

Artículo

Desarrollo de un Robot Manipulador de 3GDL Cilíndrico con Interfaz Gráfica de Usuario para Seguimiento de Trayectorias

López-García J. ¹, Cruz-Perales F. ², González-González R. ³, Chávez-Gómez A. M. ⁴, Hernández-Paredes J. M. ^{5*} y Muñoz-Barrón B. ⁶

^{1,2,3,4} Estudiante de Ingeniería Mecatrónica, Tecnológico Nacional de México / Instituto Tecnológico Superior de Huichapan, Hidalgo, México.

^{5,6} División de Ingeniería Mecatrónica/Maestría en Ingeniería Mecatrónica, Tecnológico Nacional de México / Instituto Tecnológico Superior de Huichapan, Hidalgo, México.

* Correspondencia: jmhernandez@iteshu.edu.mx.

Resumen: Este artículo presenta el desarrollo integral de un brazo robótico de 3 grados de libertad, el cual abarca desde el diseño mecánico, la manufactura, modelado matemático, simulación, control e implementación en el mundo físico. El diseño se basa en un análisis de los requisitos y especificaciones del brazo, seguido de la construcción con el método de manufactura aditiva. La instrumentación incluye la implementación de sensores para medir la posición articular del brazo la cual proporciona retroalimentación a el controlador de posición y conexión a dispositivos de potencia. Se utilizan algoritmos y sistemas de control para lograr movimientos en el espacio cartesiano. Además, se representan los algoritmos de programación que utilizan los modelos matemáticos que describen el movimiento del robot. Finalmente, se realizan dos pruebas de seguimiento de trayectorias para validar la construcción del robot en las cuales arroja un error del ± 0.1 rad en valor articular y ± 0.1 mm en valor cartesiano, este análisis se realiza en una interfaz gráfica de usuario en tiempo real.

López-García J., Cruz-Perales F., González-González R., Chávez-Gómez A. M., Hernández-Paredes J. M. y Muñoz-Barrón B. Desarrollo de un Robot Manipulador de 3GDL Cilíndrico con 2 Interfaz Gráfica de Usuario para Seguimiento de Trayectorias

Keywords: *Cinemática; Control; Trayectoria, Robot.*

REÍA 2023, 7, (2), 11-37

Recibido: 24/11/2023

Aceptado: 22/12/2023

Publicado: 31/12/2023

1. Introducción

La robótica tiene su origen por la inquietud a lo largo de los tiempos por crear reproducciones del ser humano y de otros seres vivos, pero ante todo por la necesidad de resolver y mejorar los procesos mecánicos del hombre [1]. Así como lo menciona McCloy en 1984 "La tecnología amplía el potencial humano" [2], siendo así la creación de robots cada vez más automatizados para que realicen las tareas que al ser humano le representan peligro o que requieran mayor producción en la industria. Los robots son máquinas en las que se integran componentes mecánicos, eléctricos, electrónicos y de comunicaciones, y dotadas de un sistema informático para su control en tiempo real, percepción del entorno y programación. La mayor parte de los robots industriales actuales son esencialmente brazos articulados, de hecho, según la definición del "Robot Institute of America", un robot industrial es un manipulador programable multifuncional diseñado para mover materiales, piezas, herramientas o dispositivos especiales, mediante movimientos variados, programados para la ejecución de distintas tareas [3]. Los robots se caracterizan por sus grados de libertad que son el número de movimientos independientes que puede realizar cada articulación con respecto a la anterior por lo tanto el número de grados de libertad de un robot viene dado por la suma de los grados de libertad de cada una de sus propias articulaciones [4]. Los robots tienen diferentes configuraciones, una de ellas es el robot cilíndrico que se caracteriza por remplazar una articulación de rotación en la primer

o segunda articulación de un robot cartesiano, donde el centro de la muñeca se describe mediante coordenadas cilíndricas [5].

Respecto a las aplicaciones de los robots en los últimos años, el desarrollo de la robótica experimentó avances significativos, que brindan nuevas oportunidades en diversas áreas de aplicación. Uno de los aspectos clave en este campo es el diseño y desarrollo de brazos robóticos capaces de realizar una amplia gama de tareas con precisión y eficiencia. Los brazos robóticos de múltiples grados de libertad se convirtieron en una solución versátil y altamente demandada en entornos industriales [6], médicos [7] y de investigación [8]. Como se menciona anteriormente los brazos robóticos o manipuladores se emplean para diferentes áreas, desde fines educativos en las universidades que abarca el desarrollo mecánico, hasta sus modelos matemáticos es decir la cinemática directa e inversa, además de su dinámica con el objetivo de poder controlarlos de tal forma que se adentren al mundo de la robótica [9].

Regresando a los robots cilíndricos se desarrollaron variedades de estos desde solo el diseño y la simulación hasta el desarrollo completo de estos mismo, por ejemplo [10] diseño un robot cilíndrico de 6 GDL y elaboro pruebas de estado finito para observar que tan optimo era el diseño con la finalidad de que pueda soportar cargas frontales de colisión frontal. Cabe mencionar que los robots también se pueden interconectar con más sistemas, este es el caso de [11] donde se desarrolla un brazo robótico cilíndrico para un sistema de clasificación de objetos de forma geométrica con la ayuda del reconocimiento de imagines por medio de una cámara de video. Cabe mencionar que también se estudia su modelo matemático con la finalidad de realizar simulaciones antes de su desarrollo [12] [13] [14] [15], con la finalidad de llegar a la industria y automatizar los procesos y tener una mayor respuesta en cuanto la demanda de tal manera donde ya se utiliza equipo industrial [16] [17].

