sin \theta_i & a_i \cos \theta_i \\ \sin \theta_i & \cos \alpha_i \cos \theta_i & -\sin \alpha_i \cos \theta_i & a_i \sin \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i & d_i \\ \hline 0 & 0 & 0 & 1 \end{array} \right)$$

**Figura 4:** Matriz de transformación homogénea individual.

11. Determinar la matriz de transformación homogénea total a partir de la multiplicación de las matrices individuales siguiendo el orden desde la base hasta el órgano terminal.

De esta forma, la cinemática directa del robot queda totalmente resuelta. [4]

### 1.3.2.2.- CINEMÁTICA INVERSA

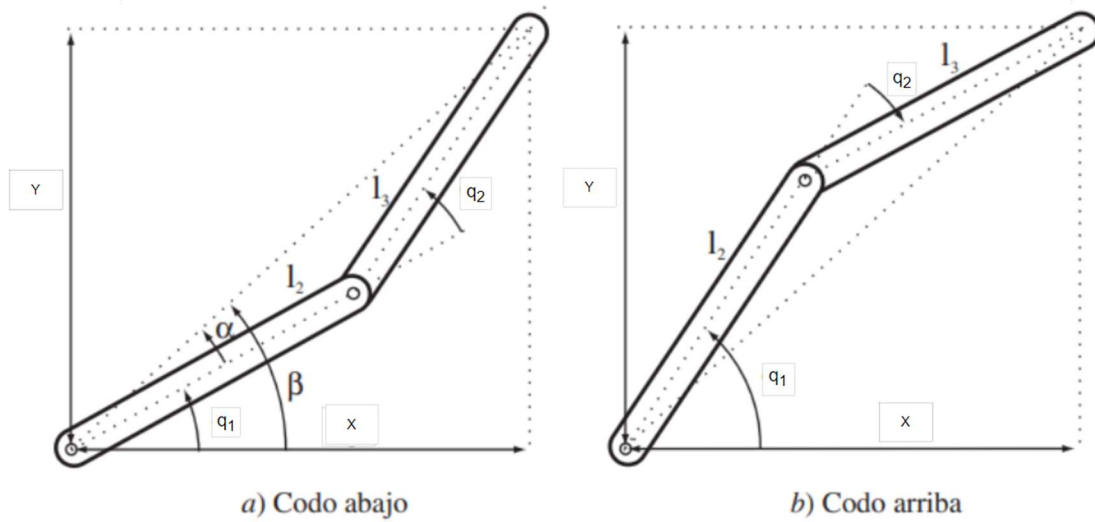
La cinemática inversa consiste en determinar dicho valor de movimiento para cada articulación con el fin de lograr situar el órgano terminal en una posición y orientación ya conocidas previamente. A diferencia de la cinemática directa, en esta no existe un algoritmo concreto para resolver el problema, además de existir múltiples soluciones para una misma posición final, debido a las configuraciones que puede optar el robot (codo arriba, codo abajo, frente, espaldas, “flip”, “no flip”).

Este método presenta a su vez el problema de las singularidades, que viene a darse cuando un determinado movimiento o configuración elimina uno o más grados de libertad como, por ejemplo, alineando dos ejes consecutivos, creando indeterminaciones en la resolución del problema volviéndolo un caso imposible de resolver, es por ello que estas zonas son de prioridad a la hora de realizar el estudio.

Del mismo modo, la resolución se vuelve más compleja ya que la geometría del robot juega un papel más importante que en el caso anterior, por ello, los robots más complejos (6 grados de libertad), son los más tediosos de resolver.

Debido a que la geometría del robot de este proyecto es relativamente sencilla, al presentar únicamente 3 grados de libertad, el método por el cual se resolverá la cinemática inversa será el método analítico con enfoque algebraico, donde los valores articulares se obtendrán directamente despejando del modelo cinemático directo.

En la **Figura 5** se muestra un ejemplo de resolución de un robot con tres articulaciones, una prismática la cual asignaremos como nombre “q3”, y dos articulaciones rotacionales, “q1” y “q2” respectivamente, el modelado de la resolución algebraica quedaría de la siguiente manera:



**Figura 5:** Vista cenital de la representación de los eslabones rotacionales del robot en configuración codo arriba y codo abajo.

Aplicando calculo trigonométrico, el resultado obtenido sería el siguiente:

q1 (configuración codo arriba)	$Arctan\left(\frac{y}{x}\right) + Arccos\left(\frac{l_2^2 + r^2 - l_3^2}{2 * l_2 * r}\right)$
q1 (configuración codo abajo)	$Arctan\left(\frac{y}{x}\right) - Arccos\left(\frac{l_2^2 + r^2 - l_3^2}{2 * l_2 * r}\right)$
q2 (configuración codo arriba)	$Arccos\left(\frac{l_2^2 - r^2 + l_3^2}{2 * l_2 * l_3}\right) - \pi$
q2 (configuración codo abajo)	$\pi - Arccos\left(\frac{l_2^2 - r^2 + l_3^2}{2 * l_2 * l_3}\right)$
q3	$l_1 - z$

**Tabla 2:** Resultados de la cinemática inversa.

Siendo  $r$  la línea que une el origen con el extremo del segundo eslabón del robot de la **Figura 5** y  $l_1$  la longitud del eslabón prismático. [5]

### 1.3.3.- CONTROL DE LA CÁMARA

Realizar el control sobre una cámara para que detecte objetos es una tarea bastante compleja, ya que existen muchos modos diferentes. Para este proyecto se elaborará un programa en lenguaje *Python* mediante el cual se detectarán las coordenadas de un objeto de interés, se referenciarán al mismo sistema de referencia del robot, y se enviarán automáticamente a este, para que sea capaz de alcanzar dicho objeto con su órgano terminal.

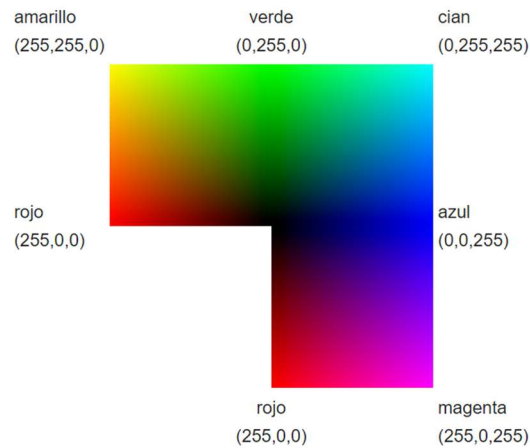
Los sistemas más comunes para realizar localización de objetos mediante el uso de cámaras son los siguientes:

- Enfoque manual, donde el usuario identifica el objeto y el programa solo ha de localizarlo.
- Localización mediante IA, donde el usuario introduce los datos del objeto que quiere identificar, por ejemplo, su forma geometría, y el programa lo localiza automáticamente.
- Reconocimiento de formas, donde el programa es capaz de identificar las formas en una imagen y posteriormente las localiza en el plano
- Reconocimiento de colores, donde el programa identifica los colores en una imagen y los sitúa en el plano.

Para este proyecto, debido a la disposición de las librerías *OpenCV2* y *NumPy* en *Python* que permite detectar colores en imágenes y realizar cálculos numéricos complejos, la opción elegida es reconocimiento por colores.

#### 1.3.3.1.- MODELO RGB

El modelo RGB es el modelo que se va a usar para detectar los colores en la imagen que tomará la cámara para poder localizar el objeto de interés. Las siglas de dicho modelo hacen referencia a *red*, *green* y *blue*, que en español significa rojo, verde y azul, y nos permite obtener cualquier color mediante la combinación de estos tres, por ejemplo, en el modelo RGB, si se quisiera representar el color azul, la combinación sería (0,0,255), ya que en esta combinación la presencia de color rojo y verde es nula, dando el máximo valor al color azul. En la **Figura 6** se puede observar una imagen con los valores correspondientes a diferentes colores. [6]



**Figura 6:** Tabla de colores RGB.

### 1.3.3.2.- LOCALIZACIÓN DEL OBJETO

Una vez localizado el objeto mediante el método de colores, se deberá localizar en el plano y referenciarlo al sistema de coordenadas del robot.

En primer lugar, se deberá calibrar la cámara, para poder determinar cuántos centímetros tanto en vertical como en horizontal es capaz de capturar dependiendo en la posición que la situemos. Una vez determinadas estas longitudes, se establecerá la relación cm/píxel conociendo los píxeles de la imagen, ya que el valor que obtendrá el programa será en píxeles, y mediante la relación anterior obtendremos el valor de las coordenadas en centímetros.

### 1.3.4.- ARDUINO

Arduino es una plataforma de *software* libre que permite crear y desarrollar proyectos electrónicos por medio de una placa con microcontrolador programable y ofreciendo un entorno de desarrollo integrado (IDE) que facilita la carga de programas en dicha placa.

La base de Arduino se encuentra en el microcontrolador que es el responsable de ejecutar las acciones programadas previamente para poder controlar otros aparatos electrónicos, puedan ser actuadores, sensores o otros componentes asociados a este mediante conexiones en los diferentes pines de la placa. Además, presenta un entorno cómodo para la programación ya que utiliza un lenguaje basado en C y C++. [7]

Dicho entorno presenta, a su vez, una amplia recopilación de librerías como pueda ser “stepper.h”, que permite el control de motores paso a paso, o “math.h” que permite al programa realizar cálculos matemáticos complejos, y es gracias a estas librerías que Arduino presenta una gran versatilidad a la hora de realizar proyectos con ella.

Gracias a la continua investigación en esta tecnología se han conseguido desarrollar gran cantidad de placas con microcontroladores que han permitido volver esta plataforma en una herramienta todavía más útil, a continuación, se muestra una lista con las placas Arduino más usadas:

- Arduino UNO
- Arduino Nano
- Arduino Mega
- Arduino Leonardo
- Arduino Due

## **2.- APLICACIONES SOFTWARE UTILIZADAS**

Para poder realizar este trabajo se han utilizado los programas que se van a explicar en este apartado. En esta lista de programas se incluyen tanto las herramientas de simulación, como de dibujo y cálculo que se han empleado.

### **2.1.- APLICACIONES DE DIBUJO Y SIMULACIÓN**

#### **2.1.1.- FUSION 360**

Fusion 360 es una plataforma de Autodesk de *software* CAD, que permite realizar tanto planos bidimensionales como modelado en 3D de piezas y ensamblajes. Es la herramienta principal de diseño que se va a emplear en este trabajo, ya que no solamente permite el modelado y ensamblaje de las diferentes partes del robot, sino que también es capaz de realizar otras funciones como análisis estáticos de fuerzas, y animaciones.

Además, esta aplicación es capaz de transformar los modelos realizados en esta a otros formatos, que nos permitirá posteriormente crear los archivos necesarios para la correcta impresión 3D de las piezas.

### **2.1.2.- ROBOT SCENE**

RobotScene es una aplicación de *software* que permite tanto el modelado de los eslabones de un robot, como el posterior ensamblaje de ellos es un escenario de trabajo, siguiendo la tabla de Denavit-Hartenberg explicada en el punto 1.3.2.1 de este mismo documento.

Esta aplicación es capaz de simular tanto el modelo cinemático directo, como el modelo cinemático inverso gracias a su propio entorno de programación basado en el lenguaje C.

Durante este trabajo, tras el modelado de las piezas en Fusion 360, y la resolución del modelo cinemático del robot, se utilizará este *software* para mostrar una simulación del funcionamiento del robot con la finalidad de comprobar que el análisis sea correcto.

### **2.1.3.- TINKERCAD Y FRITZING**

TinkerCAD y Fritzing son unas aplicaciones de diseño, principalmente de prototipos 3D y montajes electrónicos que permite además la programación de dichos montajes en un lenguaje basado en C.

En este trabajo, las herramientas de TinkerCAD y Fritzing se utilizarán conjuntamente sobre todo para la elaboración y simulación de los esquemas electrónicos del brazo robótico para comprobar su correcto funcionamiento antes de elaborar el montaje final. Además, gracias al entorno de programación que ofrecen, permitirá a su vez la simulación del programa del microcontrolador Arduino.

### **2.1.4.- PAINT 3D**

Paint 3D es una aplicación desarrollada por Microsoft que permite tanto la edición de imágenes como el dibujo en 2D y 3D.

Gracias a esta aplicación se han desarrollado, a lo largo de este trabajo, numerosos esquemas, diagramas y dibujos que se han utilizado posteriormente en esta memoria para aclarar ciertas explicaciones, además de la edición de ciertas imágenes.

Por otra parte, se empleó también para la comprobación del código de programación de la cámara, ya que, al funcionar a partir de detección de colores nos permite simular dibujos para corroborar el correcto funcionamiento de la cámara.



## **2.2.- APLICACIONES DE CÁLCULO**

### **2.2.1.- MATLAB**

MatLab, nombre abreviado de “MATriz LABoratory” es un programa de cálculo numérico que ofrece un entorno de programación con lenguaje propio (lenguaje M) el cual permite el desarrollo de algoritmos y comandos de cálculo numérico y análisis de datos. Esta aplicación es muy usada en estudios de robótica ya que permite realizar cálculo matricial complejo que es la base de la cinemática de control sobre la mayoría de los robots.

Como se ha explicado en el punto 1.3.2.1 de esta memoria, para realizar el análisis de la cinemática directa del prototipo, será necesario calcular las matrices de transformación homogéneas para cada eslabón del robot, además, al contar con la cámara, la cual añade otro sistema de referencia ajeno al principal del brazo robótico, para elaborar una correcta programación posterior sobre los motores de accionamiento de las articulaciones, el cálculo matricial será esencial en este proyecto.

## **2.3.- HERRAMIENTAS DE PROGRAMACIÓN**

### **2.3.1.- ARDUINO IDE**

Arduino IDE es una aplicación que permite tanto la escritura como la carga de programas en microcontroladores Arduino. El lenguaje que utiliza este *software* está basado en C y C++, y además de incluir las bibliotecas que incluyen estos lenguajes, se pueden encontrar bibliotecas específicas para el control de sensores y actuadores.

El movimiento de los eslabones del prototipo del robot se pretende diseñar utilizando motores NEMA 17 paso a paso, ya que son los más comunes en este tipo de proyectos, es por eso que la programación se realizara gracias a esta aplicación, mediante el uso de bibliotecas como “stepper.h”, que permiten establecer un control sobre dichos motores.

### 2.3.2.- VISUAL STUDIO CODE

Visual Studio Code es un editor de código fuente creado por Microsoft. Se caracteriza porque es una aplicación de código abierto y gratuita, con un entorno de programación muy completo, y, además, ofrece una amplia compatibilidad con diferentes lenguajes de programación, pueda ser; C, C++, Java, *Python*...

Esta aplicación se empleará para la programación de la cámara, ya que como se comentó en el apartado 1.3.3 de este documento, *Python* ha sido el lenguaje elegido para elaborar el programa de dicha cámara por ofrecer bibliotecas tanto para la detección de colores, como para el cálculo de coordenadas, además de presentar la opción de poder transmitir datos a puertos de salida del ordenador, siendo esto muy útil para enviar la información obtenida al microcontrolador Arduino y así automatizar el conjunto.

## 3.- DESARROLLO DEL MODELO

En este apartado se va a explicar cómo se llegó al desarrollo del robot partiendo de la elección, tanto de sus grados de libertad, como del tipo de articulaciones, exponiendo los diferentes problemas que se plantearon en el diseño hasta llegar al prototipo final.

Cabe aclarar que tanto el ensamblaje de los diferentes componentes como las medidas de ellos estarán reflejados en el documento **Planos**.

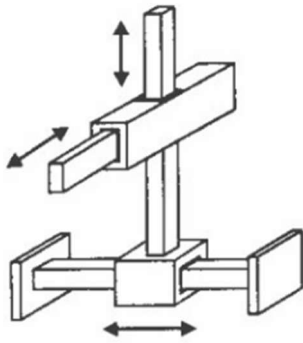
### 3.1.- ELECCIÓN DE LOS GRADOS DE LIBERTAD

Dentro del diseño del robot, el número de articulaciones que este presenta es el paso más importante de decidir, ya que las articulaciones son las que van a permitir al robot el movimiento de los diferentes eslabones para realizar su tarea. En la industria se pueden encontrar desde robots de 2 grados de libertad, que permiten el movimiento en un único plano, hasta 6 grados de libertad, que son los más complejos, pero permiten al robot el libre movimiento y orientación dentro de su espacio de trabajo tridimensional.

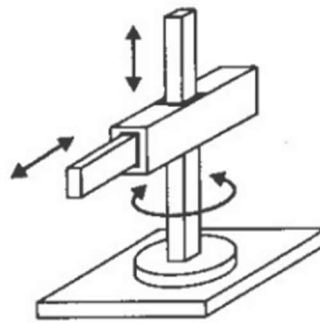
Debido a que la finalidad de este proyecto principalmente es la comunicación automática entre la cámara y el robot con el motivo de que el robot sea capaz de trabajar de manera autónoma y de forma no repetitiva, el diseño del brazo robótico se plantea lo menos complejo posible pero que sea capaz de trabajar en un espacio tridimensional, es por ello que se propone un diseño inicial de 3 articulaciones.

Una vez aclarado el número de grados de libertad que va a presentar el prototipo, el siguiente paso es determinar cómo van a ser dichas articulaciones (rotacionales o prismáticas). Dentro de los robots de 3 grados de libertad, encontramos los siguientes tipos:

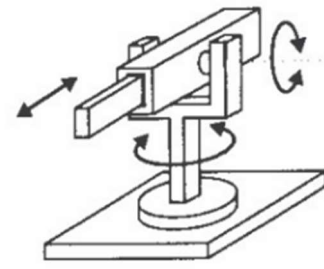
- Cartesianos (3 articulaciones prismáticas) (**Figura 7**)
- Cilíndrico (1 articulación rotacional y 2 prismáticas) (**Figura 8**)
- Polar (1 articulación prismática y 2 rotacionales) (**Figura 9**)
- Angulares (3 articulaciones rotacionales) (**Figura 10**)



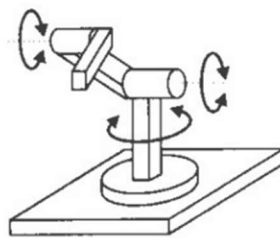
**Figura 7:** Robot cartesiano.



**Figura 8:** Robot cilíndrico.



**Figura 9:** Robot polar.



**Figura 10:** Robot angular.

El robot cartesiano, como se aprecia en la **Figura 7** es un robot que necesita más espacio que los demás en cuanto a su posicionamiento en un área de trabajo, además de ser el tipo de robot menos usado en la industria actual, por lo que este modelo de brazo robótico queda descartado para el desarrollo de este proyecto.

En cuanto al robot angular, debido a que sus eslabones deben ser más flexibles para poder realizar giros en todos los sentidos, como se puede observar en la **Figura 10**, el sistema de transmisión que ha de llevar es de mayor complejidad, y como se ha comentado anteriormente, el objeto de este proyecto no se presenta en la elaboración de un robot complejo

En cuanto a las dos opciones restantes, debido a que el prototipo se pretende ensamblar con el funcionamiento de motores paso a paso de rotación continua, la opción elegida será el robot polar, ya que presenta más articulaciones rotacionales que el cilíndrico, lo que facilitaría el ensamblaje de este.

Por tanto, como prototipo final se elige elaborar el diseño de un brazo robótico de 3 grados de libertad tipo polar.

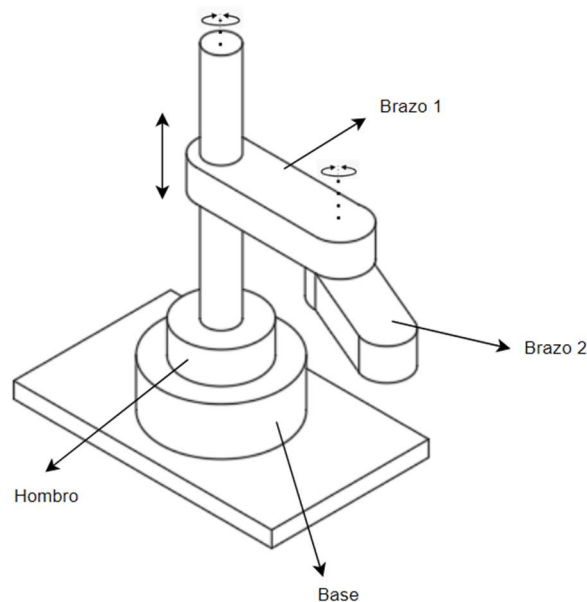
### 3.2.- DISEÑO DE LAS PARTES DEL BRAZO ROBÓTICO

Aclarado tanto los grados de libertad que dispondrá el robot, como el tipo de robot que se diseñara, siguiendo el símil robot-humano, se dividirán las partes del robot a estudiar de la siguiente manera:

- **Hombro:** Se llamará hombro al primer eslabón del brazo robótico, ya que esta servirá como apoyo para los siguientes eslabones además de ser el primer eslabón articular.
- **Brazo 1:** Se identificará como brazo 1 el primer eslabón del robot conectado al hombro.
- **Brazo 2:** Se identificará como brazo 2 el eslabón del robot conectado al brazo 1.
- **Codo:** El codo del robot representará la unión articulada entre los dos brazos anteriores.
- **Muñeca:** La muñeca representara el órgano terminal del robot, unido al brazo 2. Este será el encargado de situarse sobre los objetos

Además, todos ellos estarán sobre una base fija, que permitirá al robot el movimiento sobre ella sin tener que preocuparse de problemas de estabilidad.

Debido a que la elección ha sido de tres grados de libertad, se asociará el primer grado de libertad al hombro (rotacional), el segundo grado de libertad al codo (rotacional), y un tercer grado de libertad que permitirá al robot el desplazamiento vertical (prismático), a su vez, por simplicidad en el diseño, ambos brazos irán unidos a esta última articulación para evitar el desarrollo de transmisiones complejas, quedando un boceto del robot como se aprecia en la **Figura 11**.