En el presente artículo, se muestra el desarrollo integral de un brazo robótico de 3 grados de libertad, en la sección dos se muestra los Materiales y Métodos que abarca desde el diseño 3D, manufactura, instrumentación, modelos matemáticos hasta el control y la implementación del seguimiento de trayectorias, en la sección 3 y 4 se habla sobre los Resultados y Discusiones sobre los resultados obtenidos mediante el desarrollo de una interfaz donde se genera el seguimiento de las trayectorias, por consiguiente en la sección 5 se presentan las Conclusiones más relevantes en el desarrollo de este trabajo y por último todas las fuentes de información que se consultaron.

2. Materiales y Métodos

El desarrollo del brazo robótico cilíndrico de 3 GDL se realiza en diferentes atapas como se muestra en el Fig. 1 con la finalidad de dividir todas las áreas de trabajo.

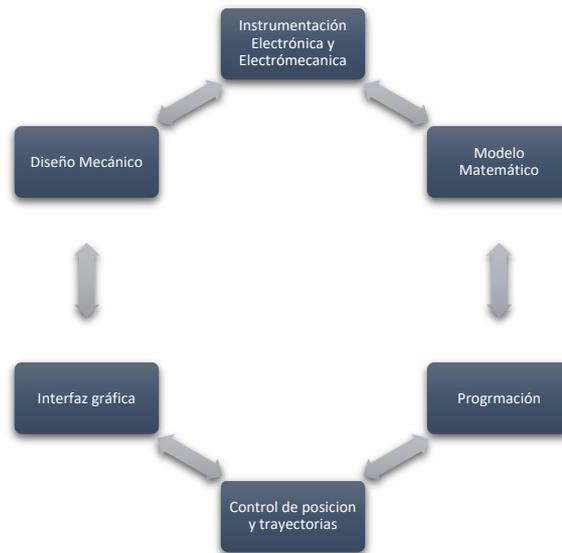


Fig. 1. Etapas para el desarrollo del brazo robótico cilíndrico de 3 GDL.

En diseño de este prototipo se basó en la metodología de diseño mecatrónico de robots industriales [18] que se muestra en el Fig. 2, donde se integra el sistema mecánico, electrónico, control y programación para obtener un diseño completamente mecatrónico.

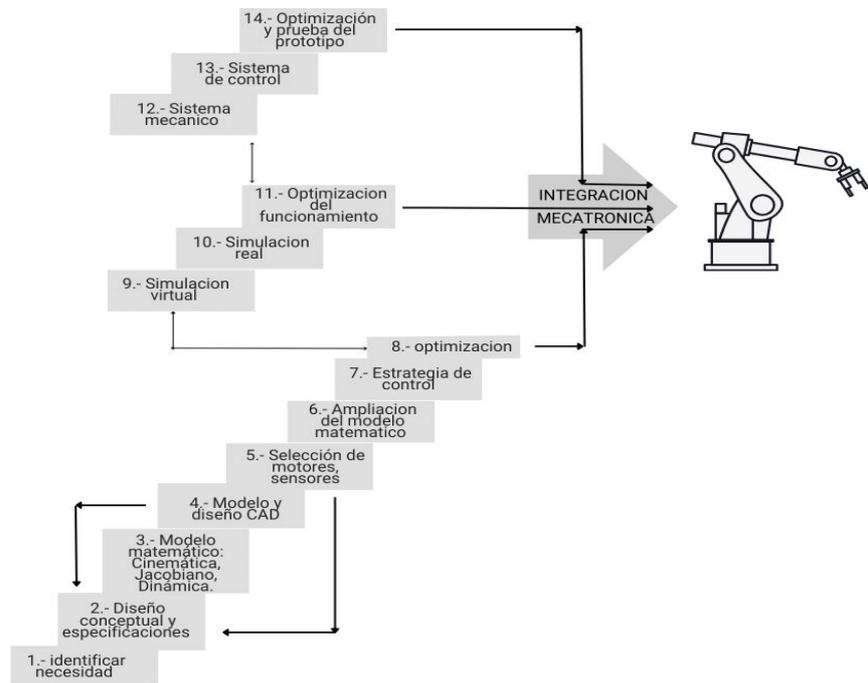


Fig. 2. Metodología de diseño que se siguió para el desarrollo del brazo robótico.

2.1. Diseño Mecánico

El diseño mecánico se parte en 3 partes como muestra el Fig. 3, definición de materiales, CAD y finalmente concluye con la manufactura y ensamble en físico.

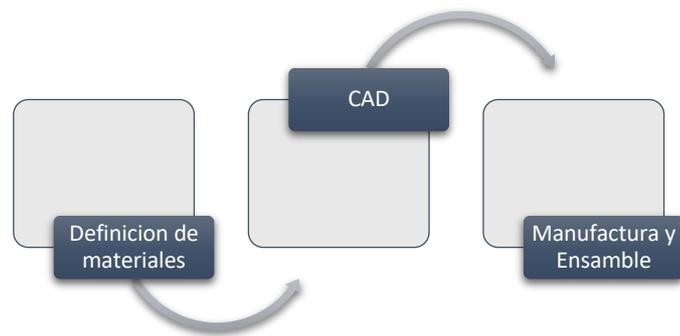


Fig. 3. Etapas del diseño Mecánico,

2.1.1. Definición de materiales

Por especificaciones técnicas como la determinación precisa de la posición del robot en cualquier momento, la resolución de pulsos por giro para un error mínimo en las trayectorias se utilizó como actuadores motores de corriente continua (CD) con encoders con las siguientes especificaciones.

Motor CD:

- Voltaje recomendado: 24V
- Relación de Reducción: 1:34
- Velocidad sin carga: 180 rpm (a 24V)
- Corriente sin carga: ≤ 100 mA (a 24V)
- Velocidad Nominal: 110 rpm
- Corriente Nominal: ≤ 01.3 A
- Torque / Par nominal: 3 kg.cm (0.29 Nm)
- Corriente de bloqueo: ≤ 2.5 A
- Par de bloqueo: 6.0 kg.cm
- Resolución del encoder: 341.2 PPR

Encoder:

- Tipo de encoder: Codificador Hall Magnético Incremental de doble fase
- Voltaje de alimentación: 3.3V – 5V
- Interfaz: PH20 (cable estándar)
- Numero básico de pulsos 11ppr
- Frecuencia de respuesta 100KHz

Para tener un controlador que lleve al sistema a las referencias deseadas en un solo microcontrolador sin la necesidad de crear una interfaz de comunicación ya sea SPI, I2C entre otras se eligió controlar con un microcontrolador ATMEGA 250 ya que la capacidad de procesamiento necesaria para ejecutar las tareas de instrumentación y control. A continuación, se muestran sus características técnicas más importantes:

- Microcontrolador: ATmega2560
- Voltaje de funcionamiento: 5 V
- Pines de E/S digitales: 54 (14 utilizables para salida PWM)
- Pines de entrada analógica: 16
- 16 MIPS
- 6 interrupciones externas

El robot presenta la necesidad de controlar la dirección de giro de sus motores en sentido horario y antihorario. Para abordar esta funcionalidad, se seleccionó la implementación de puentes H L298N. Esta elección se basa en la capacidad inherente para gestionar la polaridad de la tensión aplicada a los motores, lo que permite un control del movimiento del robot, además, ofrece la ventaja de incorporar la modulación por ancho

de pulso (PWM) para regular de forma gradual la velocidad de los motores. Esta funcionalidad de control permite ajustar la velocidad de los motores según las necesidades específicas de la aplicación, lo que resulta fundamental para lograr un control suave de la velocidad del robot en diversas situaciones.

- Integrado: L298N
- Voltaje control: 5 V
- Corriente control: 36 mA
- Voltaje operación: 5 V
- Corriente operación: 2 A
- Potencia: 25 W
- Temperatura: -20 ~ 135 °C

2.1.2. Diseño mecánico CAD

La Fig. 4 representa un robot que se encuadra dentro de la categoría de los robots cilíndricos. Esta clasificación se debe a la presencia de una articulación de rotación (denominada Art. 1) y dos articulaciones prismáticas (identificadas como Art. 2 y Art. 3). Estas articulaciones prismáticas se construyen al utilizar tornillos sin fin que se fabrican en acero inoxidable.

El robot cilíndrico, como su nombre lo sugiere, se caracteriza por su estructura y movimientos que recuerdan la forma de un cilindro. La articulación de rotación permite al robot realizar movimientos de rotación alrededor de un eje específico, mientras que las articulaciones prismáticas facilitan desplazamientos lineales a lo largo de sus ejes respectivos.

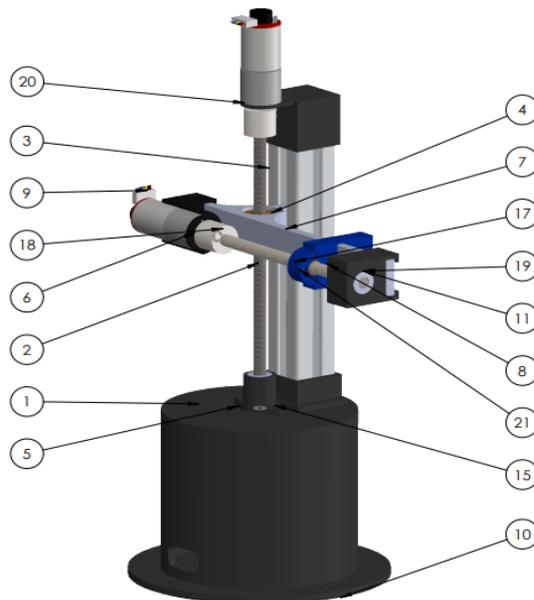


Fig. 4. Ensamble en 3D del robot cilíndrico de 3 GDL.

Además, en la Fig. 4 se observan los materiales que se utilizaron, en la Tabla 1 se detallan.

Tabla 1. Piezas del ensamble del robot.

N.º de Elemento	N.º de Pieza	Material	Cantidad
1	Base	ABS	1
2	Tornillo 2	Acero	1
3	Perfil 1	ABS	1

4	Tornillo t	Acero	1
5	Chumacera	ABS	1
6	Cople motor	Ensamble dl	2
7	Lamina	Aluminio	1
8	Tuerca de bronce	Bronce	2
9	Encoder	Electrónico	2
10	Base_2	ABS	1
11	Rodamiento	Acero	2
12	Motor grande	Ensamble	1
13	Cople	ABS	1
14	Soporte M1	ABS	1
15	Base perfil	ABS	1
16	Cople moch	ABS	2
17	Tornillo	Acero	1
18	Base M2	Aluminio	1
19	Chumacera 3	Acero	1
20	Base M1	ABS	1
21	Soporte gripper	ABS	1

El tamaño de los eslabones del robot son indispensables conocerlos (Fig. 5) para saber hasta qué puntos se puede llegar y a cuáles no se podrá. En este caso se tiene una distancia en altura máxima de 13 cm y para los ejes XY la distancia máxima alcanzada es de 9.5 cm, esta distancia está delimitada por otro círculo interior más pequeño el cual corresponde al espacio de trabajo que no puede ser alcanzada.