**Figura 11:** Boceto del robot.

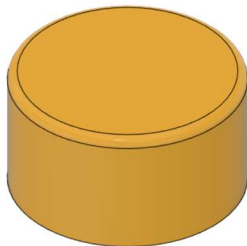
### 3.2.1.- DISEÑO DE LA BASE

La base es la parte del robot encargada de soportar todo el resto de la estructura, es por ello que debe de ser firme y segura. Además, hay que tener en cuenta que el robot va a estar en movimiento mientras esté realizando su tarea, por ello, la base ha de ser, a su vez, fija y resistente.

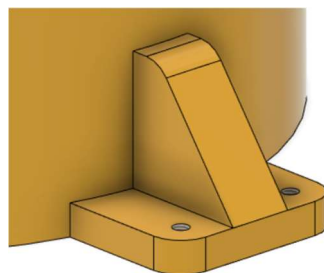
En un primer diseño, se modela la base de forma cilíndrica, al igual que se había supuesto en el boceto inicial, quedando como se muestra en la **Figura 12**.

Hecho el diseño inicial, para que la base quede fija al espacio de trabajo, se añaden 3 anclajes con nervio con una disposición triangular sobre ella (**Figura 13**), con agujeros roscados para que pueda ser atornillada, de esta manera la base queda anclada y estable.

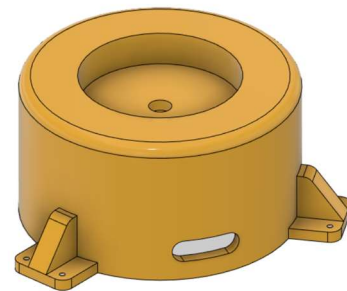
Además, para reducir el uso de espacio de la estructura del robot, se plantea añadir el motor que va a hacer girar la primera articulación dentro de la base ya que, según el diseño planteado, el hombro del robot va directamente unido a esta parte, es por ello que se modela un agujero de forma cilíndrica como se aprecia en la **Figura 14** en la parte superior de esta para poder encajar el hombro en ella por medio de un rodamiento que le permita girar, unido al motor, independiente a la base.



**Figura 12:** Diseño inicial de la base.

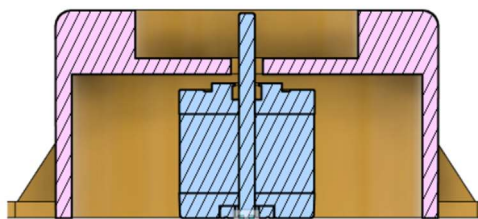


**Figura 13:** Diseño de los anclajes de la base.



**Figura 14:** Diseño final de la base.

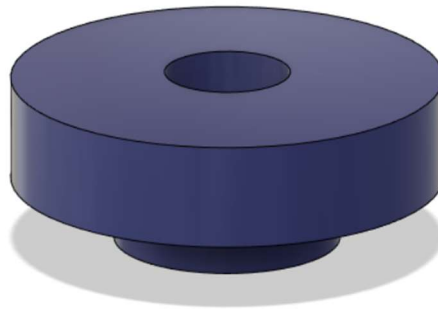
Vista la base desde un plano de sección con la disposición del motor en su interior queda reflejada en la **Figura 15**, donde se observa que el motor sobresale de la base para poder ser fijado al hombro.



**Figura 15:** Vista en sección del conjunto de la base y el motor.

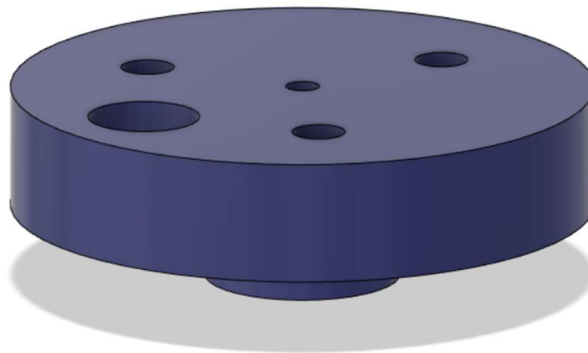
### 3.2.2.- DISEÑO DEL HOMBRO

El hombro pretende seguir el modelo del boceto de la **Figura 11**, es por ello que tomara una forma cilíndrica, al igual que la base, pero este añadirá un saliente que le permita encajar con la base, además, el sistema de movimiento prismático del robot va a recaer sobre el hombro, por ello, se diseñara una parte central hueca para encajar un husillo que será el responsable de este movimiento, quedando como se muestra en la **Figura 16**.



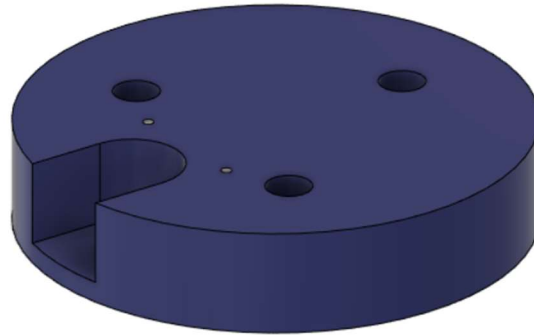
**Figura 16:** Primer diseño del hombro.

Quedando el diseño inestable para el conjunto del robot, se remodela dejando el agujero del husillo hacia un lado, y colocando en forma de triángulo agujeros para soportar varillas de acero que permitan distribuir mejor la carga sobre el hombro, quedando la geometría final como se observa en la **Figura 17**.



**Figura 17:** Diseño final del hombro.

Además, para reforzar la estabilidad de la estructura, se añade a la parte superior de las varillas una pieza similar al hombro que hará la función de fijar las varillas y soportar el motor que hará girar el husillo permitiendo el movimiento vertical de los brazos del robot como se muestra en la **Figura 18**.



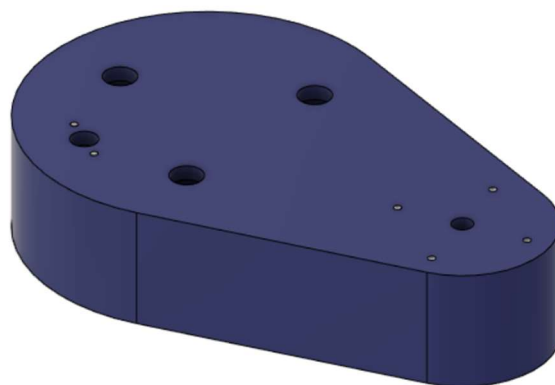
**Figura 18:** Parte superior de la estructura del robot.

### 3.2.3.- DISEÑO DEL BRAZO 1

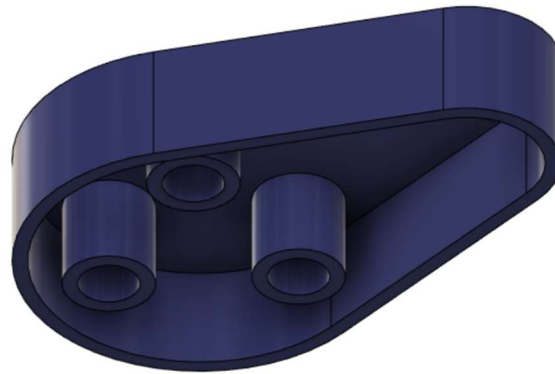
El primer brazo del robot debe ir ensamblado en las varillas que se encuentran entre la parte superior y hombro del robot, es por eso que la geometría en la unión debe ser idéntica a estas partes. Además, como este brazo va a ser el encargado del movimiento vertical de la estructura, se diseñará con unos salientes para colocar rodamientos en las partes que haya contacto entre el brazo y las varillas para facilitar el movimiento.

Como se planteó en el boceto inicial, en la parte final del brazo se incluirá la articulación del codo, por ello, se tendrá esto en cuenta para diseñar una parte donde poder situar el motor que va a permitir esta rotación.

En las **Figuras 19** y **Figura 20** se puede ver la geometría final del brazo 1 del robot en vista isométrica.



**Figura 19:** Vista isométrica superior del brazo 1.

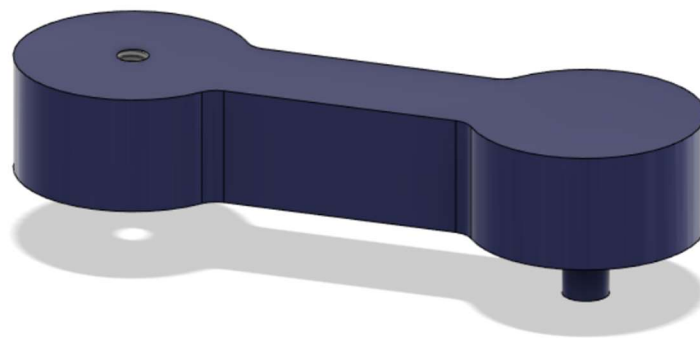


**Figura 20:** Vista isométrica inferior del brazo 1.

### 3.2.4.- DISEÑO DEL BRAZO 2

Teniendo en cuenta que el brazo 2 es el que va a disponer del órgano terminar del robot, cuya función va a ser situarse sobre el objeto de interés, se diseñará con un saliente en el cual se añadirá posteriormente una almohadilla que permita un contacto seguro entre el robot y el objeto.

Además, este eslabón irá unido al brazo anterior mediante el motor que ejerce la función de codo de la estructura, por ello se modelará con una rosca que permita el encaje entre el acople del motor y este brazo, quedando como se puede observar en la **Figura 21**.



**Figura 21:** Diseño del brazo 2.



### 3.3.- ANÁLISIS DE FUERZAS

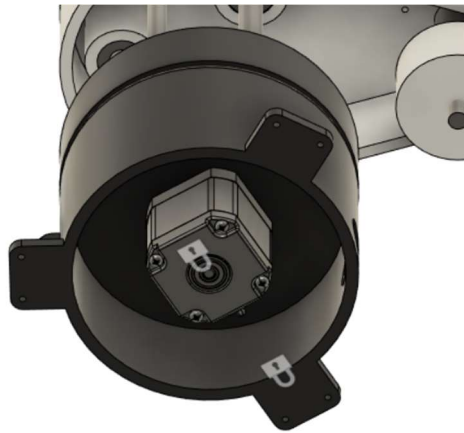
Gracias a las herramientas que ofrece Fusion 360, una vez realizado el ensamblaje del robot, se comprueba su comportamiento estático para, en caso de ser necesario, realizar los cambios en la geometría para evitar que la estructura deforme, o en el peor caso, se desmonte.

Para realizar el correcto estudio, hay que tener en cuenta los siguientes puntos:

- Materiales de los objetos a estudiar.
- Restricciones de movimiento.
- Cargas externas.

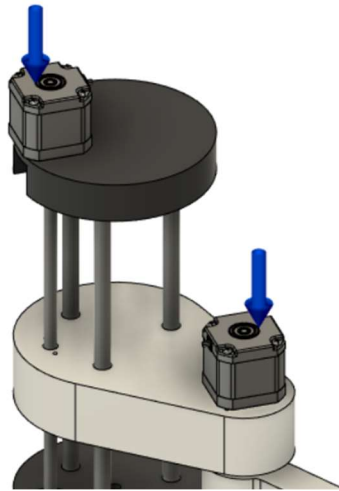
Debido a que las piezas están diseñadas para ser impresas en 3D, el material asignado para estas es el plástico ABS, comúnmente utilizado en estos dispositivos. Por otro lado, las varillas y rodamientos se definirán de acero. Añadiendo el material correspondiente a cada objeto, el programa Fusion 360 es capaz de determinar el peso mediante el volumen que ocupa cada objeto, y la densidad que presenta el material.

Las restricciones de movimiento se verán reflejadas en la base como se puede observar en la **Figura 22**. Como se ha explicado en el apartado 3.2.1 de este mismo documento, la base esta modelada con 3 anclajes para poderla fijar al suelo, además, el motor que se sitúa dentro de ella también estará fijado al suelo para evitar vibraciones.



**Figura 22:** Restricciones fijas para elaborar el estudio.

Las cargas externas vendrán dadas por el peso de los motores paso a paso colocados sobre la estructura. Atendiendo a las especificaciones del fabricante, los motores NEMA 17 presentan un peso en gramos de 240, equivalente aproximadamente a 2.4 Newtons, pero en el estudio se elegirá una carga superior, de 3 Newtons, para aumentar la seguridad en la estabilidad de la estructura. A continuación, se muestra una disposición de las cargas externas sobre el robot (**Figura 23**):

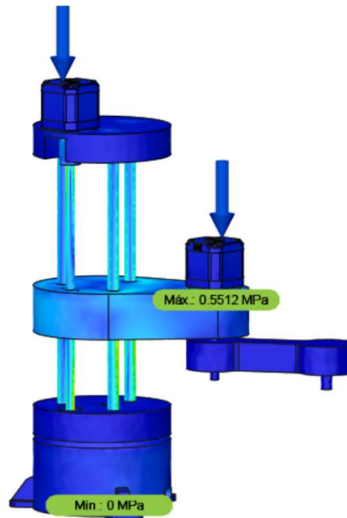


**Figura 23:** Distribución de las cargas externas.

### 3.3.1.- ANÁLISIS DE RESULTADOS

Realizado el análisis de fuerzas, los resultados obtenidos son los siguientes:

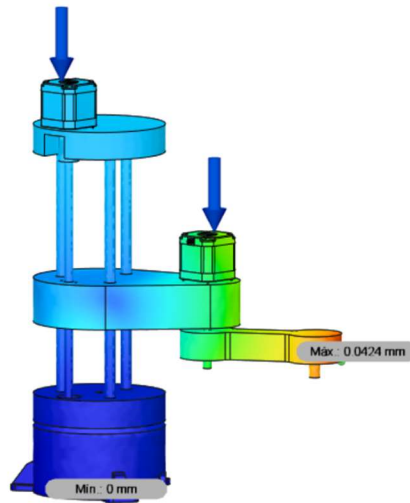
- Análisis de tensiones:



**Figura 24:** Resultados obtenidos del análisis de tensiones.

Como se puede apreciar en la **Figura 24**, la tensión máxima generada en la estructura es de 0.5 MPa, valor insignificante, por lo tanto, se puede comprobar gracias a este análisis que la estructura del robot permanecerá fija y sin quebrarse.

- Análisis de deformaciones:



**Figura 25:** Resultados obtenidos del análisis de deformaciones.

De los resultados obtenidos del análisis de deformaciones, como se observa en la **Figura 25**, se aprecia que la deformación máxima de la estructura se produce al borde del brazo 2, debido a que es el punto más lejano a un apoyo fijo, pero este valor es de 0.0424 milímetros, inapreciable para la vista humana.

Tras estudiar los resultados obtenidos por ambos análisis, se puede determinar que la estructura del robot es firme y segura en el diseño realizado.

### 3.4.- DISPOSICIÓN DE LA CÁMARA

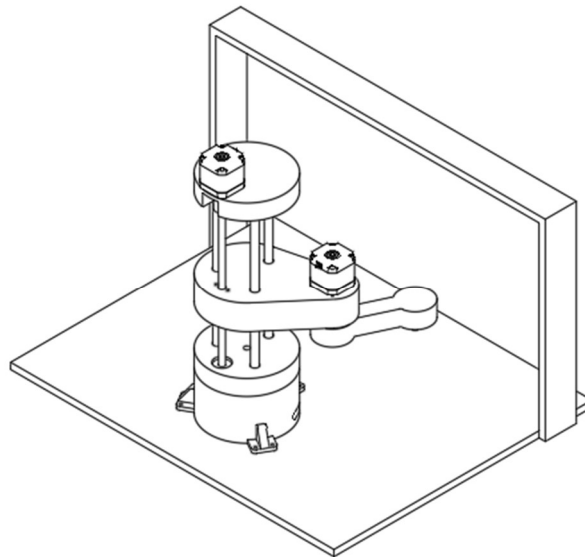
La disposición de la cámara en el espacio de trabajo del robot es un punto de estudio importante, ya que va a ser la encargada de ordenar al robot que posición debe adoptar para conseguir situarse sobre el objeto de interés.

Para determinar la altura de la cámara, se realizará una calibración de esta, tomando imágenes a varias alturas y comprobando el espacio de trabajo que es capaz de detectar a cada una de estas alturas. Los resultados de esta calibración se reflejan en la **Tabla 3**. Además, cabe tener en cuenta que las imágenes se tomarán con una resolución de 1920x1080 píxeles.

Altura (cm)	Longitud detectada en X (cm)	Longitud detectada en Y (cm)
20	32	18
30	50	29
40	63	34

**Tabla 3:** Resultados de la calibración de la altura de la cámara.

Conocido que la longitud total de ambos brazos del robot unidos es de 225 milímetros, ya la calibración realizada, se elige una altura para la cámara de 30 centímetros, ya que el espacio de trabajo capturado por esta a dicha altura no es excesivo para las medidas del robot, ni demasiado pequeño. Además, se colocará el robot a una distancia de 230 milímetros de esta para evitar golpes entre la estructura del robot y la cámara, quedando el conjunto como se muestra en la **Figura 26**, donde se puede apreciar el brazo robótico ensamblado junto a la estructura de soporte de la cámara.



**Figura 26:** Boceto del conjunto robot-soporte para la cámara.

Elegida la disposición de la cámara, para la correcta comunicación entre dicha cámara y el robot, se establecerá una relación a partir de la calibración, de centímetros por píxel quedando de la siguiente manera.

Relación en el eje X

$$\frac{50 \text{ cm}}{1920 \text{ px}} = 0.026 \text{ cm/px}$$

Relación en el eje Y

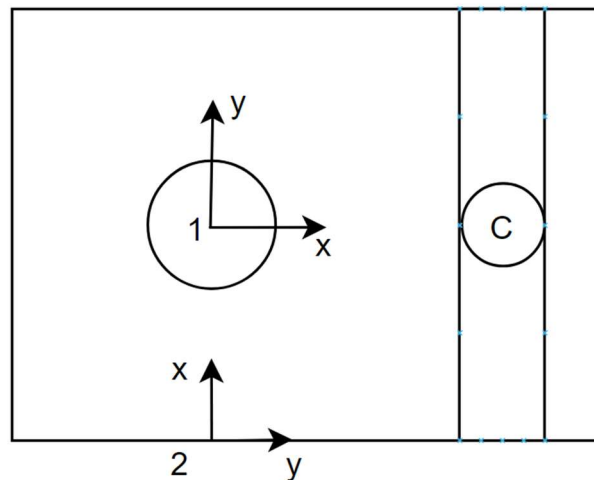
$$\frac{29 \text{ cm}}{1080 \text{ px}} = 0.026 \text{ cm/px}$$

Gracias a estas relaciones, mediante la programación explicada en el apartado anterior, la cámara enviara al robot las coordenadas en centímetros a partir de los píxeles obtenidos.

### 3.5.- RELACIÓN ENTRE LOS SISTEMAS DE REFERENCIA

Diseñado el montaje del conjunto del robot y la cámara, cabe identificar los sistemas de referencia de los elementos para conseguir de esta manera una correcta comunicación entre las coordenadas que la cámara va a detectar, y a las coordenadas que debe moverse el órgano terminal del robot.

Atendiendo a la explicación realizada en el punto 1.3.2.1 de este mismo documento, para un correcto estudio, el robot ha de tener su sistema de referencia en la base, mientras que la cámara determina su propio sistema de referencia en la esquina superior izquierda de cada foto que toma, quedando ambos sistemas de referencia de la siguiente forma:



**Figura 27:** Boceto de los sistemas de referencia del robot y la cámara.

Siendo 1 el sistema de referencia del robot, 2 el sistema de referencia de la cámara y C la disposición de la cámara en el soporte.

Mediante el programa de *Python* se referenciará el sistema de coordenadas de la cámara al robot, cambiando sus ejes X e Y entre ellos, ya que se posicionará el robot en la línea del eje X inicial de la cámara, y estableciendo el 0 en Y en la misma posición que le corresponde al eje del robot teniendo en cuenta que la longitud transversal del soporte de la cámara es de 54 cm.

De esta forma, si la cámara detecta un objeto en el punto (58,100), en cm, el robot recibirá la orden de accionar sus articulaciones hasta situar el órgano terminal en el punto (100,4) en cm.

## 4.- CONTROL CINEMÁTICO DEL MODELO

En el apartado 1.3.2 de este documento se desarrolla toda la teoría que hay que utilizar para conseguir un correcto control cinemático sobre un robot convencional. En este apartado se van a utilizar los conceptos explicados en dicho punto para, matemáticamente resolver el modelo cinemático que sigue el diseño que se ha elaborado.

Además de comprobar su correcta resolución por medio de las aplicaciones MatLab y RobotScene, comentadas en el apartado 2 de esta memoria.

### 4.1.- POSICIÓN DE REPOSO

La posición de reposo de un robot es la posición arbitraria que este adopta cuando no está ejecutando ninguna tarea.

Es importante definir esta posición ya que los cálculos de cinemática van a ir a partir de ella, es por ello que también es recomendable situar los eslabones del robot sobre los ejes de referencia. Teniendo en cuenta el boceto de la **Figura 27**, la posición óptima para definir como reposo sería con ambos brazos extendidos sobre el eje de referencia de la base X, pero debido a la disposición de la cámara, el robot quedaría en medio del espacio de trabajo que la cámara ha de fotografiar, por tanto, se opta por definir una posición de reposo donde el robot permanezca con los brazos extendidos pero fuera del espacio capturado por la cámara, siendo esta en dirección contraria al sentido positivo del eje Y de referencia, llevándolo a esa posición realizando un giro de 90 grados en sentido antihorario sobre el eje Z.