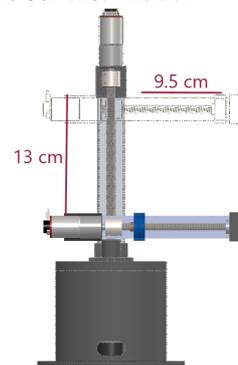


Fig. 5. Dimensiones de trabajo del robot cilíndrico de 3 GDL.

2.1.3. Desarrollo mecánico que contempla la manufactura y ensamble.

El proceso de fabricación de las piezas del robot, las cuales se muestran en el CAD de la Fig. 6 en una vista explosionada, se llevó a cabo mediante la técnica de manufactura aditiva utilizando el material ABS (Acrilonitrilo Butadieno Estireno). La selección de este material y la impresión bajo ciertos criterios específicos se tradujeron en una significativa ventaja en términos de resistencia mecánica. La manufactura aditiva con ABS permitió crear piezas sólidas y duraderas, lo que es esencial en el funcionamiento y desempeño del robot. Los criterios empleados durante la impresión incluyeron parámetros como la densidad de relleno, la orientación de las capas y la temperatura de impresión, entre otros factores, que influyen en la resistencia y calidad final de las piezas [18].

Como resultado de este proceso, se obtuvieron piezas con alta resistencia mecánica, lo que asegura una mayor durabilidad y confiabilidad en el funcionamiento del robot durante su operación y en las tareas que llevará a cabo. La elección adecuada de la tecnología de impresión y el material utilizado contribuyó de manera significativa a la calidad y eficiencia del prototipo del robot

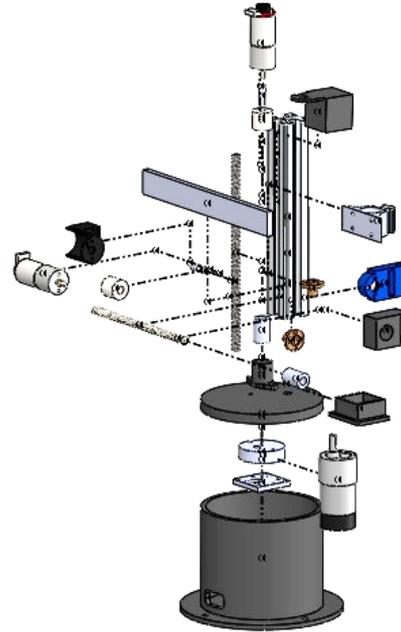


Fig. 6. Vista explosionada del robot cilíndrico de 3 GDL

En el contexto del ensamblaje, se obtuvo un prototipo, el cual se presenta en la Fig. 7, que exhibe las piezas fundamentales de dicho robot.

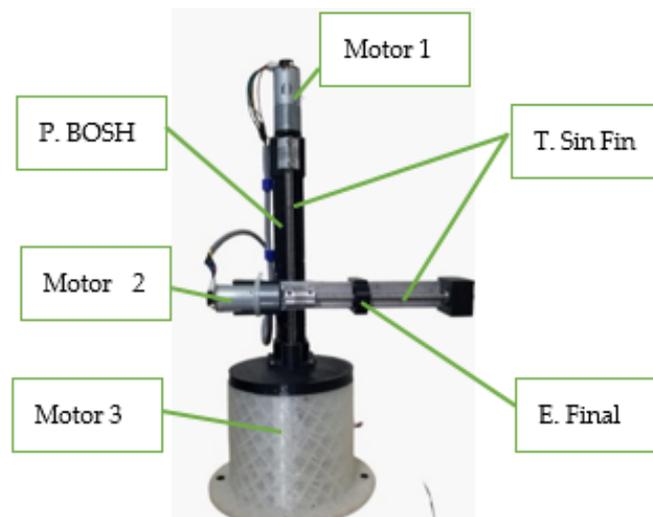


Fig. 7. Ensamble final con piezas manufacturadas.

2.2. Instrumentación electrónica y electromecánica.

Una característica crucial de los actuadores en los robots es su capacidad para retroceder o avanzar, es decir, permitir que los motores giren tanto en sentido horario como antihorario, esta tarea la desempeñan *puentes H*. Los puentes H son circuitos electrónicos que se utilizan para controlar la dirección del flujo de corriente en los motores. Estos dispositivos permiten invertir la polaridad de la tensión aplicada a los motores, lo

que posibilita su giro en ambos sentidos, lo que es esencial para el movimiento y la manipulación adecuada del robot [19].

Además, para conocer la posición precisa del robot durante su operación, los motores trabajan en conjunto con encoders. Los encoders son dispositivos que generan pulsos eléctricos en respuesta al movimiento del motor. Estos pulsos se utilizan en la programación para determinar el desplazamiento y la posición angular del robot. De esta manera, se obtiene información en tiempo real sobre la ubicación del robot en su entorno.