En cuanto a la altura que deberán estar los brazos, al tratarse la segunda articulación de tener carácter prismático, se toma un valor arbitrario de 200 milímetros, ya que es una posición segura para que no haya interferencias en el sistema

En la siguiente tabla se muestran los valores articulares que habría de adoptar el robot para llegar a dicha posición de reposo, siendo  $q_1$  el valor de la articulación rotacional del hombro,  $q_2$  el valor de la articulación prismática, y  $q_3$  el valor de la articulación rotacional del codo

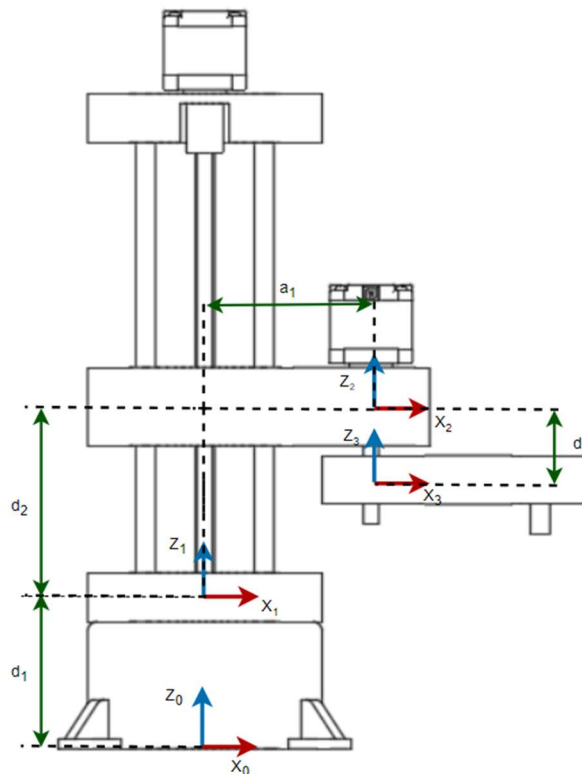
$q_1$ (°)	$q_2$ (mm)	$q_3$ (°)
-90	200	0

**Tabla 4:** Valores articulares en la posición de reposo.

#### 4.2.- CINEMÁTICA DIRECTA

Resuelta la posición de reposo, y siguiendo los pasos para resolver la cinemática directa del modelo que se exponen en el apartado 1.3.2.1, en primer lugar, se identifica cada uno de los eslabones del robot y se ordenan numéricamente, siendo el eslabón 0 la base, el eslabón 1 el hombro, el eslabón 2 el brazo 1, y el eslabón 3 el brazo 2.

En segundo lugar, se identifica cada una de las articulaciones, y se colocan los ejes tal y como se ha explicado en el marco teórico en cada una de las articulaciones, siendo  $Z_i$  y  $X_i$  los ejes pertenecientes a la articulación  $i$ . En la **Figura 28** se puede observar la localización de los ejes de cada articulación sobre el robot.



**Figura 28:** Situación de los ejes del robot.

Siendo:

- $d_1$  la distancia entre los ejes de la base y los ejes de la primera articulación en dirección  $Z$ , tomando un valor de 70 milímetros
- $d_2$  la distancia entre los ejes de la primera articulación y la segunda en dirección  $Z$ . Esta posición, como se explica anteriormente será. Para el caso del estudio se fija un valor de referencia de 200 milímetros
- $a_1$  la distancia entre los ejes de la primera articulación y la segunda en dirección  $X$ , tomando un valor de 100 milímetros
- $d_3$  es la distancia entre los ejes de la segunda articulación y los ejes de la tercera en dirección  $Z$ , tomando un valor de -20 milímetros

El siguiente paso consiste en elaborar la tabla Denavit-Hartenberg según se muestra en la **Tabla 1** para este caso particular, quedando de la siguiente manera:

	$\Phi$ (°)	d (mm)	a (mm)	$\alpha$ (°)
<b>0-1</b>	$q_1$	70	0	0
<b>1-2</b>	0	$q_2$	100	0
<b>2-3</b>	$q_3$	-20	0	0

**Tabla 5:** Parámetros Denavit-Hartenberg para el caso particular.

Donde para realizar los cálculos posteriores se tomará como valores de  $q_1$ ,  $q_2$  y  $q_3$  los valores asignados anteriormente como posición de reposo.

A continuación, se calcula cada una de las matrices homogéneas individuales para cada eslabón siguiendo la formula mostrada en la **Figura 4**. Para este apartado se va a utilizar el *software* MatLab ya que ofrece las herramientas necesarias para el cálculo matricial correspondiente, quedando las matrices individuales de la siguiente forma:

- Matriz para el paso del eje 0 al eje 1

$$\begin{pmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 70 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

- Matriz para el paso del eje 1 al eje 2

$$\begin{pmatrix} 1 & 0 & 0 & 100 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 200 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

- Matriz para el paso del eje 2 al eje 3

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -20 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Para completar la resolución de la cinemática directa del robot, el último paso es obtener la matriz homogénea total del sistema a partir de la multiplicación en orden de las matrices individuales anteriores. Para esta parte se utilizará el MatLab al igual anteriormente, quedando la matriz de transformación homogénea del conjunto del robot de la siguiente manera:

$$\begin{pmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & -100 \\ 0 & 0 & 1 & 250 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Con esto, el modelo cinemático directo queda totalmente resuelto.



### 4.3.- CINEMÁTICA INVERSA

Obtenida la cinemática directa del modelo, el siguiente paso es determinar las ecuaciones que rigen la cinemática inversa del robot. Como se ha explicado anteriormente, la cinemática inversa de un robot es capaz de encontrar un valor articular para cada articulación del robot con la finalidad de que consiga posicionar su órgano terminal en una posición conocida.

Para la resolución de la cinemática inversa del modelo, se seguirán los pasos explicados anteriormente en el apartado 1.3.2.2.

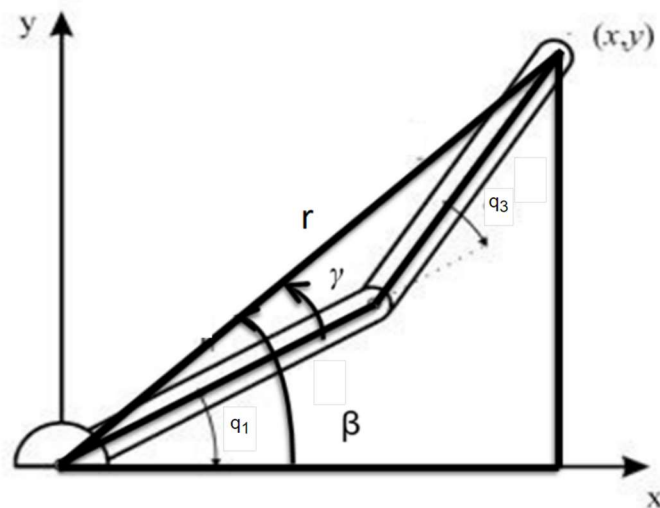
En primer lugar se resolverán las articulaciones rotacionales, siguiendo la **Tabla 2**, obteniendo las siguientes ecuaciones para el diseño del proyecto.

q1 (configuración codo arriba)	$Arctan\left(\frac{y}{x}\right) + Arccos\left(\frac{100^2 + r^2 - 100^2}{2 * 100 * r}\right)$
q1 (configuración codo abajo)	$Arctan\left(\frac{y}{x}\right) - Arccos\left(\frac{100^2 + r^2 - 100^2}{2 * 100 * r}\right)$
q3 (configuración codo arriba)	$Arccos\left(\frac{100^2 - r^2 + 100^2}{2 * 100 * l_3}\right) - \pi$
q3 (configuración codo abajo)	$\pi - Arccos\left(\frac{100^2 - r^2 + 100^2}{2 * 100 * l_3}\right)$

**Tabla 6:** Resolución de la cinemática inversa.

Siendo X e Y los valores en coordenadas cartesianas del objeto de interés.

La obtención del valor de r se ha resuelto de la siguiente manera:

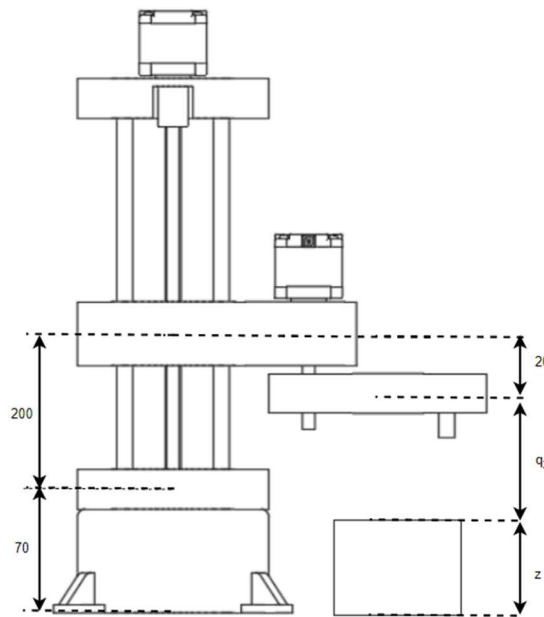


**Figura 29:** Obtención del parámetro r.

Tomando como valor:

$$r = \sqrt{x^2 + y^2}$$

Para la obtención del valor de  $q_2$  cabe tener en cuenta la longitud del eslabón prismático. Para determinar estas ecuaciones se ha optado por el boceto de la posición de reposo, ya que esta posición va a marcar el límite de desplazamiento de dicho eslabón como se muestra en la **Figura 30**.



**Figura 30:** Obtención del parámetro  $q_2$ .

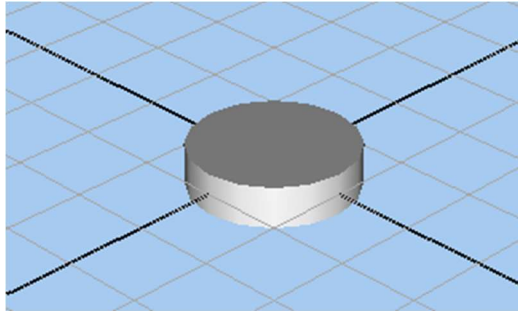
Dando como resultado:

$$q_2 = 200 + 70 - 20 - z$$

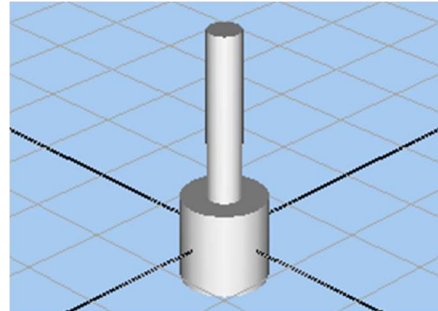
Resueltas las ecuaciones para obtener los valores de  $q_1$ ,  $q_2$  y  $q_3$ , la cinemática inversa del modelo queda resuelta.

#### 4.4.- VALIDACIÓN DE LA CINEMÁTICA

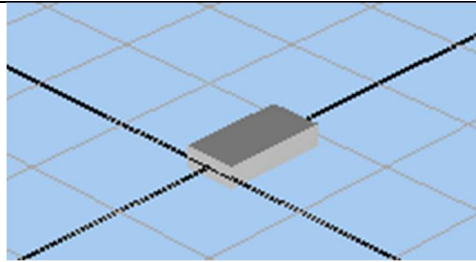
Para la comprobación de la cinemática del modelo, mediante la aplicación RobotScene, explicada en apartados anteriores, se elaboran eslabones similares a los del modelo real, pero de manera simplificada, ya que para el estudio de la cinemática únicamente es necesaria la longitud de cada eslabón, quedando como se muestra a continuación.



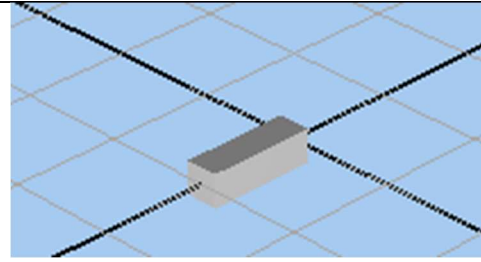
**Figura 31:** Modelado de la base en RobotScene.



**Figura 32:** Modelado del eslabón 1 en RobotScene.



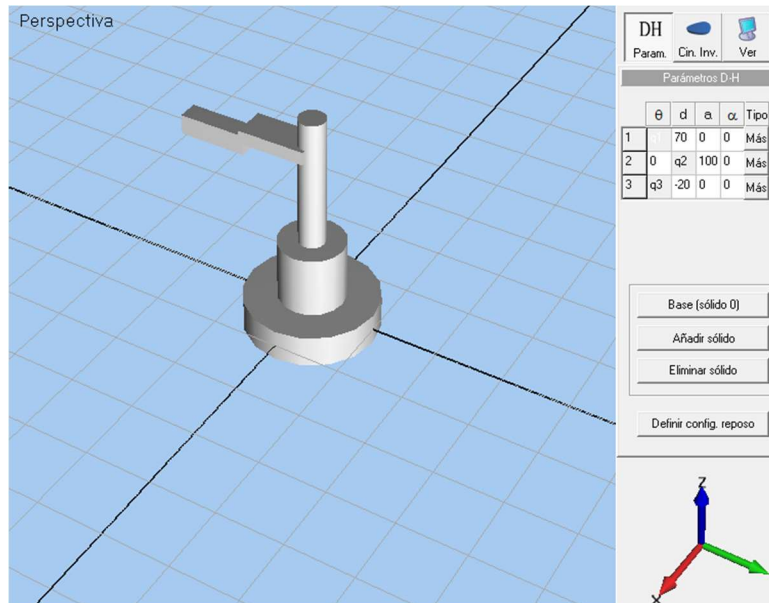
**Figura 33:** Modelado del eslabón 2 en RobotScene.



**Figura 34:** Modelado del eslabón 3 en RobotScene.

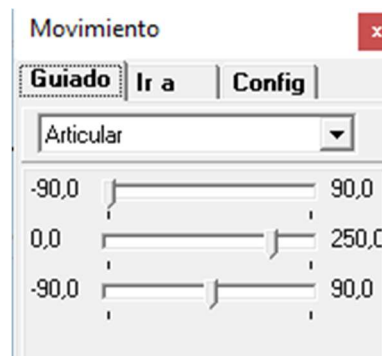
Modelados una vez los diferentes eslabones según el diseño real, el siguiente paso es unirlos para formar la estructura del robot. Para conseguir ensamblar las distintas partes correctamente se sigue la **Tabla 5** del punto anterior para que la simulación sea igual a la realidad, ya que esta tabla muestra la disposición de los ejes de las articulaciones referenciados entre ellos. En la **Figura 35** se puede observar el ensamblaje completo del robot en RobotScene.

DISEÑO, ENSAMBLAJE Y PROGRAMACIÓN DEL CONTROL CINEMÁTICO DE UN BRAZO ROBÓTICO DE TRES GRADOS DE LIBERTAD MEDIANTE CÁMARAS.



**Figura 35:** Ensamblaje del robot en RobotScene.

Para la comprobación del correcto ensamblaje de la simulación, y con ello el modelo cinemático directo, se comprueba, mediante la tabla mostrada en la **Figura 36** que los eslabones se mueven conjuntamente como es esperado, además, gracias a esta tabla de puede ajustar el rango de cada articulación como se desee.



**Figura 36:** Tabla de control del movimiento articular.

En cuanto a la cinemática inversa, una vez obtenidas las ecuaciones en el apartado anterior, gracias al entorno de programación que ofrece el programa, se elabora un código capaz de resolver dichas ecuaciones dados unos valores de coordenadas que se encuentren dentro del rango articular del robot.

Para su comprobación, se ordenará al robot que vaya a determinadas coordenadas, anotando el valor que asocia a cada articulación, y posteriormente se moverán las articulaciones en la tabla que se muestra en la **Figura 36** hasta obtener el valor anotado, comprobando que el punto del órgano terminal es el mismo en ambos casos. El código elaborado en este apartado se utilizará posteriormente en el código de Arduino, es por eso que se explicará más detalladamente en los siguientes puntos del documento.

## 5.- COMPONENTES ELECTRÓNICOS

### 5.1.- MOTORES PASO A PASO NEMA 17

Los motores NEMA 17 (**Figura 37**) son los elegidos para los movimientos de las diferentes articulaciones, debido a que su programación en Arduino es sencilla, y al tratarse de motores paso a paso, el movimiento que van a proporcionar al robot ser controlado.

Estos componentes se tratan de motores bipolares, por lo que ofrecen 4 salidas que irán conectadas a los pines correspondientes como se explicará en los apartados próximos, además cabe destacar que su voltaje de operación suele ser de 12 voltios, aunque puede funcionar con voltajes menores.

Las hojas de documentación técnica de los motores utilizados se mostraran en los siguientes apartados.



**Figura 37:** Motor NEMA 17.

## 5.2.- PLACA ARDUINO UNO

Las placas Arduino (**Figura 38**) permiten crear proyectos electrónicos relacionando un programa informático con diferentes componentes electrónicos. Dentro de todas las opciones que presentan las placas Arduino, la opción elegida es Arduino UNO, ya que es la más económica y al mismo tiempo cumple con las necesidades del proyecto.

Esta placa contiene 14 pines de entrada y salida donde se conectará cada motor NEMA 17. Esta placa es capaz de ofrecer un voltaje a los pines de salida de 3.3 o 5 voltios.

Además, para conectar la placa con el ordenador para tanto cargar el programa como para darle alimentación, se utilizará un cable USB 2.0.

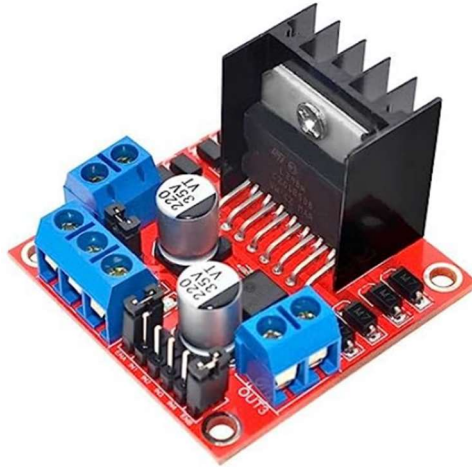


**Figura 38:** Placa Arduino UNO.

## 5.3.- MODULOS L298n

Los módulos L298n (**Figura 39**) son controladores de motores tanto de corriente continua como paso a paso. Estos dispositivos permiten el control de la velocidad de rotación de dichos motores y sentido de giro de su eje, además de ofrecer una intensidad de corriente mayor que la placa Arduino.

Presenta 4 pines de salida, que son los que proporcionan al motor los pulsos eléctricos, y 4 pines de entrada que son los que se conectan al Arduino para recibir las instrucciones, además presenta otros 3 pines para la alimentación y toma tierra.



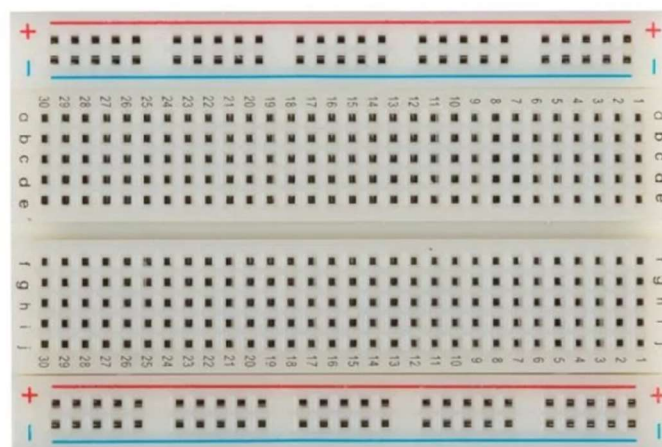
**Figura 39:** Modulo L298n.

#### 5.4.- PLACA DE PRUEBAS

Para poder establecer la conexión de todos los aparatos electrónicos de manera segura y organizada se utilizará una placa de pruebas (**Figura 40**), debido a que estas presentan gran cantidad de pines conectados internamente entre ellos.

Las placas de pruebas se dividen en tres partes, la parte de color rojo, que sirve para dar alimentación a la placa, la parte de color azul que hace la función de toma tierra, y la parte central sin color, donde se encuentran todos los pines para conectar los componentes eléctricos.

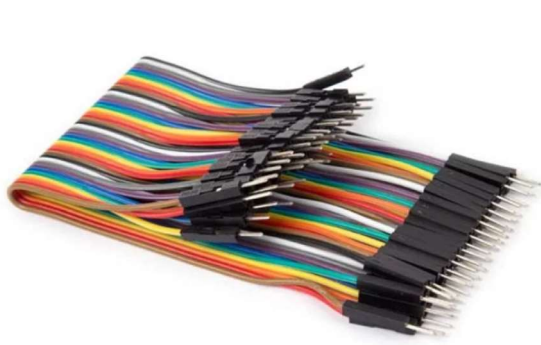
Cabe destacar que, en una placa de pruebas, los pines de una misma columna están conectados todos entre ellos, mientras que los pines de una misma fila son independientes.



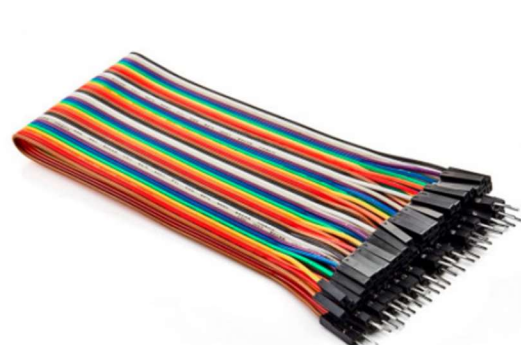
**Figura 40:** Placa de pruebas.

### 5.5.- CABLES PUENTE

Los cables puente se utilizarán para establecer la conexión entre los diferentes componentes electrónicos que forman parte del proyecto. Se utilizarán tantos cables macho-macho (**Figura 41**), para cuando la distancia entre los componentes a conectar sea menor de 40 centímetros, y cables macho-hembra (**Figura 42**) para cuando la distancia entre dichos componentes sea mayor de 40 centímetros, permitiendo establecer empalmes entre ellos para abarcar mayor recorrido.



**Figura 41:** Cables macho-macho.



**Figura 42:** Cables macho-hembra.

### 5.6.- ALIMENTACIÓN EXTERNA

Se utilizará una fuente de alimentación externa (**Figura 43**) capaz de suministrar a todos los aparatos electrónicos la suficiente potencia para su funcionamiento. Esta fuente se conectará a la placa de pruebas, y dichos aparatos, a su vez, a ella, y mediante la conexión de los diferentes pines de la placa, se conseguirá aportar el voltaje necesario a cada componente.



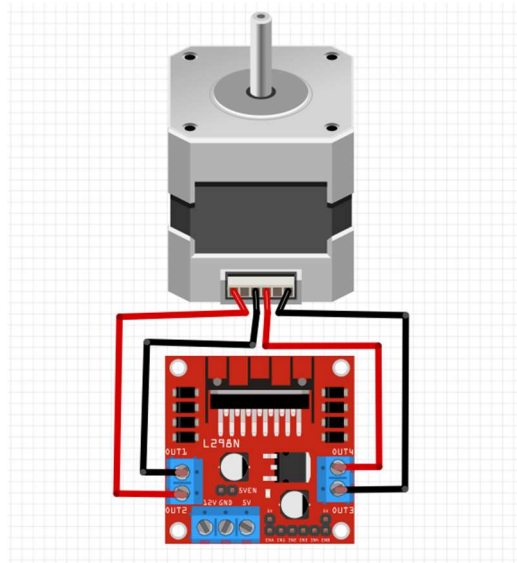
**Figura 43:** Fuente de alimentación externa.



## 5.8.- CONEXIONADO ENTRE LOS COMPONENTES

La conexión entre los diferentes componentes electrónicos se simulará en primer lugar en los programas TnkerCAD y Fritzing, mencionados anteriormente, para comprobar que todos los componentes estén bien conectados. Seguidamente se procederá al montaje real siguiendo la simulación.

En primer lugar, se simulará el conexionado de los motores NEMA 17, con los módulos I298n. Como todos ellos son iguales, solo se mostrará el montaje de 1 conjunto.

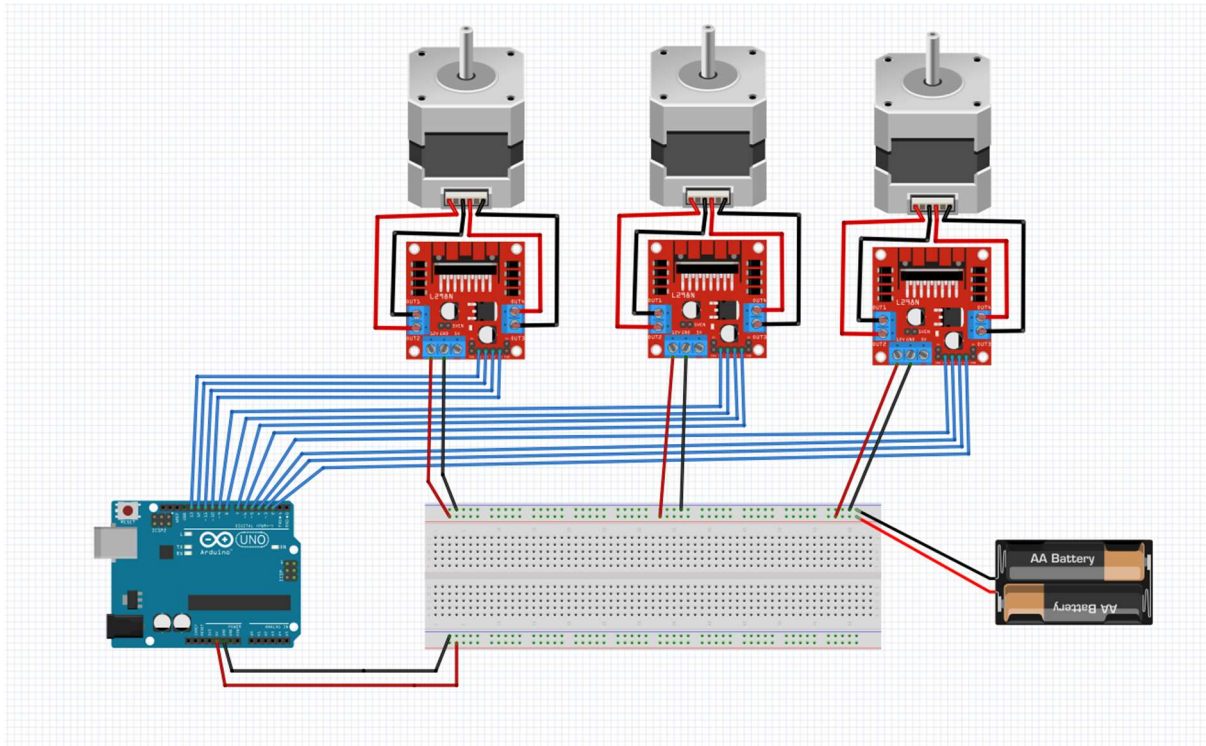


**Figura 44:** Conexionado entre el motor y el módulo I298n.

Los cables se conectarán de manera que las parejas de pines de las bobinas del motor A, A- y B, B-, se conecten a la misma pareja de salidas del controlador I298n, como se observa en la **Figura 44**. Estas parejas se identificarán en las hojas de especificaciones de los motores en apartados posteriores.

Conectados todos los motores con su respectivo controlador, el siguiente paso es establecer la conexión con Arduino. Para realizar esta conexión se utilizará la placa de pruebas que alimentará a cada controlador y motor, y nos ayudará a tener el montaje más organizado.

Cabe destacar que los pines conectados a los controladores deben ser los mismos que los usados después en la programación.



**Figura 45:** Conexionado de todos los elementos.

El conjunto de pilas mostrado en la **Figura 45**, simula la fuente de alimentación de sistema a 12 voltios.

## 6.- PROGRAMACIÓN

La programación llevada a cabo en este proyecto se dividirá en dos partes. En primer lugar, la programación asociada al robot, elaborada en Arduino, y en segundo lugar, la programación asociada a la cámara, elaborada en *Python*.

### 6.1.- ARDUINO

Para establecer control de las articulaciones del robot se elaborará un código Arduino que se explicará por partes en este apartado.

En primer lugar, se incluyen las librerías que se van a utilizar durante la elaboración del código. Estas librerías son la librería “Stepper.h”, encargada de establecer la conexión del motor con el microcontrolador Arduino, y la librería “math.h”, que nos va a permitir realizar cálculos matemáticos.

```
1 #include <math.h> //Se incluye la libreria para realizar cálculos matemáticos
2 #include <Stepper.h> //Se incluye la libreria para establecer el control sobre los motores paso a paso
```

**Código 1:** Librerías de Arduino.

En segundo lugar se declaran las variables que se van a utilizar tanto para almacenar datos, como por ejemplo las coordenadas X, Y, Z que ha de alcanzar el robot, como las longitudes de los eslabones. Además, se definen los pasos que han de realizar los motores en cada vuelta. Cabe destacar que los motores NEMA 17 realizan giros de 1.8°.

```
5 #define PasosPorVuelta 200 //Se definen los pasos que da el motor en una vuelta
6 float X, Y, Z; //Declaración de las variables X, Y, Z
7 float L1 = 100, L2 = 100, L3 = 200, conf; //Declaración de las variables de longitud
8 float beta, gamma, delta, r; //Declaración de las variables utilizadas para los calculos
9 float q1, q2, q3; //Declaración de las variables equivalentes a las articulaciones
10
```

**Código 2:** Declaración de las variables de Arduino.

Seguidamente se define cada motor que se va a utilizar, asociándolo a la articulación a la cual va a proporcionar el movimiento, y se establecen los pines de conexión con Arduino de cada motor.

```
11 Stepper Motorq1(PasosPorVuelta, 2, 3, 4, 5); //Se define los pasos por vuelta y pines del motor asociado a la articulacion 1
12 Stepper Motorq2(PasosPorVuelta, 6, 7, 8, 9); //Se define los pasos por vuelta y pines del motor asociado a la articulacion 2
13 Stepper Motorq3(PasosPorVuelta, 10, 11, 12, 13); //Se define los pasos por vuelta y pines del motor asociado a la articulacion 3
```

**Código 3:** Definición de los motores paso a paso.

Una vez todas las variables y motores definidos, se inicia el *setup* del programa, que es el responsable de establecer la posición de reposo del robot y asignar a cada variable un valor. El *setup* se divide en dos partes, la primera parte establece la conexión con la placa Arduino y los parámetros de trabajo de los motores, y en segundo lugar recibe y almacena un vector enviado por el programa Python que contiene las coordenadas del objeto y la configuración del robot elegida. Cabe destacar que en este paso se utiliza la relación de cm/píxel establecida anteriormente para que el robot sepa las coordenadas en cm a las cuales ha de enviar su órgano terminal.

```
15 void setup() {
16     Serial.begin(9600); // Establece la comunicación entre Arduino y ordenador a 9600 baudios
17     Motorq1.step(0); //Lleva la articulación 1 a la posición de reposo
18     Motorq2.step(0); //Lleva la articulación 2 a la posición de reposo
19     Motorq3.step(0); //Lleva la articulación 3 a la posición de reposo
20     Motorq1.setSpeed(90); //Asigna una velocidad de rotación de 90rpm al motor 1
21     Motorq2.setSpeed(90); //Asigna una velocidad de rotación de 90rpm al motor 2
22     Motorq3.setSpeed(90); //Asigna una velocidad de rotación de 90rpm al motor 3
23 }
```

**Código 4:** Primera parte del Setup.

```
24 while (!Serial.available()) {} //Espera a recibir datos para avanzar
25 String data = Serial.readStringUntil(','); //Recibe los datos de la posicion 1 de un vector enviado por python
26 Z = data.toFloat(); //Asigna el primer elemento del vector a la variable Z
27
28 data = Serial.readStringUntil(','); //Recibe los datos de la posicion 2 de un vector enviado por python
29 conf = data.toFloat(); //Asigna el primer elemento del vector a la variable Z
30
31 data = Serial.readStringUntil(','); //Recibe los datos de la posicion 3 de un vector enviado por python
32 X = data.toFloat() * 0.026; //Asigna el primer elemento del vector a la variable Z
33
34 data = Serial.readStringUntil('\n'); //Recibe los datos de la posicion 4 un vector enviado por python
35 Y = data.toFloat() * 0.026 -26; //Asigna el primer elemento del vector a la variable Z
36
37 }
38
```

**Código 5:** Segunda parte del Setup.

Establecido el *setup* del programa, el siguiente paso es iniciar el *loop*, que va a ser el responsable de ejecutar las acciones que se escriban dentro de él. El *loop* se divide en dos partes. En primer lugar, la parte del cálculo de las articulaciones a partir de las coordenadas recibidas. Este paso se resuelve utilizando las ecuaciones de la cinemática inversa resueltas en apartados anteriores y esta parte del código es la misma parte que se emplea en la comprobación de la cinemática en el RobotScene. La segunda parte del *loop* corresponde con la transformación de radianes a grados de los resultados de las articulaciones, y el envío de la señal a los motores en pasos.

```
40 void loop() {
41
42     beta = atan(Y/X); //Cálculo de la variable beta
43     r = sqrt(X*X+Y*Y); //Cálculo de la variable r
44     gamma = acos((L1*L1+r*r-L2*L2)/(2*r*L1)); //Cálculo de la variable gamma
45     delta = acos((L1*L1+L2*L2-r*r)/(2*L2*L1)); //Cálculo de la variable delta
46
47     q1 = beta - gamma; //Cálculo del valor de la articulación 1 en caso de elegir la configuración codo abajo
48     if (conf == 1){
49         q1 = beta + gamma; //Cálculo del valor de la articulación 1 en caso de elegir la configuración codo arriba
50     }
51     Serial.println(q1);
52
53     q3 = PI - delta; //Cálculo del valor de la articulación 3 en caso de elegir la configuración codo abajo
54     if(conf == 1){
55         q3 = delta - PI; //Cálculo del valor de la articulación 3 en caso de elegir la configuración codo arriba
56     }
57
58     q2 = L3 + 70 -20 - Z; //Cálculo del valor de la articulación 2
```

**Código 6:** Primera parte del loop.

```
60     q1 = 180 * q1/PI; //Cambio de radianes a grados
61     q3 = 180 * q3/PI; //Cambio de radianes a grados
62
63     Motorq1.step(q1 / 1.8); //Envía al motor 1 los pasos que ha de realizar para alcanzar el punto deseado
64     Motorq2.step(q2 / 1.8); //Envía al motor 2 los pasos que ha de realizar para alcanzar el punto deseado
65     Motorq3.step(q3 / 1.8); //Envía al motor 3 los pasos que ha de realizar para alcanzar el punto deseado
66
67 }
```

**Código 7:** Segunda parte del loop.

Con esto, el código de Arduino para el control del robot queda completado.

## 6.2.- PYTHON

Para el código de la cámara se emplea el lenguaje Python, debido a las librerías que ofrece.

En primer lugar se importan al programa las librerías que se van a utilizar durante este. Para el control de la cámara, las librerías que se han utilizado son: “cv2”, que permite establecer el control sobre una o varias cámaras conectadas al dispositivo, “numpy”, que es la responsable de la identificación de colores y realización de operaciones sobre ellos mediante el modelo RGB explicado anteriormente en este documento, “serial”, que permite la conexión del programa Python con los puertos de salida del ordenador, estableciendo así, de manera indirecta, la conexión con Arduino para realizar la automatización, y “time”, para establecer paradas de tiempo.

```
1 import cv2 #Importación de la librería OpenCV
2 import numpy as np ##Importación de la librería numpy como np
3 import serial ##Importación de la librería serial
4 import time ##Importación de la librería time
5
```

*Código 8: Librerías de Python.*

Importadas las librerías. El siguiente paso es la declaración de variables. En esta parte del código, se definen las variables del puerto de salida en el cual se va a conectar Arduino, así como el rango de colores según la tabla RGB mostrada en la **Figura 6** que va a detectar la cámara. Cabe destacar que en esta imagen se muestra el rango de colores rojo, pudiéndose cambiar a cualquier otro color utilizando la tabla RGB.

```
7 puerto = "COM3" #Define el puerto al que va conectado arduino
8 rojobajo = np.array([0,100,20],np.uint8) #Define el rango de colores bajo
9 rojoalto = np.array([0,255,255],np.uint8) #Define el rango de colores alto
10 celda = 50 #Defina la variable celda como 10
```

*Código 9: Declaración de variables en Python.*

Definidas las variables, se definen las funciones que se van a utilizar en el programa, en este caso dos. La primera función va a ser la encargada de, una vez la imagen haya sido tomada, realizar una cuadrícula sobre ella con la intención de, posteriormente, comprobar que la calibración de la cámara sea correcta. La segunda función corresponderá con la obtención de las coordenadas de todos los puntos cuyos colores coincidan con los definidos en las variables anteriores.

```
12 def cuadrícula(capt,celda): #Se define la función de cuadricular la imagen
13
14     alto,ancho, _ = capt.shape #Se identifica el ancho y alto de la imagen
15     for x in range(0,ancho,celda):
16         cv2.line(capt,(x,0),(x,alto),(0,255,0),1) #Se dibujan líneas de color verde a lo largo de todo
17                                                     #el alto de la imagen a una distancia igual al valor de la celda
18     for y in range(0,alto,celda):
19         cv2.line(capt,(0,y),(ancho,y),(0,255,0),1) #Se dibujan líneas de color verde a lo largo de todo
20                                                     #el ancho de la imagen a una distancia igual al valor de la celda
21
22     return capt #Devuelve la imagen cuadrículada
```

**Código 10:** Primera función.

```
def encontrar_coordenadas(capt):
    coordenadas = [] #Define un vector con el nombre de coordenadas
    for contorno in contours:
        centro = cv2.moments(contorno) #Busca el centro de los puntos de los colores asignados
        if centro["m00"] != 0:
            x = int(centro["m10"] / centro["m00"]) #Localiza la coordenada X del punto localizado utilizando en momento de la geometría del punto
            y = int(centro["m01"] / centro["m00"]) #Localiza la coordenada Y del punto localizado utilizando el momento de la geometría del punto
            coordenadas.append((x,y)) #Asigna la variable 1 del vector coordenadas al punto X y la variable 2 al punto Y
    return coordenadas #Devuelve un vector con las coordenadas
```

**Código 11:** Segunda función.

Definidas las funciones, el siguiente paso es conectar la cámara con el programa, y una vez conectada, definir los píxeles de la imagen que ha de tomar, en este caso 1920x1080. Seguidamente, la cámara toma una captura y la guarda en la carpeta donde se encuentra el código para utilizarla posteriormente.

```
36 cap = cv2.VideoCapture(0) #Abre la cámara
37 cap.set(cv2.CAP_PROP_FRAME_WIDTH, 1920) #Establece el ancho en píxeles de la imagen tomada
38 cap.set(cv2.CAP_PROP_FRAME_HEIGHT, 1080) #Establece el alto en píxeles de la imagen tomada
39 ret, capt = cap.read() #Captura una imagen y la guarda en la variable capt
40 cv2.imwrite('Imagen.jpg',capt) #Guarda la imagen en la carpeta donde se encuentra el programa
41 cap.release() #Cierra la cámara
```

**Código 12:** Conexión y puesta en marcha de la cámara.

Realizada la captura con la cámara, se procede a identificar los colores dentro de la imagen, y a resaltar los que cumplan con el rango RGB definido en el apartado de variables. Cabe destacar que para que el programa identifique los colores correctamente ha de cambiar el formato de la imagen a HSV. Una vez localizados los colores definidos, el programa inicializa ambas funciones anteriores, con el fin de localizar los puntos encontrados.

```
43 frameHSV = cv2.cvtColor(capt,cv2.COLOR_BGR2HSV) #Transformación de BRG a HSV
44 mask = cv2.inRange(frameHSV,rojobajo,rojoalto) #Rango de colores
45 contours, _aux = cv2.findContours(mask,cv2.RETR_TREE,cv2.CHAIN_APPROX_SIMPLE) #Encuentra el borde de las zonas con el color definido
46 cv2.drawContours(capt, contours, -1, (255,0,0), 3) #Dibuja los bordes encontrados
47 capt_cuadr = cuadrícula(capt,celda) #Inicia la función de cuadricular la imagen
48 puntos_objeto = encontrar_coordenadas(capt) #Inicia la función de encontrar las coordenadas
49 for coordenadas in puntos_objeto:
50     print("Coordenadas: ({}).format(coordenadas[0],coordenadas[1])) #Muestra en pantalla las coordenadas de los puntos encontrados
```

**Código 13:** Localización de los puntos mediante los colores definidos.

Localizados los puntos, el programa pide al usuario tanto la altura del objeto como la configuración que va a desear que adopte el robot.

```
52 Z = int(input("Introduzca el valor de la altura: ")) #Pide al usuario el valor de la altura del objeto
53                                     #y lo almacena en la variable Z
54 C = int(input("Que configuración desea? (1 codo arriba, 2 codo abajo): ")) #Pide al usuario la configuración
55                                     #que desea y lo almacena en la variable C
```

**Código 14:** *Obtención de la altura del objeto y configuración del robot.*

Obtenidos los datos de las coordenadas, el valor de la altura, y la configuración, el siguiente paso es crear un vector con estos datos para enviarlo al puerto que esta conectado Arduino, así, el programa de Arduino será capaz de realizar las operaciones de cinemática a partir de las coordenadas.

```
56 Enviar = "{} , {} , {} , {}".format(Z,C,coordenadas[1],coordenadas[0]) #Define enviar como una cadena de 4 valores
57 ser = serial.Serial(puerto, 9600) #Establece la conexión con el puerto definido
58 time.sleep(2) #Deja un descanso de 2 segundos para asegurarse de la correcta conexión
59 ser.write(Enviar.encode()) #Envía los datos de la cadena Enviar al puerto definido
```

**Código 15:** *Conexión con Arduino.*

Enviados los datos, la última instrucción que realiza el programa es mostrar la imagen que ha tomado, cuadrículada, y con los colores definidos resaltados para que el usuario pueda comprobar la correcta detección del objeto.

```
61 cv2.imshow('Captura',capt_cuadr) #Abre una ventana mostrando la imagen
62 cv2.waitKey(0) #Espera a que se pulse la X para cerrar la ventana
63 ser.close() #Finaliza el programa
```

**Código 16:** *Cierre del programa*

Con ambos códigos resueltos y con la posibilidad de comprobación ya sea, visualmente en el de Python, o mediante simulación el de Arduino, tanto la cámara como el robot están listos para trabajar conjuntamente de manera autónoma. El usuario únicamente debería de iniciar ambos programas cuando quiera tomar una nueva imagen para enviarla al robot.

## 7.- CONCLUSIONES

Diseñado el ensamblaje de un robot de tres grados de libertad controlado por una cámara, con sus correspondientes subensamblajes del soporte, parte superior y brazos por medio del software Fusion 360, añadiendo el ensamblaje del soporte para la cámara, ha supuesto un periodo largo de aprendizaje y desarrollo de habilidades en diferentes programas informáticos muy diversos entre ellos, tanto de diseño, como de programación y simulación, como de cálculo.

Además, tanto el estudio de la cinemática del modelo, que ha permitido el profundizar en el campo de la robótica, como el diseño y montaje de todo los esquemas y componentes eléctricos, ha supuesto un acercamiento a los modelos utilizados en la industria actual.

Cabe mencionar que durante el desarrollo del proyecto han aparecido numerosas adversidades que se han conseguido superar hasta conseguir un diseño final del robot capaz de trabajar como se había planteado desde un principio.

El punto más destacable de implementación de nuevas tecnologías en este proyecto se basa en conseguir el control automático del robot mediante el uso de la cámara, permitiendo a este trabajar de manera autónoma pudiendo así cambiar sus trayectorias en cada momento necesario a diferencia de los robots convencionales industriales, que suelen estar diseñados para realizar tareas programadas previamente y muy repetitivas.

Algunas propuestas de mejora que se han planteado durante el desarrollo del proyecto y que podrían servir para en un futuro desarrollar un modelo más complejo capaz de realizar trabajos más costos y en espacios de trabajo juntamente con otros robots han sido:

- Implementación de un cuarto grado de libertad, que permita al robot añadir una pinza en su extremo del órgano terminal para que, una vez localizado el objeto, sea capaz de cogerlo y transportarlo.
- Implementación de detección de formas para que el usuario pueda mandar ordenes al robot de realizar tareas tanto por búsqueda de colores como por búsqueda de formas para aumentar su funcionalidad.
- Adición de más cámaras que sean capaces conjuntamente de modelar el espacio de trabajo mediante un programa para permitir al robot que sortee posibles obstáculos que se encuentren en sus trayectorias obtenidas.