2.2.1. Especificaciones del motor y encoder

Se mencionarán las especificaciones que se necesitan de cada motor para poder obtener las posiciones:

Motor articulación 1:

- Pulsos por revolución encoder: 16 PPR
- Relación de engranes: 100:1

Motor articulación 2 y 3:

- Pulsos por revolución encoder: 11 PPR
- Relación de engranes: 45:1

2.2.2. Obtención de posiciones angulares y lineales

Para obtener las posiciones angulares que es en el caso de la articulación 1 que es en radianes de este motor de acuerdo al encoder se realiza un cálculo de matemático. La fórmula es la siguiente:

$$PosAngular = \left(\frac{contador}{1600} \right) (2 * \pi) \quad (1)$$

Donde: contador es el número de pulsos del encoder que se están contado en la interrupción del programa. El número 1600 corresponde al número de pulsos que se requiere que cuente el encoder para que el eje del motor de una vuelta completa. El motor cuenta con una caja de engranes con una relación de 100 a 1. Esto es por cada 100 vueltas que del eje del motor en su eje final de engranes apenas se dará una vuelta. Como el encoder del motor tiene 16 pulsos por vuelta se tiene que:

$$1 \text{ vuelta} = 100 * 16 = 1600. \quad (2)$$

En el caso de las articulaciones 2 y 3 cambian las ecuaciones debido a que son lineales, el desplazamiento lineal debemos conocer el avanza del tornillo sin fin por una vuelta. Conociendo también el número de pulsos del encoder para una vuelta del motor se tiene:

$$PosLineal = 0.2 * contador / 495 \quad (3)$$

Donde: 0.2 es el avance del tornillo sin fin por una vuelta, esto es 2 mm por una vuelta. Contador es el número de pulsos del encoder que se están contado en la interrupción del programa. 495 son los pulsos del motor que se requieren para dar una vuelta, entonces:

$$1 \text{ vuelta} = 45 * 11 = 495 \quad (4)$$

El diagrama electrónico representado en la Fig. 8 muestra cómo se integran los puentes H y los encoders en el sistema de control del robot. Esta combinación de componentes y programación en el microcontrolador permite lograr el control sobre los motores y obtener información valiosa sobre la posición del robot durante su funcionamiento, lo que es esencial para la ejecución de tareas específicas y la navegación segura del robot en su entorno.

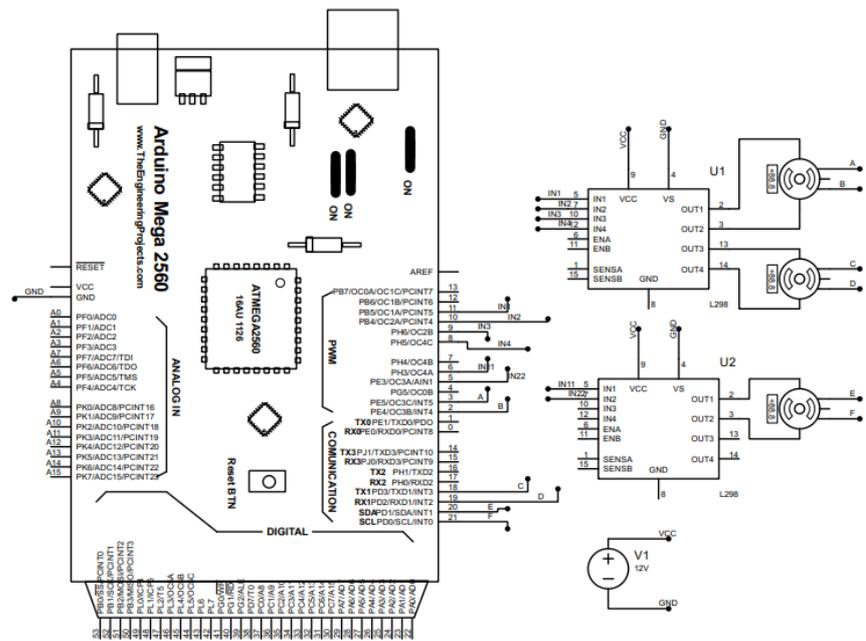


Fig. 8. Circuito electrónico para el robot.

2.3. Modelado matemático

El modelo matemático de un robot cilíndrico de 3 grados de libertad (GDL) es de gran importancia en el diseño, análisis, control y planificación de movimientos del robot [21]. Algunas de las razones fundamentales que destacan la relevancia de este modelo son las siguientes:

- Análisis cinemático.
- Planificación de trayectorias.
- Control y cinemática inversa.
- Simulación y validación.

2.3.1. Modelo Cinemático Directo

Tengamos en cuenta que la cinemática del robot estudia el movimiento del mismo con respecto a un sistema de referencias sin considerar todas las fuerzas que intervienen dividiendo en dos campos que son [22]:

Modelo Cinemático directo (MCD): que consiste en determinar cuál es la posición y orientación del efector final de robot, con respecto a un sistema de coordenadas que se toma como referencia, conocidos los valores de las articulaciones y los parámetros geométricos de los elementos del propio robot.

Modelo Cinemático inverso (MCI): este resuelve la configuración que debe adoptar el propio robot para una posición y orientación del extremo conocidas.

Para el modelo cinemático se corrobora bajo tres métodos que son los siguientes:

- **Modelo Geométrico**

En la Fig. 9 se observa un diagrama del brazo robótico de 3 grados de libertad RPP.