## 8.- OBJETIVOS DE DESARROLLO SOSTENIBLES

Objetivos de Desarrollo Sostenibles	Alto	Medio	Bajo	No Procede
ODS 1. Fin de la pobreza				X
ODS 2. Hambre cero.				X
ODS 3. Salud y bienestar				X
ODS 4. Educación de calidad.		X		
ODS 5. Igualdad de género.				X
ODS 6. Agua limpia y saneamiento.				X
ODS 7. Energía asequible y no contaminante.	X			
ODS 8. Trabajo decente y crecimiento económico	X			
ODS 9. Industria, innovación e infraestructuras.	X			
ODS 10. Reducción de las desigualdades.				X
ODS 11. Ciudades y comunidades sostenibles.				X
ODS 12. Producción y consumo responsables.				X
ODS 13. Acción por el clima.				X
ODS 14. Vida submarina.				X
ODS 15. Vida de ecosistemas terrestres.				X
ODS 16. Paz, justicia e instituciones sólidas.				X
ODS 17. Alianzas para lograr objetivos.				X

*Tabla 7: Objetivos de desarrollo sostenibles*

Debido a que el desarrollo planteado en este proyecto se basa en la aplicación de las nuevas tecnologías a la hora de plantear una propuesta de mejora del sistema industrial actual por medio de uso de robots automáticos monitorizados por cámaras, los objetivos de desarrollo sostenibles se basan principalmente en la innovación. Por este motivo se califica el objetivo de desarrollo sostenible número 9 como alto.

Por otra parte, se pretende reflejar tanto lo estudiado durante los cursos de ingeniería, como plasmar un modelo para la posible futura investigación y propuesta de mejora para la incorporación del modelo diseñado en la industria moderna. Es por ello que los objetivos de desarrollo sostenible números 4 y 8 se clasifican con alta puntuación.

Para finalizar, la energía eléctrica consumida por el robot se puede clasificar como energía fácil de conseguir en la industria y renovable, debido a que todo su funcionamiento se basa en energía eléctrica, por esta razón, se califica el objetivo de desarrollo número 7 con puntuación alta.

Los demás objetivos de desarrollo sostenibles no son relevantes en el planteamiento de este proyecto.

## 9.- BIBLIOGRAFÍA

- [1] Junior Report (2022). *Historia de la robótica*, Periódico la Vanguardia.
- [2] González Sorribes, Antonio (2022). *Tema 2: Localización espacial. Apuntes de la asignatura de robótica*, Universidad Politécnica de Valencia.
- [3] González Sorribes, Antonio (2022). *Tema 3: Cinemática directa. Apuntes de la asignatura de robótica*, Universidad Politécnica de Valencia.
- [4] A.Barrientos, L.F. Peñín, C.Balanguer, R.Aracil (1997). *Fundamentos de robótica*, McGraw Hill.
- [5] González Sorribes, Antonio (2022). *Tema 4: Cinemática inversa. Apuntes de la asignatura de robótica*, Universidad Politécnica de Valencia.
- [6] P. Llasera, Jaime (2020) *RGB y CMYK: Qué son y cuándo usar cada modo de color*, <https://imborrable.com/blog/rgb-y-cmyk/>
- [7] Fernández, Yúbal (2022), *Qué es Arduino, cómo funciona y qué puedes hacer con uno*, <https://www.xataka.com/basics/que-arduino-como-funciona-que-puedes-hacer-uno>

DISEÑO, ENSAMBLAJE Y PROGRAMACIÓN DEL CONTROL CINEMÁTICO DE UN BRAZO ROBÓTICO DE TRES GRADOS DE LIBERTAD MEDIANTE CÁMARAS.

## **PRESUPUESTO**

## 1.- MATERIALES

Referencia	Unidad	Descripción	Precio (€)
M1	Ud.	Arduino UNO Rev3	24.19
M2	Ud.	Cable Macho-Macho 20cm	0.10
M3	Ud.	Cable Macho-Hembra 20cm	0.10
M4	Ud.	Mini BreadBoard 400 pines	3.71
M5	Ud.	Motor paso a paso NEMA 17 17HS15-1504S-X1	13.30
M6	Ud.	Motor paso a paso NEMA 17 17HS08-1004S	10.78
M7	Ud.	Módulo L298N	3.02
M8	Ud.	Acoplamiento flexible 5mm a 8mm	1.26
M9	Ud.	Rodamiento lineal LM10UU	1.47
M10	Ud.	Rodamiento de bolas 6008-2RS	13.18
M11	Ud.	Rodamiento de bolas 608-2RS	1.44
M12	Ud.	Varilla cilíndrica de acero 400mm de longitud y 10 mm de diámetro	11.56
M13	Ud.	Varilla cilíndrica roscada 400mm de longitud y 8 mm de diámetro	8.08
M14	Ud.	Tornillo M3x10mm	0.12
M15	Ud.	Tornillo M3x20mm	0.15
M16	Ud.	Tornillo M3x30mm	0.17
M17	Ud.	Tornillo M5x30mm autoperforantes	0.22
M18	Ud.	Adhesivo comercial de montaje	5.80
M19	Ud.	Goma eva para las almohadillas	0.53
M20	Ud.	Tablero de madera comercial aglomerado 120x60 cm	19.00
M21	Ud.	WebCam ASHU Streaming 1080P	19.83

**Tabla 8:** Materiales utilizados

## 2.- MANO DE OBRA

Referencia	Unidad	Descripción	Precio (€)
MO1	h	Ingeniero nóvel	20
MO2	h	Técnico de impresión 3D	10
MO3	h	Técnico de mecanizado	10
MO4	h	Carpintero	10

**Tabla 9:** Mano de obra

#### 4.- ENSAMBLAJE DEL SOPORTE

Ref.	Unidad	Descripción	Precio (€)	Cantidad	Parcial (€)
E1	Ud.	Ensamblaje del soporte de la base de la estructura del robot			
P1	Ud.	Base del robot	6	1	6
P2	Ud.	Hombro del robot	6	1	6
M5	Ud.	Motor paso a paso NEMA 17 17HS15-1504S-X1	13.3	1	13.3
M10	Ud.	Rodamiento de bolas 6008-2RS	13.18	1	13.18
M11	Ud.	Rodamiento de bolas 608-2RS	1.44	1	1.44
M12	Ud.	Varilla cilíndrica de acero 400mm de longitud y 10 mm de diámetro	11.56	3	34.68
M13	Ud.	Varilla cilíndrica roscada 400mm de longitud y 8 mm de diámetro	8.08	1	8.08
M15		Tornillo M3x20mm	0.15	6	0.9
MO2	h	Técnico de impresión 3D	10	5	50
MO3	h	Técnico de mecanizado	10	0.5	5
	%	Medios auxiliares sobre los costes	138.58	10	13.58
				TOTAL	152.16

**Tabla 10:** Presupuesto del ensamblaje del soporte

#### 5.- ENSAMBLAJE DE LA PARTE SUPERIOR

Ref.	Unidad	Descripción	Precio (€)	Cantidad	Parcial (€)
E2	Ud.	Ensamblaje de la parte superior encargada de proporcionar estabilidad al robot			
P3	Ud.	Parte superior	6	1	6
M5	Ud.	Motor paso a paso NEMA 17 17HS15-1504S-X1	13.18	1	13.18
M8	Ud.	Acoplamiento flexible 5mm a 8mm	1.26	1	1.26
MO2	h	Técnico de impresión 3D	10	2.5	25
	%	Medios auxiliares sobre los costes	45.44	10	4.54
				TOTAL	49.98

**Tabla 11:** Presupuesto del ensamblaje de la parte superior

## 6.- ENSAMBLAJE DE LOS BRAZOS

Ref.	Unidad	Descripción	Precio (€)	Cantidad	Parcial (€)
E3	Ud.	Ensamblaje formado por los dos brazos del robot			
P4	Ud.	Brazo 1	6	1	6
P5	Ud.	Brazo 2	6	1	6
M6	Ud.	Motor paso a paso NEMA 17 17HS08-1004S	10.78	1	10.78
M8	Ud.	Acoplamiento flexible 5mm a 8mm	1.26	1	1.26
M9	Ud.	Rodamiento lineal LM10UU	1.47	3	4.41
M13	Ud.	Varilla cilíndrica roscada 400mm de longitud y 8 mm de diámetro	8.08	1	8.08
M14	Ud.	Tornillo M3x10mm	0.12	1	0.12
M16		Tornillo M3x30mm	0.17	4	0.68
MO2	h	Técnico de impresión 3D	10	5	50
MO3	h	Técnico de mecanizado	10	0.5	5
	%	Medios auxiliares sobre los costes	92.33	10	9.23
				TOTAL	101.56

**Tabla 12:** Presupuesto del ensamblaje de los brazos

## 7.- ENSAMBLAJE DEL SOPORTE DE LA CÁMARA

Ref.	Unidad	Descripción	Precio (€)	Cantidad	Parcial (€)
E4	Ud.	SopORTE para la cámara hecho a partir de un tablero de madera comercial cortado por un carpintero			
M17	Ud.	Tornillo M5x30mm autoperforantes	0.22	8	1.76
M20	Ud.	Tablero de madera comercial aglomerado 120x60 cm	19.00	1	19
M21	Ud.	WebCam ASHU Streaming 1080P	19.83	1	19.83
MO4	h	Carpintero	10	5	50
	%	Medios auxiliares sobre los costes	90.59	10	9.059
				TOTAL	99.65

**Tabla 13:** Presupuesto del ensamblaje del soporte de la cámara

## 8.- COSTE TOTAL

Ref.	Unidad	Descripción	Precio (€)	Cantidad	Parcial (€)
E1	Ud.	Ensamblaje del soporte de la base de la estructura del robot	152.16	1	152.16
E2	Ud.	Ensamblaje de la parte superior encargada de proporcionar estabilidad al robot	49.98	1	49.98
E3	Ud.	Ensamblaje formado por los dos brazos del robot	101.56	1	101.56
E4	Ud.	Soporte para la cámara hecho a partir de un tablero de madera comercial cortado por un carpintero	77.84	1	91.65
M1	Ud.	Arduino UNO Rev3	24.19	1	24.19
M2	Ud.	Cable Macho-Macho 20cm	0.10	30	3
M3	Ud.	Cable Macho-Hembra 20cm	0.10	10	1
M4	Ud.	Mini BreadBoard 400 pines	3.71	1	3.71
M7	Ud.	Módulo L298N	3.02	3	9.06
M18	Ud.	Adhesivo comercial de montaje	5.80	1	5.80
M19	Ud.	Goma eva para las almohadillas	0.53	1	0.53
MO1	h	Ingeniero nóvel	20	300	6000
		PRECIO TOTAL SIN IVA			6450.64
	%	IVA	1354.63	21	
		PRECIO TOTAL CON IVA			7805.27

**Tabla 14:** Presupuesto del ensamblaje del proyecto

Tras la realización de todos los presupuestos, cabe aclarar que tras hablar con el técnico de impresión, es precio estimado por pieza es de 6€ debido a las horas empleadas, material de impresión, y peso de la pieza.

El precio final del modelado y ensamblaje del prototipo propuesto asciende a 7805.27 € teniendo en cuenta las horas trabajadas por el estudiante para realizar este proyecto.

DISEÑO, ENSAMBLAJE Y PROGRAMACIÓN DEL CONTROL CINEMÁTICO DE UN BRAZO ROBÓTICO DE TRES GRADOS DE LIBERTAD MEDIANTE CÁMARAS.

## **PLIEGO DE CONDICIONES**



## 1.- OBJETO

El objeto del presente pliego de condiciones se refiere tanto a la fabricación como al ensamblaje de las piezas del modelo del robot de 3 grados de libertad y del soporte de la cámara que va a ser la responsable del movimiento de dicho robot.

El ámbito de aplicación de este documento se extiende a la fabricación de las piezas y a la obtención de todos los demás componentes presentes durante el ensamblaje del modelo. En determinadas situaciones se podrán optar soluciones alternativas a las exigidas en este documento siempre que no implique cambios notables en las especificaciones descritas y su necesidad este justificada.

## 2.- CONDICIONES DE LOS MATERIALES

### 2.1.- ENSAMBLAJE DEL SOPORTE

El ensamblaje del soporte tendrá la función de fijar el robot a una superficie y aportar firmeza y estabilidad a este. Dispondrá de unas dimensiones generales de 130x148x330 mm, además estará formado por la base, el hombro, tres varillas de acero y una varilla de acero roscada.

Al ensamblaje del soporte se le añadirá un motor NEMA 17 modelo 17HS15-1504S-X1, de dimensiones generales 42x42x39 mm y  $\varnothing 5 \times 22$  mm de eje, 240g de peso, y 45Ncm de par de rotación.

#### 2.1.1.- BASE

La base tendrá la función de fijar la estructura del robot a la superficie de apoyo, además de soportar el peso de este. Sus dimensiones generales son  $\varnothing 120 \times 65$  mm, y está fabricada con poliestireno de alto impacto, para las dimensiones específicas, consultar el plano del documento **Planos** de este proyecto.

#### 2.1.2.- HOMBRO

El hombro tendrá la función de ejercer como primera articulación del robot, permitiendo el giro sobre su eje perpendicular y la base en 360 °, además de soportar las piezas que posteriormente se ensamblaran a él. Sus dimensiones generales son  $\varnothing 120 \times 40$  mm, y está fabricada con poliestireno de alto impacto, para las dimensiones específicas, consultar el plano del documento **Planos** de este proyecto.

### **2.1.3.- VARILLAS**

Las varillas, concretamente 3, tendrán la función de ejercer un soporte para el ensamblaje posterior de los brazos sobre el hombro, además de permitir el desplazamiento vertical sobre ellas. Las dimensiones generales son  $\varnothing 10 \times 260$  mm y están fabricadas de acero AISI 1045.

### **2.1.4.- VARILLA ROSCADA**

La varilla roscada tendrá la función de proporcionar el movimiento vertical a los brazos mediante su giro proporcionado por el acople con un motor NEMA 17 colocado en la parte superior del robot. Sus dimensiones generales son  $\varnothing 8 \times 235$  mm y está fabricada de acero.

### **2.1.5.- CONTROL DE CALIDAD**

Una vez fabricadas las piezas, se realizarán mediciones generales de estas y se compararán con las mediciones especificadas en el documento Planos. Si alguna de ellas no cumple con el diseño realizado, se descartará y fabricará de nuevo. Se realizará especial detenimiento en la comprobación de los orificios, sobre todo en los roscados, y si no cumplen con las dimensiones se descartará la pieza y se fabricará de nuevo a excepción de que los orificios que no cumplan las especificaciones que sean planos y se puedan corregir a mano sin comprometer la estructura de la pieza.

Comprobadas las piezas se comprobará que encajen entre ellas según el ensamblaje mostrado en el documento Planos. Si algunas de estas piezas no encajan como es de esperar, se volverán a revisar todas las medidas con especial detenimiento y se descartará la pieza que sea responsable del mal ensamblaje.

## **2.2.- ENSAMBLAJE DE LA PARTE SUPERIOR**

El ensamblaje de la parte superior tendrá la función de cerrar la estructura del robot por la parte superior de este proporcionándole rigidez y mayor estabilidad además de proporcionar movimiento rotacional a la varilla roscada encargada del movimiento vertical del robot. Dispondrá de unas dimensiones generales de  $\varnothing 120 \times 72$  mm. Estará formado por la parte superior y un motor NEMA 17 modelo 17HS15-1504S-X1, de dimensiones generales  $42 \times 42 \times 39$  mm y  $\varnothing 5 \times 22$  mm de eje, 240g de peso, y 45Ncm de par de rotación.

### **2.2.1.- PARTE SUPERIOR**

La parte superior tendrá la función de fijar las varillas de acero de la base, dando al robot mayor estabilidad y seguridad en sus movimientos. Sus dimensiones generales son  $\varnothing 120 \times 25$  mm, y está fabricada con poliestireno de alto impacto, para las dimensiones específicas, consultar el plano del documento **Planos** de este proyecto.

### **2.2.2.- CONTROL DE CALIDAD**

Una vez fabricada la pieza, se realizarán mediciones generales de estas y se compararán con las mediciones especificadas en el documento Planos. Si alguna de ellas no cumple con el diseño realizado, se descartará y fabricará de nuevo. Se realizará especial detenimiento en la comprobación de los orificios, y si no cumplen con las dimensiones se podrán corregir a mano siempre y cuando no comprometan la estructura de la pieza.

Comprobadas las piezas se comprobará que encajen entre ellas según el ensamblaje mostrado en el documento Planos. Si algunas de estas piezas no encajan como es de esperar, se volverán a revisar todas las medidas con especial detenimiento y se descartará la pieza que sea responsable del mal ensamblaje.

## **2.3.- ENSAMBLAJE DE LOS BRAZOS**

El ensamblaje de los brazos tendrá la función unir ambos brazos del robot entre ellos para el posterior montaje sobre el resto de la estructura, además de proporcionar movimiento relativo entre ellos. Dispondrá de unas dimensiones generales de  $120 \times 185 \times 127$  mm. Estará formado por el brazo 1, el brazo 2 y una varilla roscada.

Al ensamblaje de los brazos se añadirá un motor NEMA 17 modelo 17HS08-1004S, de dimensiones generales  $42 \times 42 \times 20$  mm y  $\varnothing 5 \times 20$  mm de eje, 140g de peso, y 16Ncm de par de rotación.

### **2.3.1.- BRAZO 1**

El brazo 1 tendrá la función de establecer la conexión entre el brazo 2 del robot y la estructura de la base, además de servir como soporte para el motor NEMA 17 responsable del movimiento de la última articulación del robot. Sus dimensiones generales son  $120 \times 190 \times 40$  mm, y está fabricada con poliestireno de alto impacto, para las dimensiones específicas, consultar el plano del documento **Planos** de este proyecto.

### **2.3.2.- BRAZO 2**

El brazo 2 tendrá la función de órgano terminal del conjunto. Sus dimensiones generales son 50x150x25 mm, y está fabricada con poliestireno de alto impacto, para las dimensiones específicas, consultar el plano del documento **Planos** de este proyecto.

### **2.3.3.- VARILLA ROSCADA**

La varilla roscada tendrá la función de unir el brazo 1 con el brazo 2 además de ofrecer el movimiento a este segundo brazo mediante la conexión con un acople al motor NEMA 17. Sus dimensiones generales son  $\varnothing 8 \times 55$  mm, y está fabricada de acero.

### **2.3.4.- CONTROL DE CALIDAD**

Una vez fabricadas las piezas, se realizarán mediciones generales de estas y se compararán con las mediciones especificadas en el documento Planos. Si alguna de ellas no cumple con el diseño realizado, se descartará y fabricará de nuevo. Se realizará especial detenimiento en la comprobación de los orificios, sobre todo en los roscados, y si no cumplen con las dimensiones se descartará la pieza y se fabricará de nuevo a excepción de que los orificios que no cumplan las especificaciones que sean planos y se puedan corregir a mano sin comprometer la estructura de la pieza.

Comprobadas las piezas se comprobará que encajen entre ellas según el ensamblaje mostrado en el documento Planos. Si algunas de estas piezas no encajan como es de esperar, se volverán a revisar todas las medidas con especial detenimiento y se descartará la pieza que sea responsable del mal ensamblaje.

## **2.4.- ENSAMBLAJE DEL SOPORTE DE LA CÁMARA**

El ensamblaje del soporte de la cámara tendrá como función soportar la cámara que va a ordenar al robot el movimiento, además de establecer un espacio de trabajo fijo para este. Dispondrá de unas dimensiones generales de, y está formado por una plancha base, dos planchas laterales y una plancha de soporte.

### **2.4.1.- PLANCHA BASE**

La plancha base tendrá la función de soportar la estructura del tanto del robot como de la cámara, además de establecer los límites del espacio de trabajo del robot. Sus dimensiones generales son 415x520x20mm, y está fabricada madera comercial aglomerada.

#### **2.4.2.- PLANCHAS LATERALES**

Las planchas laterales tendrán la función de sostener el soporte para la cámara a la altura indicada. Se colocará una a cada lateral para proporcionar mayor estabilidad a la cámara. Sus dimensiones generales son 310x60x20 mm, y está fabricada madera comercial aglomerada.

#### **2.4.3.- PLANCHA DE SOPORTE**

La plancha de soporte tendrá como objetivo sostener la cámara para que sea capaz de realizar capturas a todo el espacio de trabajo del robot. Sus dimensiones generales son 520x60x20, y está fabricada madera comercial aglomerada.

#### **2.4.4.- CONTROL DE CALIDAD**

Una vez fabricadas las piezas, se realizarán mediciones generales de estas y se compararán con las mediciones especificadas en el documento Planos. Si alguna de ellas no cumple con el diseño realizado, se descartará y fabricará de nuevo. Se realizará especial detenimiento en la comprobación de los orificios, sobre todo en los roscados, y si no cumplen con las dimensiones se descartará la pieza y se fabricará de nuevo.

Comprobadas las piezas se comprobará que encajen entre ellas según el ensamblaje mostrado en el documento Planos. Si algunas de estas piezas no encajan como es de esperar, se volverán a revisar todas las medidas con especial detenimiento y se descartará la pieza que sea responsable del mal ensamblaje.

### **3.- CONDICIONES DE EJECUCIÓN**

#### **3.1.- DESCRIPCIÓN**

Todas las piezas pertenecientes al robot se fabricarán única y exclusivamente mediante impresión 3D, las piezas relacionadas en el ensamblaje del soporte de la cámara se podrán fabricar de cualquier tipo de madera comercial siempre y cuando cumpla con las medidas de los planos.

La fabricación de cada una de las piezas se realizará en un entorno de trabajo totalmente limpio para evitar que la estructura sea dañada debido a la contaminación de los materiales.

Se ensamblará la plancha de soporte con las planchas laterales mediante 4 tornillos de métrica 5 de 30 mm de longitud autoperforantes, y en la parte superior de las planchas laterales se ensamblará interiormente la plancha de soporte con 4 tornillos de métrica 5 de 30 mm de longitud autoperforantes.

Se fijará mediante adhesivo una WebCam capaz de captar imágenes con una resolución de 1920x1080 píxeles en el centro de la plancha de soporte.

Se ensamblará la base con el hombro por medio de un rodamiento rígido de bolas 6008-2RS, y a su vez, en el hombro se colocarán en cada uno de los orificios de las varillas planas almohadillas rectangulares de 10x30 mm.

Se introducirá cada una de las varillas en los respectivos agujeros, y en el agujero de la varilla roscada se introducirá, en primer lugar, un rodamiento rígido de bolas 608 2RS, para posteriormente introducir la varilla roscada en su interior.

Se colocará el primer motor NEMA 17 modelo 17HS15-1504S-X1 en el centro de la plancha de la base, a una distancia de la cámara de 230 mm, sobre una almohadilla para evitar las vibraciones, y se dejará caer el ensamblaje del soporte del robot sobre él, introduciendo el eje del motor por el orificio de la base.

Por la parte superior del hombro, se introducirá adhesivo comercial para fijar el eje del motor al hombro del robot, y se anclará la base a la plancha base con 6 tornillos de métrica 3 y longitud 20mm, o en su defecto, se pegará con el adhesivo comercial.

Se ensamblará el brazo 1 con el motor NEMA 17 modelo 17HS08-1004S mediante 4 tornillos de métrica 3, y longitud 30mm. Se conectará el motor a un acople de 5mm a 8mm, y a este acople se añadirá la varilla roscada correspondiente al ensamblaje de los brazos. Se colocarán a presión 3 rodamientos lineales LM10UU de 30 mm de altura en cada uno de los agujeros del brazo 1.

Se ensamblará el brazo 2 a la varilla roscada simplemente roscando por el orificio de métrica 8, dejando el saliente de este brazo en la parte inferior.

Se introducirá el ensamblaje de los brazos en las varillas del ensamblaje del soporte por medio de los agujeros con los rodamientos lineales.

Se fijará una turca de latón de métrica 8 en el agujero del brazo correspondiente a la varilla roscada mediante 1 tornillo de métrica 3 y 10 mm de longitud.

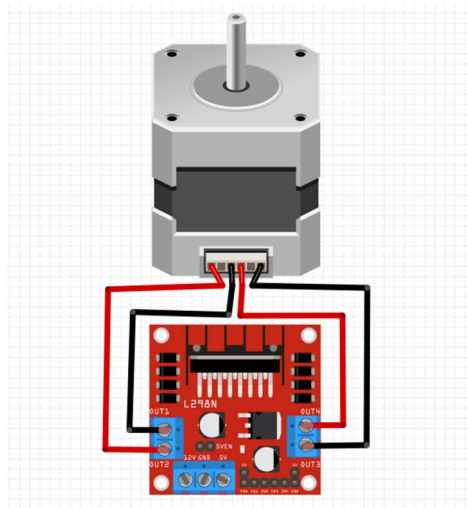
Se ensamblará la parte superior del robot con un motor NEMA 17 modelo 17HS15-1504S-X1 y se fijará mediante 2 tornillos de métrica 3 y 40mm de longitud, o en su defecto se pegará con adhesivo comercial sobre una almohadilla. A este motor se le conectará un acople de 5mm a 8mm.

En cada uno de los orificios de la parte superior se introducirá una almohadilla rectangular de dimensiones 10x30mm.

Se unirá el ensamblaje de la parte superior al ensamblaje del soporte introduciendo cada una de las varillas en su respectivo agujero, y se fijará la varilla roscada al acople del motor.

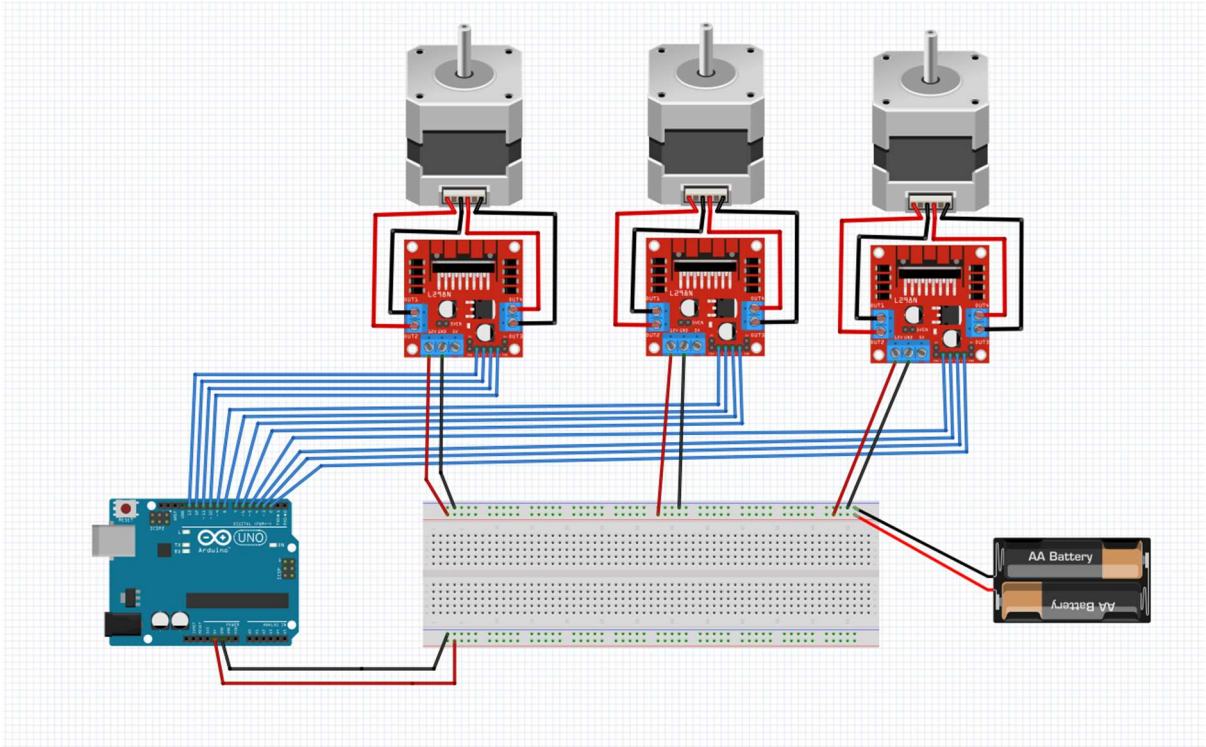
Para el montaje electrónico, se utilizarán tantos cables macho-macho y macho-hembra que sean necesarios para conectar los pines de los motores a los pines de los controladores L298n, situados junto al resto de componentes electrónicos en la plancha soporte.

El conexionado de los motores con los controladores L298n se realizará de la siguiente manera, teniendo en cuenta que las parejas de pines del motor A, A- y B, B- irán conectadas al mismo lado del controlador. Los pines se consultarán en las hojas de especificaciones técnicas de cada motor



**Figura 46:** Conexionado entre el motor y el módulo l298n

Una vez conectados los tres motores, se realizará la conexión a la placa Arduino, teniendo en cuenta que los pines conectados a los controladores coincidan con los pines asignados a cada motor en el código de Arduino, quedando de la siguiente manera.



**Figura 47:** Conexión de todos los elementos

Las baterías simularán la conexión a una fuente de alimentación externa de 12 voltios.

### 3.2.- CONTROL DE CALIDAD

Se comprobará que tras el ensamblaje se hayan empleado todas las piezas y componentes descritos en este pliego. Se realizará una inspección del conjunto del ensamblaje comparándolo con los planos y esquema electrónico para su verificación.



#### 4.- PRUEBA DE SERVICIO

Terminado el ensamblaje se realizarán una serie de pruebas para comprobar el correcto funcionamiento del conjunto no sin antes efectuar unas comprobaciones previas que permitan determinar el estado del ensamblaje para evitar comportamientos no deseados durante las pruebas.

COMPROBACIONES PREVIAS
Comprobación del correcto ajuste de los tornillos y varillas del ensamblaje
Comprobación que todos los elementos descritos en los planos hayan sido utilizados
Inspección del ensamblaje en busca de grietas o defectos generados por las presiones de apriete o del peso de los componentes
Comprobación del conexionado electrónico según el esquema descrito
Comprobación que los cables del ensamblaje no interfieran en el movimiento del robot
Comprobación que tanto la cámara como el robot se encuentren en su posición indicada

**Tabla 15:** *Comprobaciones previas a las pruebas de servicio.*

Una vez realizadas las comprobaciones, se ejecutará en primer lugar el código referido al control de la cámara, y se comprobará que la cámara toma imágenes al iniciar el programa y es capaz de reconocer los colores y situarlos en un plano.

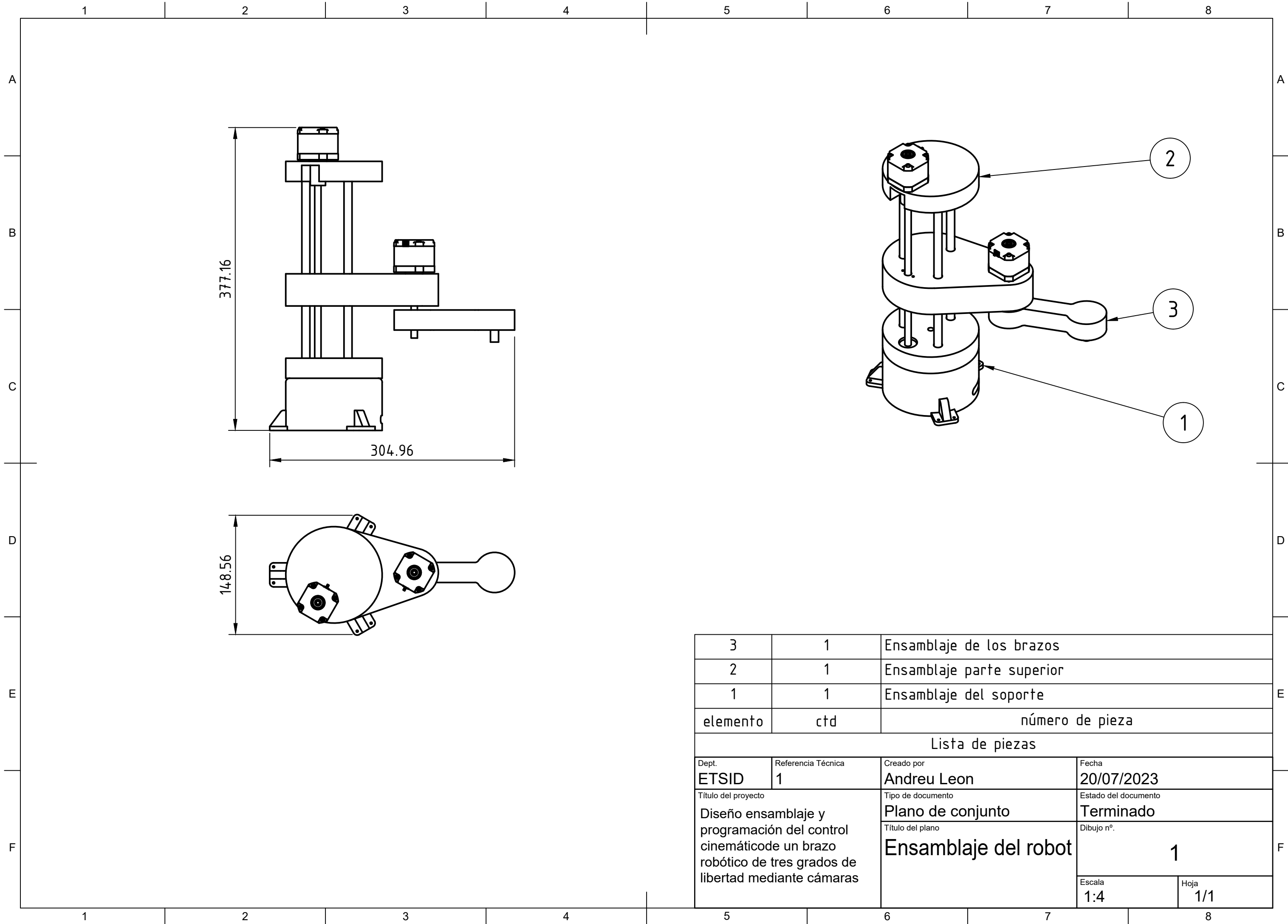
En segundo lugar, se ejecutará el código del robot, y se asignaran a mano unos valores articulares arbitrarios para comprobar el funcionamiento de los motores.

En último lugar, se ejecutará al mismo tiempo el código de la cámara y el del robot, para comprobar la correcta comunicación automática entre ellos.

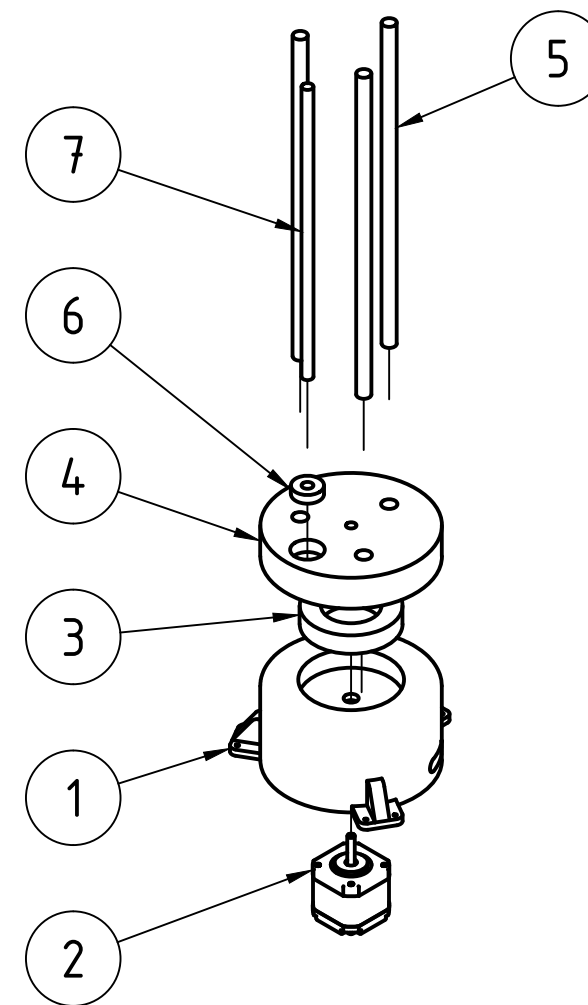
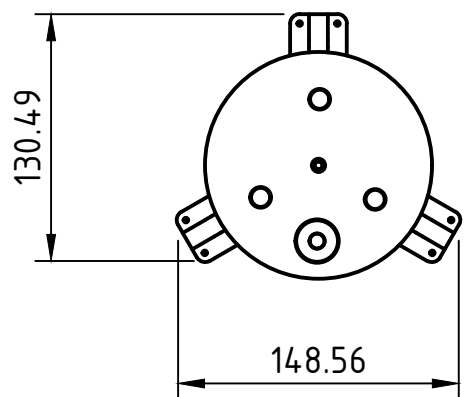
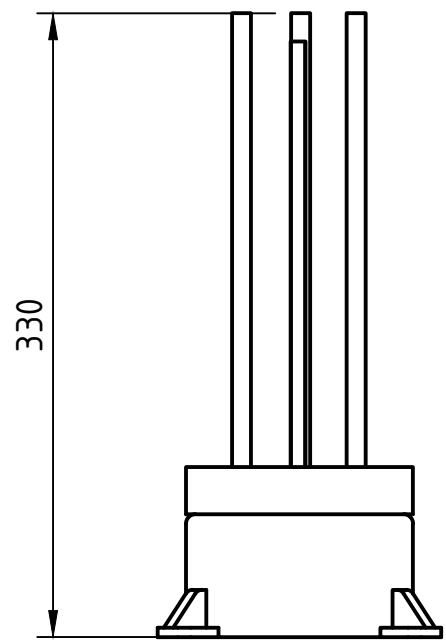
Una vez realizadas dichas pruebas, se volverán a realizar las comprobaciones de la Tabla, con la finalidad de revisar que el ensamblaje no haya sufrido ningún daño durante las pruebas.

DISEÑO, ENSAMBLAJE Y PROGRAMACIÓN DEL CONTROL CINEMÁTICO DE UN BRAZO ROBÓTICO DE TRES GRADOS DE LIBERTAD MEDIANTE CÁMARAS.

## **PLANOS**



3	1	Ensamblaje de los brazos	
2	1	Ensamblaje parte superior	
1	1	Ensamblaje del soporte	
elemento	ctd	número de pieza	
Lista de piezas			
Dept. <b>ETSID</b>	Referencia Técnica <b>1</b>	Creado por <b>Andreu Leon</b>	Fecha <b>20/07/2023</b>
Título del proyecto <b>Diseño ensamblaje y programación del control cinemático de un brazo robótico de tres grados de libertad mediante cámaras</b>		Tipo de documento <b>Plano de conjunto</b>	Estado del documento <b>Terminado</b>
		Título del plano <b>Ensamblaje del robot</b>	Dibujo n°. <b>1</b>
		Escala <b>1:4</b>	Hoja <b>1/1</b>

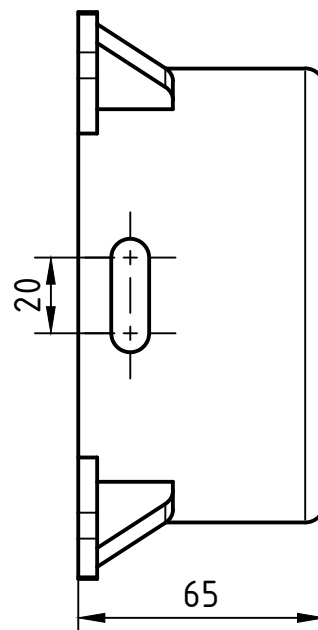
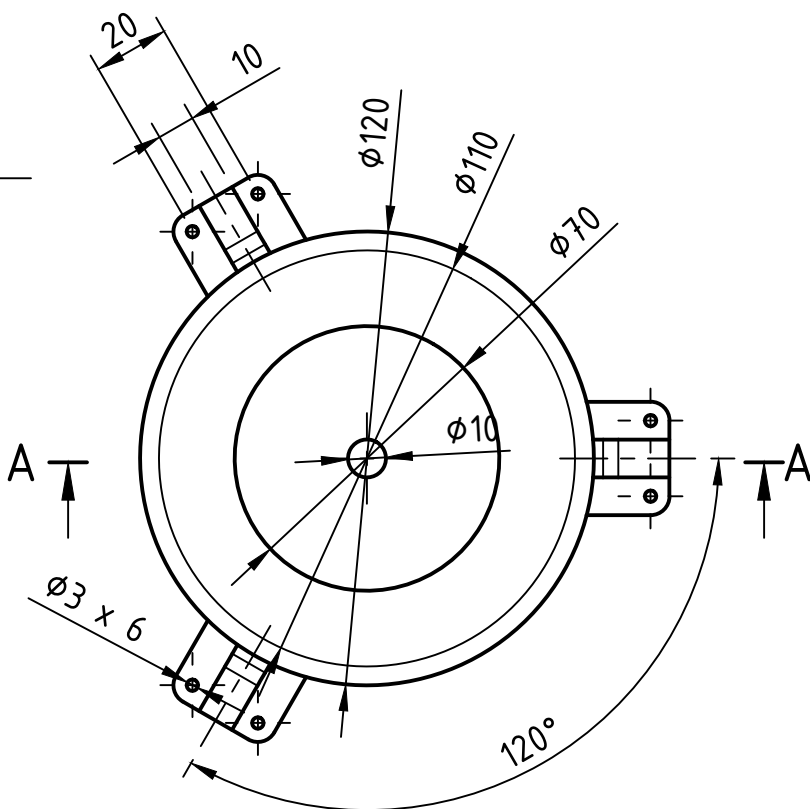
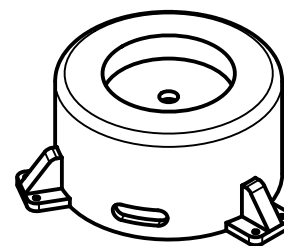
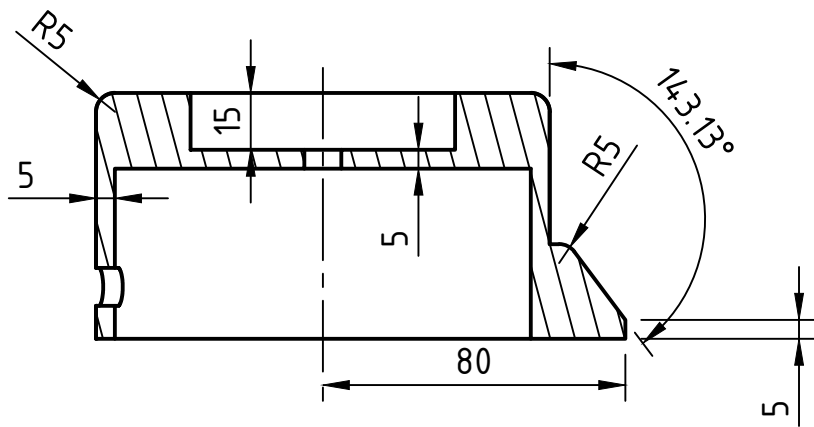


7	1	Varilla roscada $\varnothing 8 \times 235$ mm
6	1	Rodamiento 608-2RS
5	3	Varilla de acero $\varnothing 10 \times 260$ mm
4	1	Hombro
3	1	Rodamiento 6008-2RS
2	1	Motor NEMA 17 17HS15-1504S-X1
1	1	Base
elemento	ctd	número de pieza