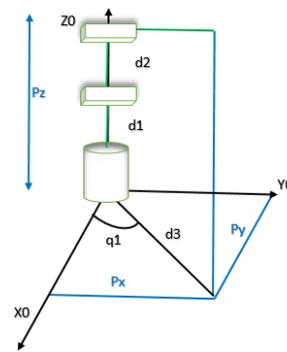


Fig. 9. Diagrama para el modelo cinemático directo para un robot de 3 grados de libertad con el método geométrico.

Para obtener las coordenadas se resuelve las resultantes del triángulo que se forma haciendo uso del seno y el coseno.

$$Px = d3 \cos(q1) \tag{5}$$

$$Py = \text{sen}(q1)d3 \tag{6}$$

$$Pz = d1 + d2 \tag{7}$$

• **Método de matrices homogéneas**

Cabe resaltar que hay diferentes formas para obtener la cinemática directa, ahora resolverá a través del método de matrices homogéneas. En el diagrama de la Fig. 10 realizado se especifican los sistemas de orientación de las traslaciones y rotaciones que se tienen que realizar para llegar al sistema final y con estas orientaciones se obtienen sus matrices y después se multiplican todas ellas.

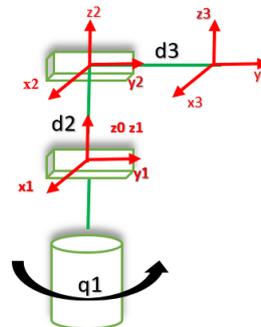


Fig. 10. Diagrama de sistemas de orientaciones y traslaciones.

Por lo tanto, se obtienen las matrices de rotación y traslaciones.

$$Rz(q1) = \begin{bmatrix} \cos(q1) & -\text{sen}(q1) & 0 & 0 \\ \text{sen}(q1) & \cos(q1) & 0 & 0 \\ 0 & 0 & 1 & d1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{8}$$

$$Tx(d1) = \begin{bmatrix} 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & d2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{9}$$

$$Tx(d2) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d3 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (10)$$

$$Tx(d3) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (11)$$

Una vez teniendo todas las matrices se multiplican

$$H = Rz(q1)Tx(d1)Tx(d2)Tx(d3), \quad (12)$$

Obteniendo el resultado final de la ecuación 13.

$$H = \begin{bmatrix} \text{sen}(q1) & 0 & \text{co s}(q1) & d3\text{co s}(q1) \\ -\text{co s}(q2) & 0 & \text{sen}(q1) & d3\text{sen}(q1) \\ 0 & -1 & 0 & d1 + d2 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (13)$$

• Método Devanit-Haartenberg

Para encontrar los parámetros DH se debe de seguir el algoritmo desarrollado por Denavit y Hartenberg.

a = distancia entre los ejes Z0 y Z1 medida a lo largo de X1.

α = ángulos entre Z0 y Z1 medidos entorno a X1.

d = distancia que existe entre los ejes X0 y X1 medida a lo largo de Z0.

θ = ángulo que existe entre X0 y X1 medida entorno a Z0.

Para el caso de la formulación estándar de DH la transformación genérica tiene la forma:

$${}^{i-1}T_i^{\text{estandar}} = \begin{bmatrix} c\theta_i & -s\theta_i * c\alpha_i & s\theta_i * c\alpha_i & a_i * c\theta_i \\ s\theta_i & c\theta_i * c\alpha_i & -c\theta_i * c\alpha_i & a_i * s\theta_i \\ 0 & s\alpha_i & c\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (14)$$

Por lo tanto, se desarrolla la siguiente tabla.

Tabla 2. Tabla DH del robot cilíndrico.

Eslabón	l_i	α_i	d_i	θ_i
1	0	0	d1	q1
2	0	$-\pi/2$	d2	$-\pi/2$
3	0	0	d3	0

Con la tabla 1 se obtienen las matrices de transformación que se sustituyen en ecuación 14 con los valores de acuerdo a los parámetros.

$$H_0^1 = \begin{bmatrix} \cos(q_1) & -\text{sen}(q_1) & 0 & 0 \\ \text{sen}(q_1) & \cos(q_1) & 0 & 0 \\ 0 & 0 & 1 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (10)$$

$$H_1^2 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & d_2 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (11)$$

$$H_2^3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (12)$$

Por consiguiente, se procede a obtener la matriz de transformación homogénea que relaciona el sistema 3 con el sistema.

$$H_0^3 = H_0^1 H_1^2 H_2^3. \quad (13)$$

$$H_0^3 = \begin{bmatrix} \text{sen}(q_1) & 0 & \cos(q_1) & d_3 \cos(q_1) \\ -\cos(q_1) & 0 & \text{sen}(q_1) & d_3 \text{sen}(q_1) \\ 0 & -1 & 0 & d_1 + d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (14)$$

Donde se obtiene una comprobación más como en los diferentes métodos anteriores del modelo cinemático directo para el robot manipulador cilíndrica que está dada por la siguiente forma de la Ecuación 15.

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = F_R(q) = \begin{bmatrix} d_3 \cos(q_1) \\ \text{sen}(q_1) d_3 \\ d_1 + d_2 \end{bmatrix}, \quad (15)$$

2.3.2. Modelo Cinemático Inverso

En el diagrama de la Fig. 9, se visualiza un triángulo que se forma a partir de las coordenadas que componen el sistema, específicamente las coordenadas Px, d3 y el ángulo q1. La determinación de las variables desconocidas requiere la aplicación de un proceso de despeje. El orden de resolución de estas variables se determinará en función de las relaciones y ecuaciones que describen el sistema.