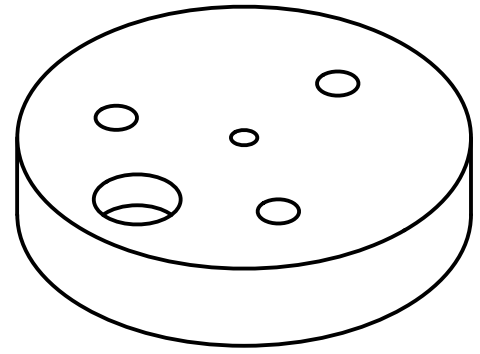
Lista de piezas

Dept. <b>ETSID</b>	Referencia Técnica <b>1.1</b>	Creado por <b>Andreu Leon</b>	Fecha <b>20/07/2023</b>
Título del proyecto <b>Diseño ensamblaje y programación del control cinemático de un brazo robótico de tres grados de libertad mediante cámaras</b>		Tipo de documento <b>Plano de conjunto</b>	Estado del documento <b>Terminado</b>
		Título del plano <b>Ensamblaje del soporte</b>	Dibujo n°. <b>2</b>
		Escala <b>1:4</b>	Hoja <b>1/1</b>

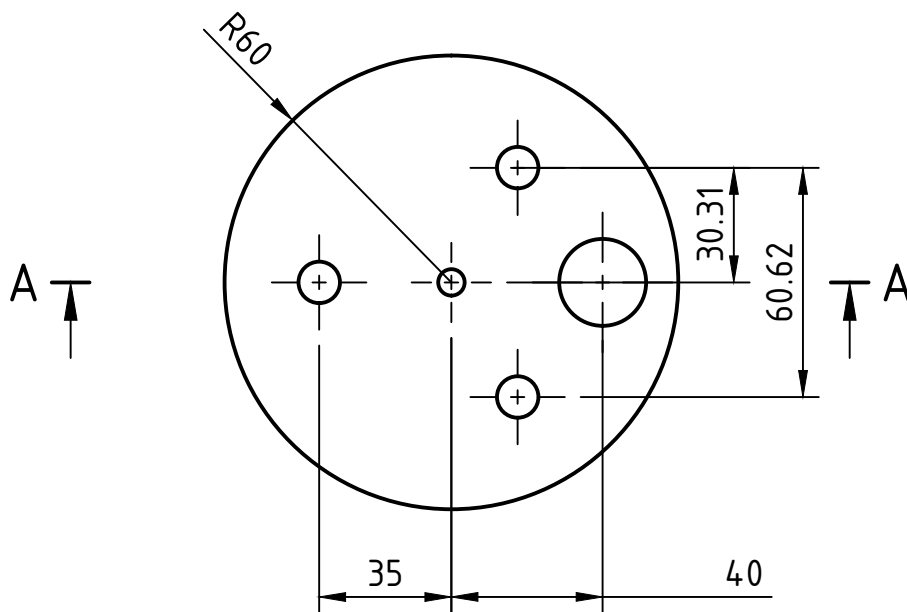
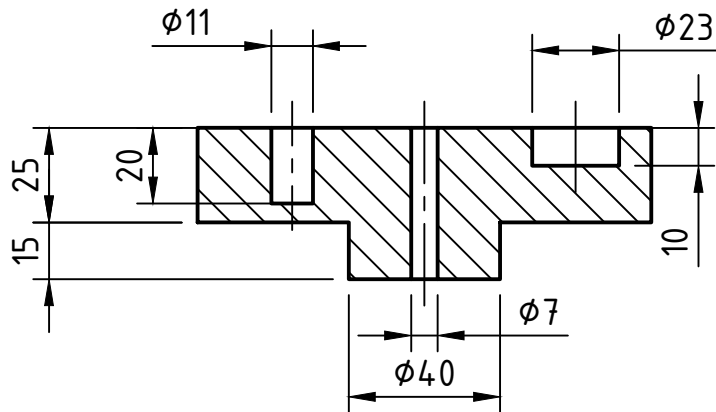
A-A (1:2)



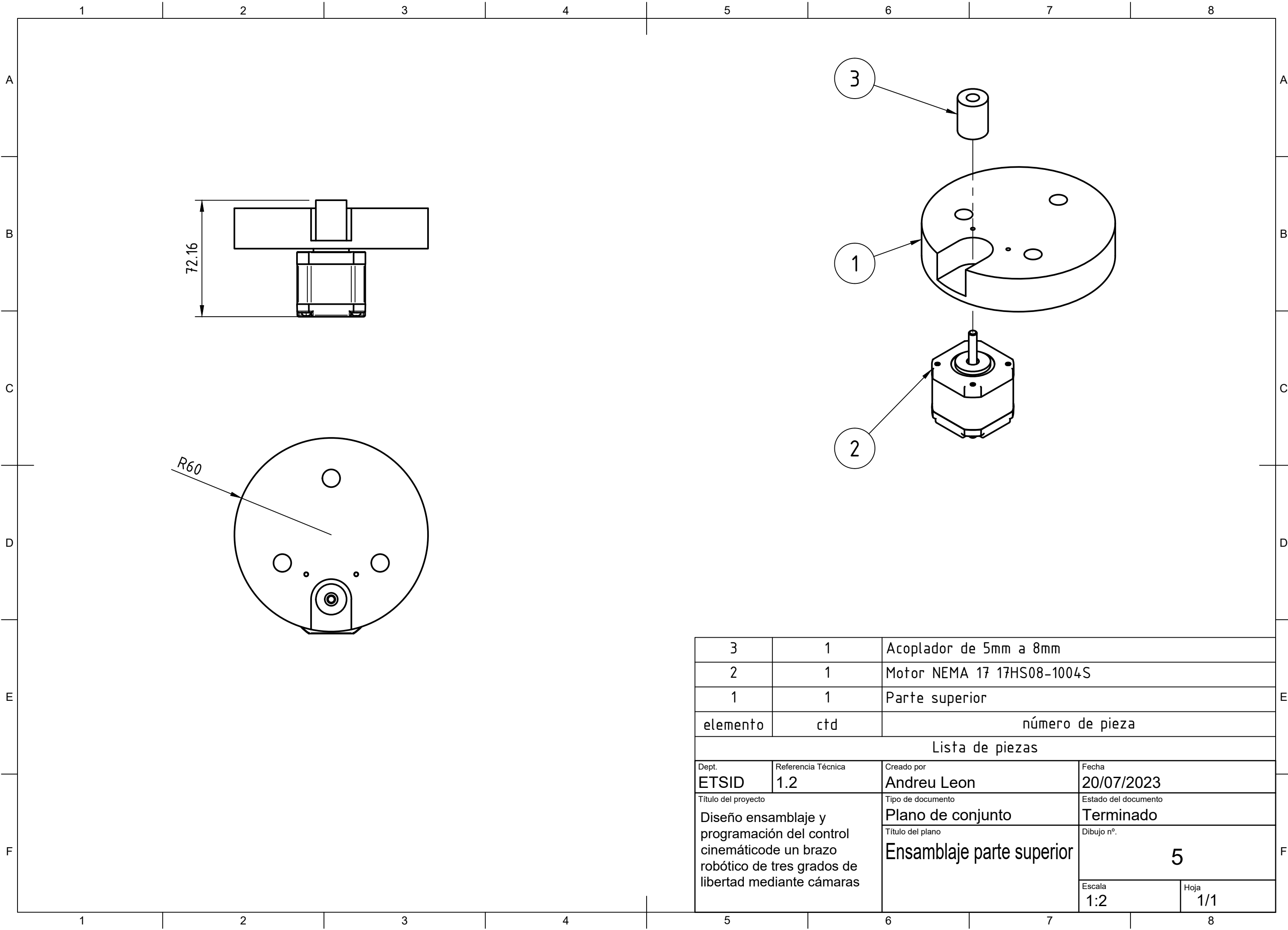
Dept. <b>ETSID</b>	Referencia Técnica <b>1.1.1</b>	Creado por <b>Andreu Leon</b>	Fecha <b>20/07/2023</b>
Título del proyecto <b>Diseño ensamblaje y programación del control cinemático de un brazo robótico de tres grados de libertad mediante cámaras</b>		Tipo de documento <b>Plano de pieza</b>	Estado del documento <b>Terminado</b>
		Título del plano <b>Base</b>	Dibujo nº <b>3</b>
		Escala <b>1:2</b>	Hoja <b>1/1</b>



A-A (1:2)

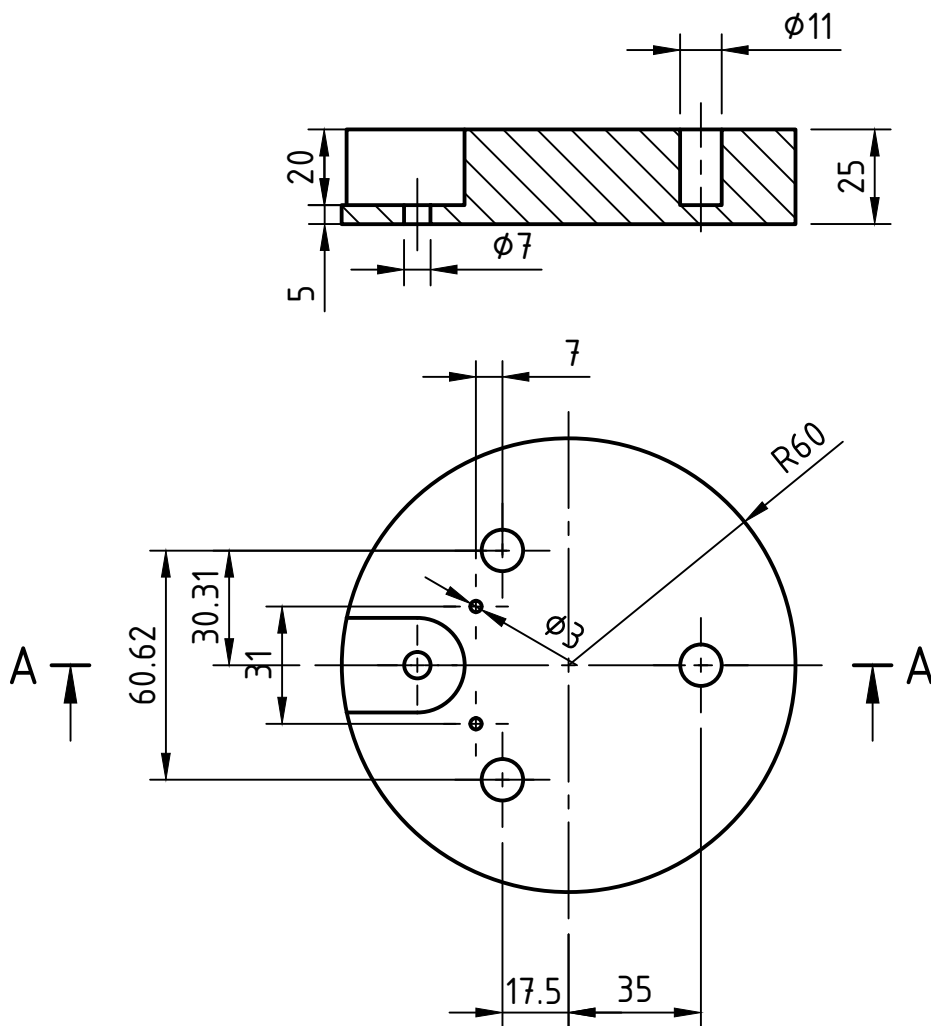


Dept. <b>ETSID</b>	Referencia Técnica <b>1.1.2</b>	Creado por <b>Andreu Leon</b>	Fecha <b>20/07/2023</b>
Título del proyecto <b>Diseño ensamblaje y programación del control cinemático de un brazo robótico de tres grados de libertad mediante cámaras</b>		Tipo de documento <b>Plano de pieza</b>	Estado del documento <b>Terminado</b>
		Título del plano <b>Hombro</b>	Dibujo nº <b>4</b>
		Escala <b>1:2</b>	Hoja <b>1/1</b>



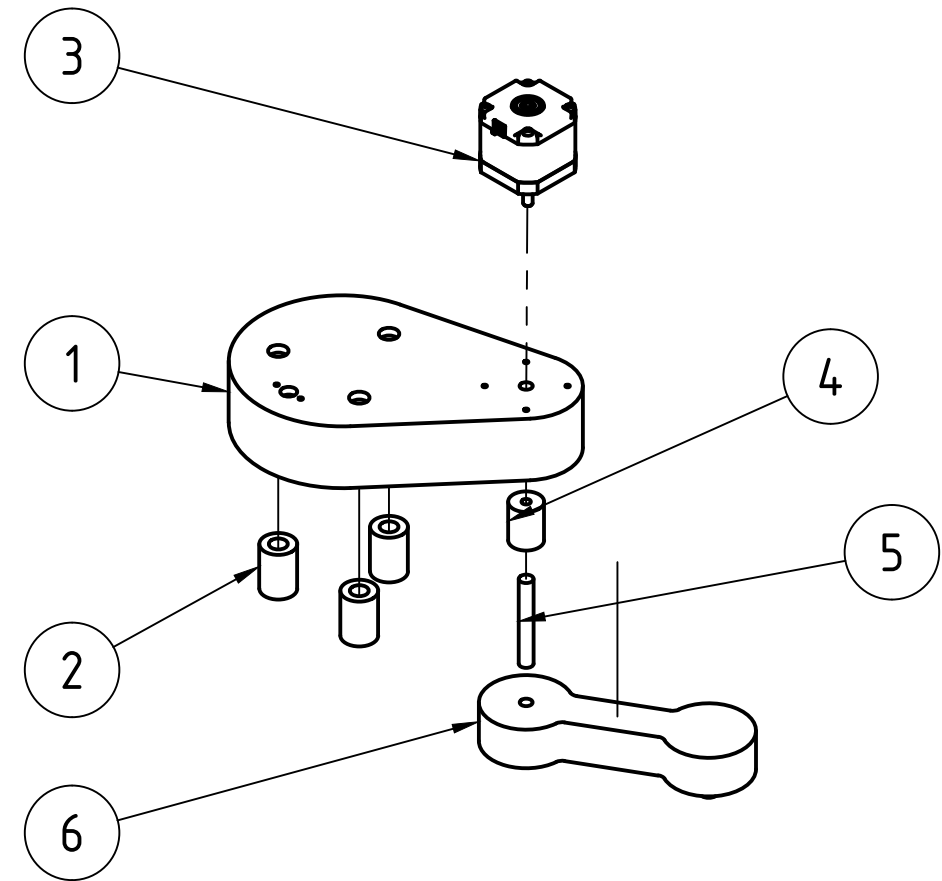
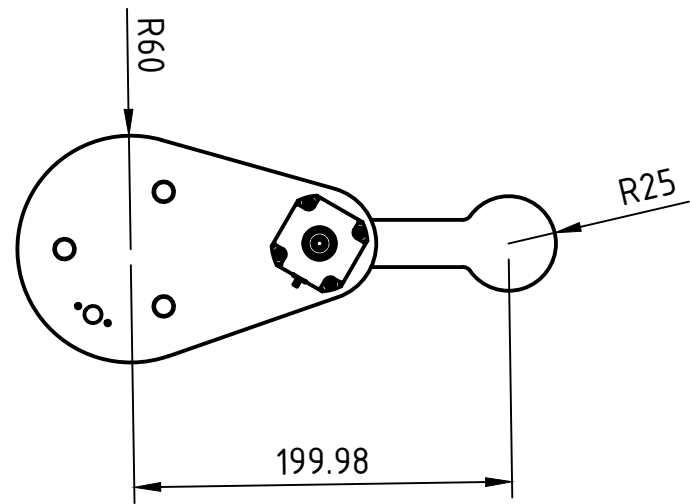
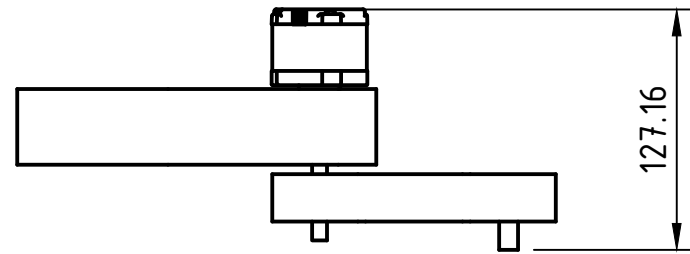
3	1	Acoplador de 5mm a 8mm	
2	1	Motor NEMA 17 17HS08-1004S	
1	1	Parte superior	
elemento	ctd	número de pieza	
Lista de piezas			
Dept. <b>ETSID</b>	Referencia Técnica <b>1.2</b>	Creado por <b>Andreu Leon</b>	Fecha <b>20/07/2023</b>
Título del proyecto <b>Diseño ensamblaje y programación del control cinemático de un brazo robótico de tres grados de libertad mediante cámaras</b>		Tipo de documento <b>Plano de conjunto</b>	Estado del documento <b>Terminado</b>
		Título del plano <b>Ensamblaje parte superior</b>	Dibujo nº. <b>5</b>
		Escala <b>1:2</b>	Hoja <b>1/1</b>

### A-A (1:2)



Dept. <b>ETSID</b>	Referencia Técnica <b>1.2.1</b>	Creado por <b>Andreu Leon</b>	Fecha <b>20/07/2023</b>
Título del proyecto <b>Diseño ensamblaje y programación del control cinemático de un brazo robótico de tres grados de libertad mediante cámaras</b>		Tipo de documento <b>Plano de pieza</b>	Estado del documento <b>Terminado</b>
		Título del plano <b>Parte superior</b>	Dibujo nº <b>6</b>
		Escala <b>1:2</b>	Hoja <b>1/1</b>



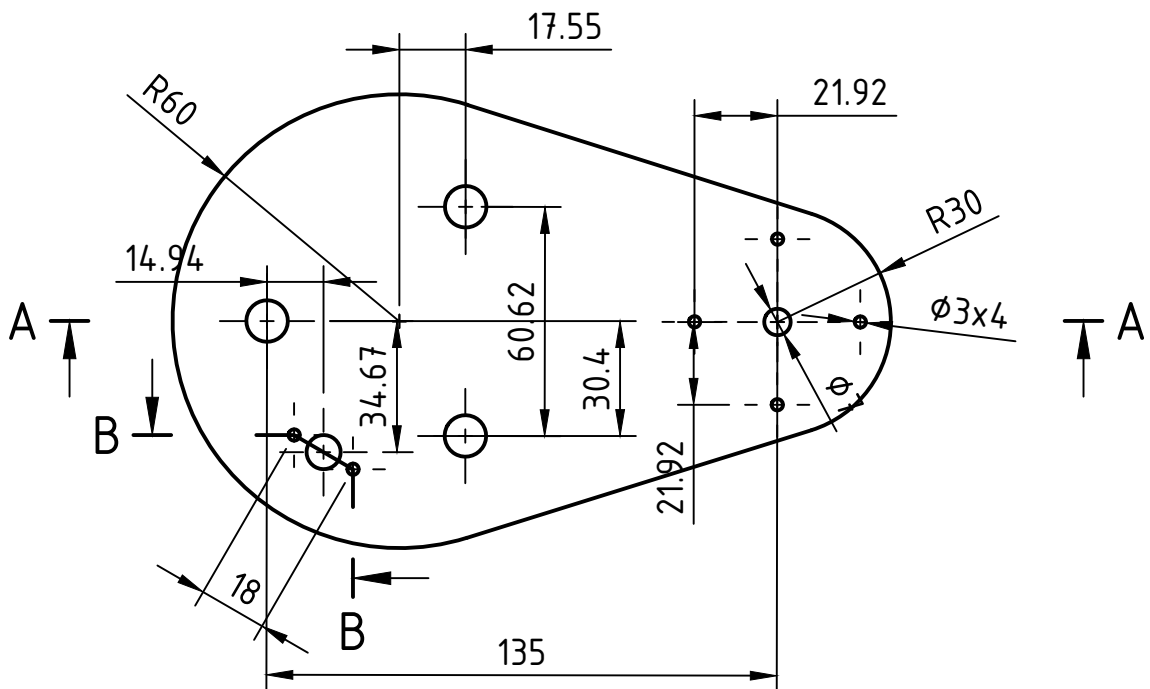
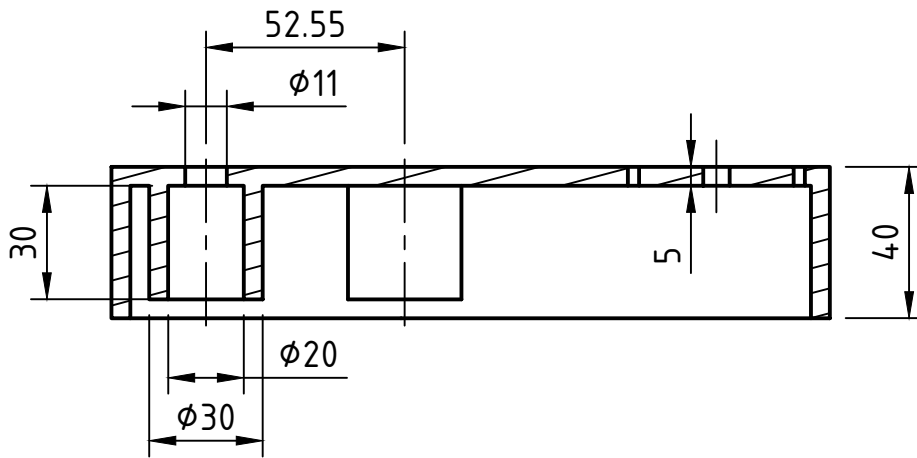
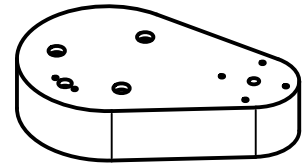


6	1	Brazo 2
5	1	Varilla roscada $\varnothing 8 \times 55$ mm
4	1	Acoplador de 5mm a 8mm
3	1	Motor NEMA 17 17HS08-1004S
2	3	Rodamiento LM10UU
1	1	Brazo 1
elemento	ctd	número de pieza

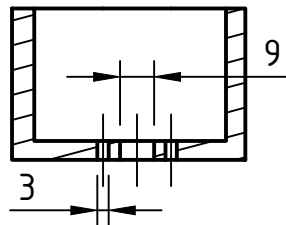
Lista de piezas

Dept. <b>ETSID</b>	Referencia Técnica <b>1.3</b>	Creado por <b>Andreu Leon</b>	Fecha <b>20/07/2023</b>
Título del proyecto <b>Diseño ensamblaje y programación del control cinemático de un brazo robótico de tres grados de libertad mediante cámaras</b>		Tipo de documento <b>Plano de conjunto</b>	Estado del documento <b>Terminado</b>
		Título del plano <b>Ensamblaje de los brazos</b>	Dibujo nº. <b>7</b>
		Escala <b>1:4</b>	Hoja <b>1/1</b>

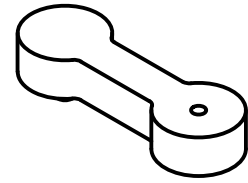
A-A (1:2)



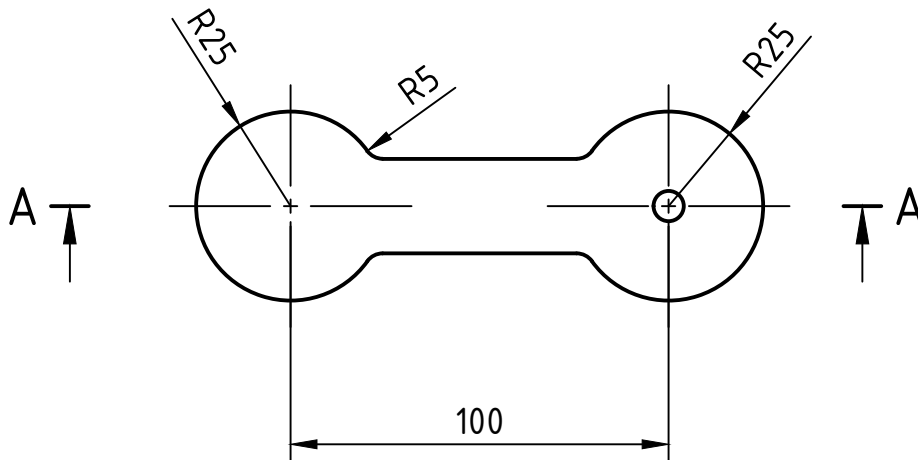
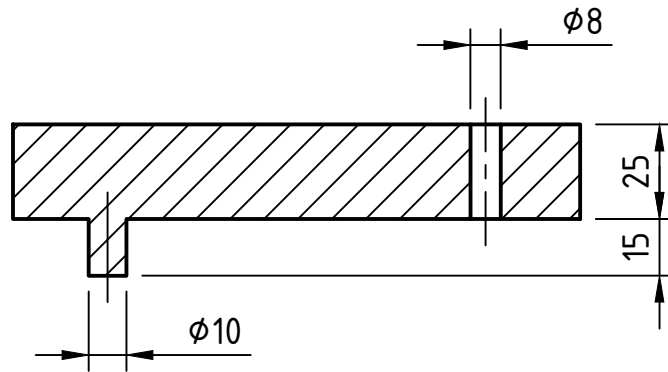
B-B (1:2)



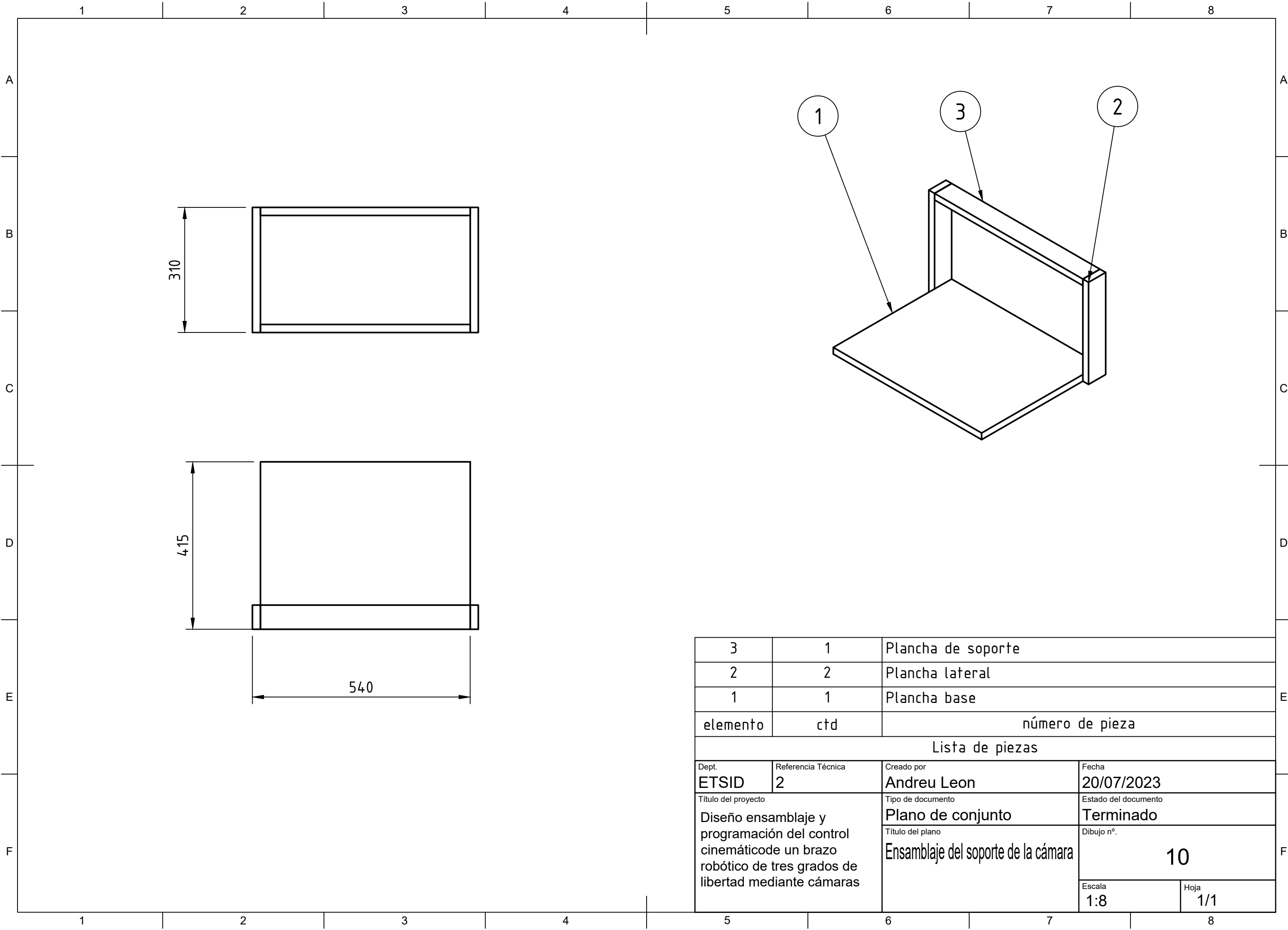
Dept. <b>ETSID</b>	Referencia Técnica <b>1.3.1</b>	Creado por <b>Andreu Leon</b>	Fecha <b>20/07/2023</b>
Título del proyecto <b>Diseño ensamblaje y programación del control cinemático de un brazo robótico de tres grados de libertad mediante cámaras</b>		Tipo de documento <b>Plano de pieza</b>	Estado del documento <b>Terminado</b>
		Título del plano <b>Brazo 1</b>	Dibujo nº <b>8</b>
		Escala <b>1:2</b>	Hoja <b>1/1</b>



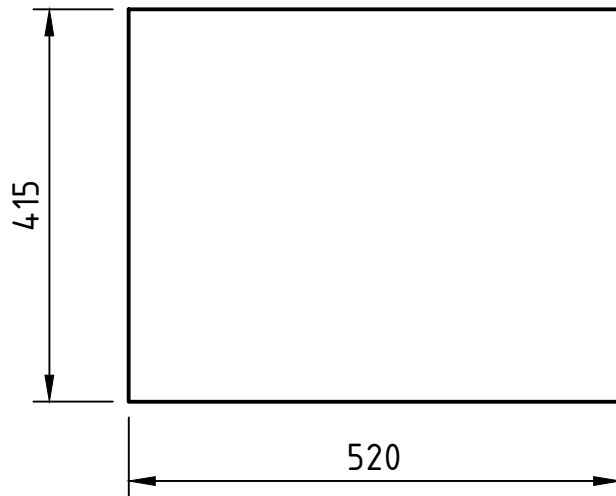
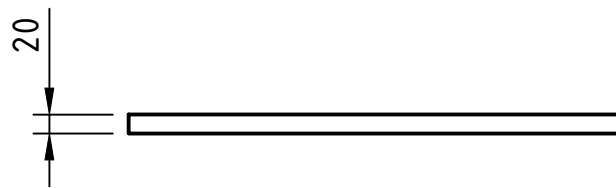
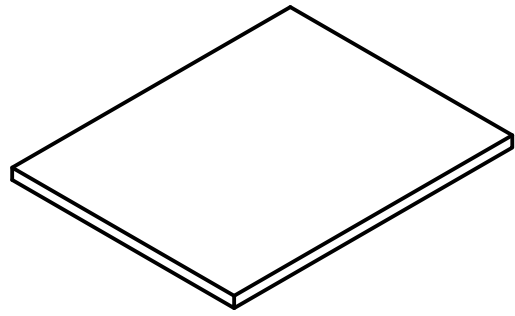
A-A (1:2)



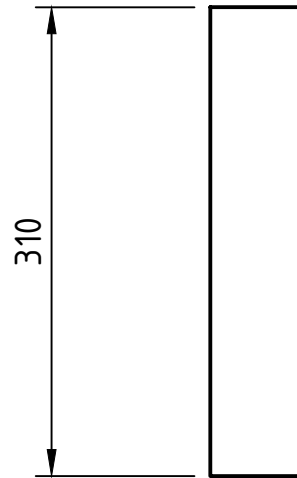
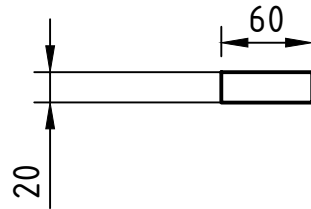
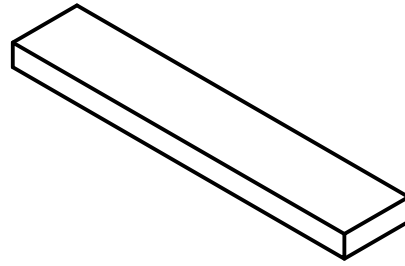
Dept. <b>ETSID</b>	Referencia Técnica <b>1.3.2</b>	Creado por <b>Andreu Leon</b>	Fecha <b>20/07/2023</b>
Título del proyecto <b>Diseño ensamblaje y programación del control cinemático de un brazo robótico de tres grados de libertad mediante cámaras</b>		Tipo de documento <b>Plano de pieza</b>	Estado del documento <b>Terminado</b>
		Título del plano <b>Brazo 2</b>	Dibujo nº <b>9</b>
		Escala <b>1:2</b>	Hoja <b>1/1</b>



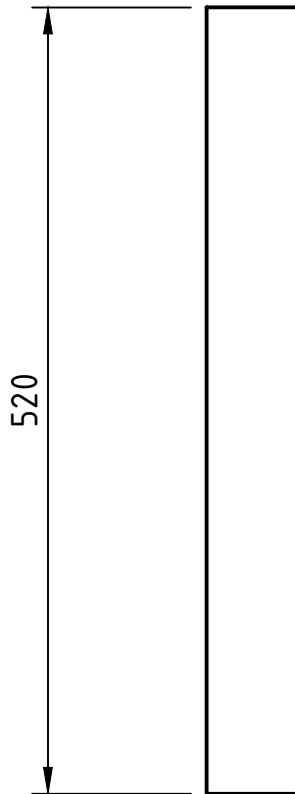
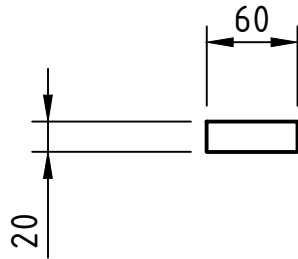
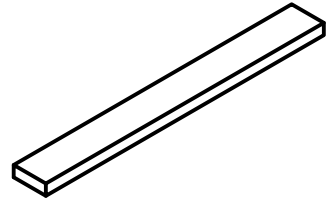
3	1	Plancha de soporte	
2	2	Plancha lateral	
1	1	Plancha base	
elemento	ctd	número de pieza	
Lista de piezas			
Dept. <b>ETSID</b>	Referencia Técnica <b>2</b>	Creado por <b>Andreu Leon</b>	Fecha <b>20/07/2023</b>
Título del proyecto <b>Diseño ensamblaje y programación del control cinemático de un brazo robótico de tres grados de libertad mediante cámaras</b>		Tipo de documento <b>Plano de conjunto</b>	Estado del documento <b>Terminado</b>
		Título del plano <b>Ensamblaje del soporte de la cámara</b>	Dibujo nº. <b>10</b>
		Escala <b>1:8</b>	Hoja <b>1/1</b>



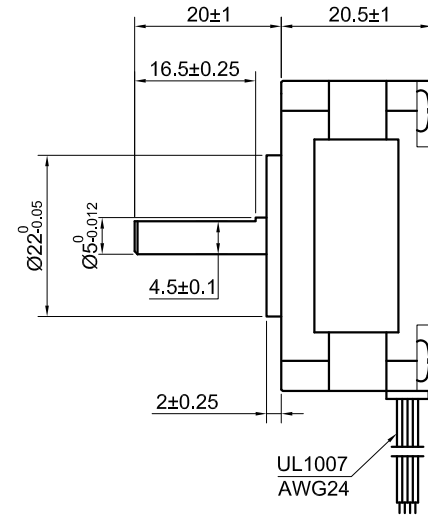
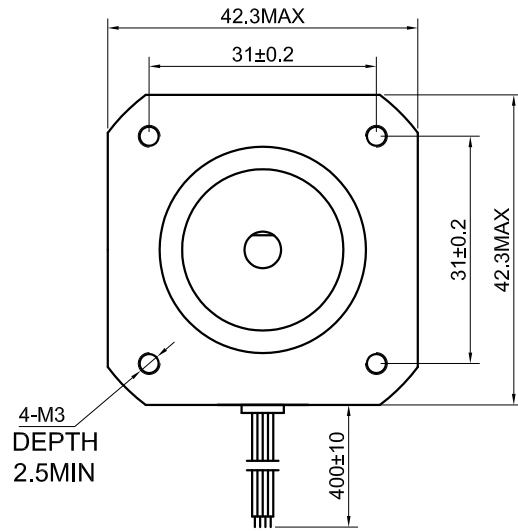
Dept. <b>ETSID</b>	Referencia Técnica <b>2.1</b>	Creado por <b>Andreu Leon</b>	Fecha <b>20/07/2023</b>
Titulo del proyecto <b>Diseño ensamblaje y programación del control cinemático de un brazo robótico de tres grados de libertad mediante cámaras</b>		Tipo de documento <b>Plano de pieza</b>	Estado del documento <b>Terminado</b>
		Título del plano <b>Plancha base</b>	Dibujo nº <b>11</b>
		Escala <b>1:8</b>	Hoja <b>1/1</b>



Dept. <b>ETSID</b>	Referencia Técnica <b>2.2</b>	Creado por <b>Andreu Leon</b>	Fecha <b>20/07/2023</b>
Titulo del proyecto <b>Diseño ensamblaje y programación del control cinemático de un brazo robótico de tres grados de libertad mediante cámaras</b>	Tipo de documento <b>Plano de pieza</b>		Estado del documento <b>Terminado</b>
	Titulo del plano <b>Plancha lateral</b>		Dibujo nº <b>12</b>
	Escala <b>1:5</b>		Hoja <b>1/1</b>



Dept. <b>ETSID</b>	Referencia Técnica <b>2.3</b>	Creado por <b>Andreu Leon</b>	Fecha <b>20/07/2023</b>
Titulo del proyecto <b>Diseño ensamblaje y programación del control cinemático de un brazo robótico de tres grados de libertad mediante cámaras</b>	Tipo de documento <b>Plano de pieza</b>	Estado del documento <b>Terminado</b>	
	Título del plano <b>Plancha de soporte</b>	Dibujo nº <b>13</b>	
		Escala <b>1:5</b>	Hoja <b>1/1</b>

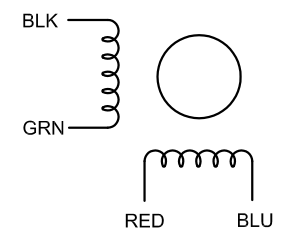


SPECIFICATION	CONNECTION	BIPOLAR
AMPS/PHASE		1.00
RESISTANCE/PHASE(Ohms)@25°C		3.70±10%
INDUCTANCE/PHASE(mH)@1KHz		4.50±20%
HOLDING TORQUE(Nm)[lb-in]		0.16[1.42]
STEP ANGLE(°)		1.80
STEP ACCURACY(NON-ACCUM)		±5.00%
ROTOR INERTIA(g-cm <sup>2</sup> )		22.00
WEIGHT(Kg)[lb]		0.14[0.31]
TEMPERATURE RISE:MAX.80°C (MOTOR STANDSTILL;FOR 2PHASE ENERGIZED)		
AMBIENT TEMPERATURE -10°C~50°C[14°F~122°F]		
INSULATION RESISTANCE 100 Mohm (UNDER NORMAL TEMPERATURE AND HUMIDITY)		
INSULATION CLASS B 130°C[266°F]		
DIELECTRIC STRENGTH 500VAC FOR 1MIN.(BETWEEN THE MOTOR COILS AND THE MOTOR CASE)		
AMBIENT HUMIDITY MAX.85%(NO CONDENSATION)		

TYPE OF CONNECTION (EXTERN)		MOTOR	
PIN NO	BIPOLAR	LEADS	WINDING
1	A —	BLK	A
2	A\ —	GRN	A\
3	B —	RED	B
4	B\ —	BLU	B\

FULL STEP 2 PHASE-Ex ,  
WHEN FACING MOUNTING END (X)

STEP	A	B	A\	B\		CCW
1	+	+	-	-	↓ ↑	↑ ↓
2	-	+	+	-		
3	-	-	+	+		
4	+	-	-	+		

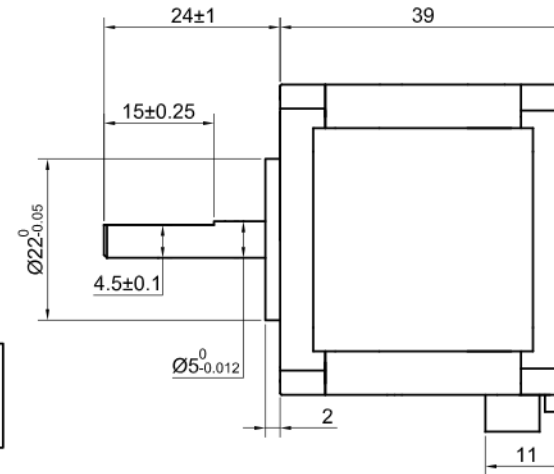
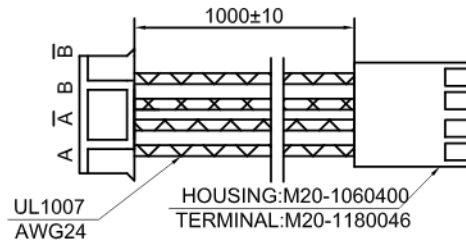
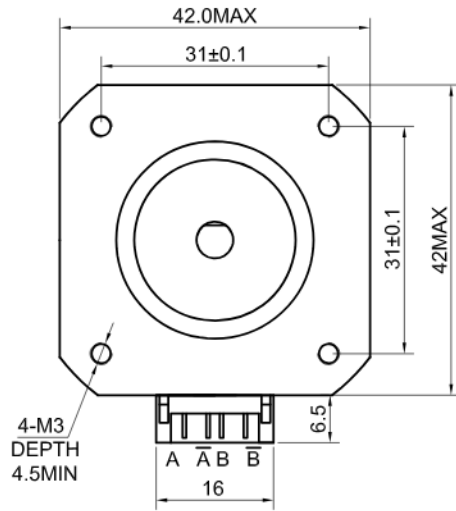


APVD		8.18.2018
CHKD		
1:1	DRN	
SCALE	SIGNATURE	DATE

STEPPER MOTOR

17HS08-1004S



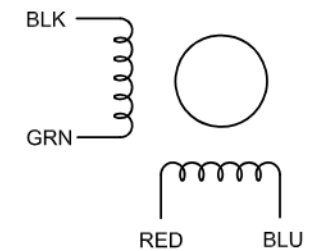


SPECIFICATION	CONNECTION	BIPOLAR
AMPS/PHASE		1.50
RESISTANCE/PHASE(Ohms)@25°C		2.30±10%
INDUCTANCE/PHASE(mH)@1KHz		4.40±20%
HOLDING TORQUE(Nm)[lb-in]		0.45[3.98]
STEP ANGLE(°)		1.80
STEP ACCURACY(NON-ACCUM)		±5.00%
ROTOR INERTIA(g-cm <sup>2</sup> )		54.00
WEIGHT(Kg)[lb]		0.28[0.62]
TEMPERATURE RISE:MAX.80°C (MOTOR STANDSTILL;FOR 2PHASE ENERGIZED)		
AMBIENT TEMPERATURE -10°C~50°C[14°F~122°F]		
INSULATION RESISTANCE 100 Mohm (UNDER NORMAL TEMPERATURE AND HUMIDITY)		
INSULATION CLASS B 130°C[266°F]		
DIELECTRIC STRENGTH 500VAC FOR 1MIN.(BETWEEN THE MOTOR COILS AND THE MOTOR CASE)		
AMBIENT HUMIDITY MAX.85%(NO CONDENSATION)		

TYPE OF CONNECTION (EXTERN)		MOTOR	
PIN NO	BIPOLAR	LEADS	WINDING
1	A —	BLK	A
2	A' —	GRN	A'
3	B —	RED	B
4	B' —	BLU	B'

FULL STEP 2 PHASE-Ex. ,  
WHEN FACING MOUNTING END (X)

STEP	A	B	A'	B'		CCW
1	+	+	-	-	↓	↑
2	-	+	+	-		
3	-	-	+	+	↑	↓
4	+	-	-	+		



**STEPPERONLINE®**

	APVD		
	CHKD		
1:1	DRN		
SCALE	SIGNATURE	DATE	

**STEPPER MOTOR**

17HS15-1504S-X1