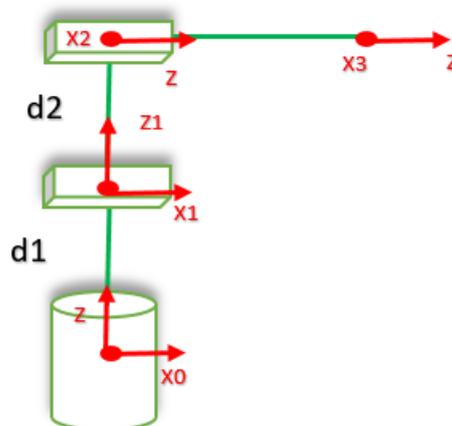


Fig. 11. Diagrama para el modelo cinemático inverso para un robot de 3 grados de libertad con el método geométrico.

El primer paso consiste en despejar la variable d2, que es posible de obtener a partir de las coordenadas y la relación entre los puntos del triángulo que se mencionó anteriormente. Una vez que se ha determinado d2, será posible continuar con el cálculo de d3 y q1. El proceso de

despeje, siguiendo las relaciones geométricas y las ecuaciones pertinentes, permitirá obtener las soluciones para d_2 , d_3 y q_1 , brindando así una descripción completa de las coordenadas y ángulos que caracterizan el sistema representado en la Fig. 9.

$$Pz = d_1 + d_2 \sin(q_2) = Pz - d_1 \quad (16)$$

$$d_3^2 = Px^2 + Py^2$$

$$d_3 = \sqrt{Px^2 + Py^2}, \quad (17)$$

$$\tan(q_1) = \frac{Py}{Px}$$

$$q_1 = \arctan\left(\frac{Py}{Px}\right), \quad (18)$$

2.3.2 Comprobación matemática simulada

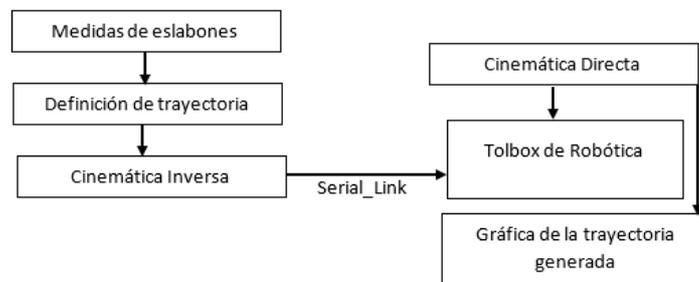


Fig. 12. Comprobación de la simulación con la ayuda del toolbox robótico.

Para la simulación del robot cilíndrico en MATLAB y con el toolbox de robótica se describe la Fig. 12: 1) Medidas de los eslabones: Esta etapa implica la obtención precisa de las longitudes de cada eslabón del robot manipulador. Estas medidas son esenciales para la posterior definición del modelo cinemático. 3) Definición de la trayectoria: Se establece la trayectoria que se espera que el robot manipulador siga. Esto puede ser una serie de posiciones y orientaciones en el espacio tridimensional que el robot debe alcanzar.

4) Cálculo cinemático inverso: A partir de la trayectoria definida, se realiza el cálculo para determinar la configuración articular requerida para que el robot siga esa trayectoria. Este proceso implica el uso de ecuaciones matemáticas que relacionan la posición y orientación del efector final con las articulaciones del robot. 5) Aplicación de la librería de Matlab y uso de SerialLink: Se emplea la librería de Matlab específicamente diseñada para la cinemática de robots. La clase SerialLink se utiliza para modelar y trabajar con manipuladores seriales, facilitando los cálculos y la representación del robot.

6) Cálculo de la cinemática directa: Se utiliza la información obtenida de las medidas de los eslabones y la configuración articular para calcular la posición y orientación del efector final del robot manipulador y se generan gráficos que representan la trayectoria calculada en comparación con la trayectoria definida inicialmente. Esto permite visualizar y verificar la precisión del modelo cinemático tanto en su versión directa como inversa.

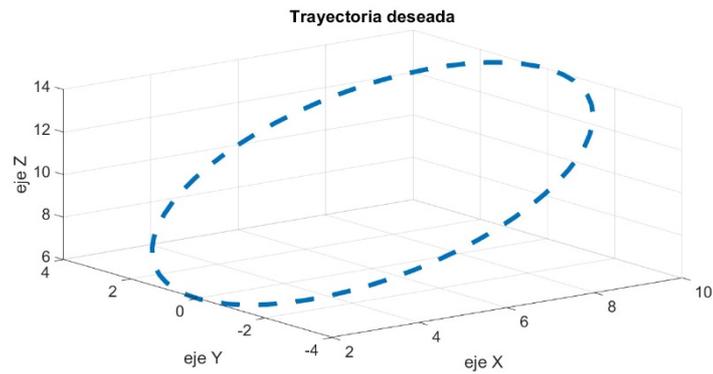


Fig. 13. Circulo deseado.

Estos valores deben ser comprobados con la cinemática directa es por eso que también se agrega y para ver que este correcta se gráfica el círculo deseado Fig. 13 y el que se está obteniendo con la cinemática directa. Por último, los valores de la cinemática directa se pasan al ploteo para ver la simulación como se muestra en la Fig. 14.

Trayectoria deseada vs Trayectoria real

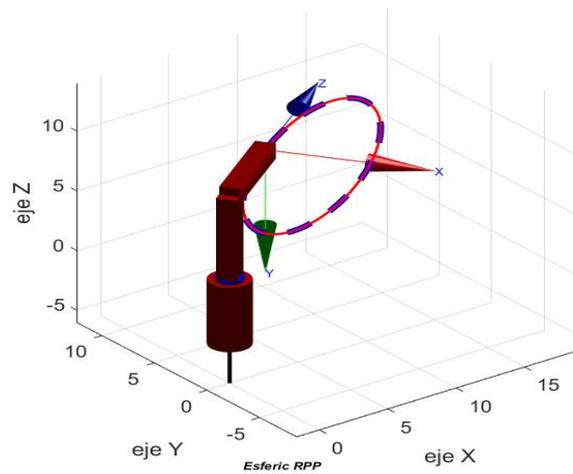


Fig. 14. Simulación del robot cilíndrico RPP siguiendo una trayectoria.

2.4. Programación de modelos matemáticos

El proceso descrito es una implementación de los modelos cinemáticos en un sistema que involucra tanto el Microcontrolador como el software MATLAB para monitorear y controlar el robot en tiempo real.

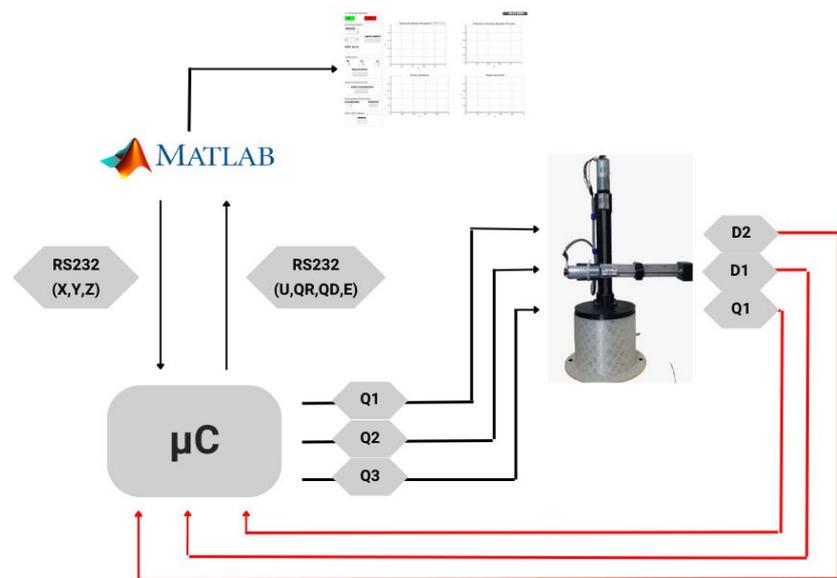


Fig. 15. Modelo matemático introducido a la programación.

De acuerdo al Fig. 15, esta dividido en secciones que son las siguientes:

2.4.1. Microcontrolador y el Modelo Cinemático Inverso (MCI).

Las ecuaciones del modelo cinemático inverso (MCI) se programan en el microcontrolador. Esto significa que cuando se envían las coordenadas cartesianas X, Y y Z desde MATLAB al microcontrolador utilizando las ecuaciones inversas para calcular las posiciones correspondientes de q1, d1 y d2 esto con las ecuaciones 1 y 3 que se analizaron en las secciones anteriores. Estas variables calculadas representan los ángulos y distancias necesarios para posicionar el robot en el espacio tridimensional de acuerdo con las coordenadas recibidas.

2.4.2. MATLAB y el Modelo Cinemático Directo (MCD).

Posteriormente, el microcontrolador devuelve estas variables (q1, d1 y d2) a MATLAB. Donde en MATLAB, se programaron las ecuaciones del modelo cinemático directo (MCD). Es decir, MATLAB toma los valores de q1, d1 y d2 y, mediante las ecuaciones inversas, calcula nuevamente las coordenadas X, Y y Z correspondientes. El resultado final es que MATLAB obtiene las coordenadas X, Y y Z que se han calculado en el microcontrolador, lo que permite graficar la posición del robot en el espacio tridimensional.

2.4.2. Control Bidireccional entre MATLAB y el Microcontrolador

Como se puede observar en los dos últimos subtemas la comunicación entre el software y el microcontrolador se logra un control bidireccional entre el microcontrolador y MATLAB para mover el robot a una posición deseada y obtener una visualización gráfica de su ubicación en el espacio. Esto es esencial para el control y la planificación de movimientos precisos del robot en diversas aplicaciones.

2.5. Control de posición articular y generación de trayectorias.

Para lograr el control de posición articular, se emplea la técnica del control Proporcional Integral Derivativo (PID) en tiempo discreto que lo describe la ecuación general:

$$[u[k] = K_p \cdot e[k] + K_i \cdot \sum_{i=0}^k e[i] \cdot T + K_d \cdot \frac{e[k] - e[k-1]}{T}] \quad (19)$$

Donde:

- $u[k]$ es la señal de control en el instante de tiempo discreto k .
- $e[k]$ es el error en el instante de tiempo discreto k . El error se calcula como la diferencia entre la referencia deseada y la salida actual del sistema: $e[k] = r[k] - y[k]$, donde $r[k]$ es la referencia deseada y $y[k]$ es la salida actual.
- K_p, K_i, K_d son las constantes de ganancia del controlador proporcional, integral y derivativo, respectivamente.
- T es el tiempo de muestreo, que representa el intervalo de tiempo entre las actualizaciones del controlador.

Otro tema importante a tomar en cuenta para el controlador PID es método de sintonización que se usó, en este caso es Ziegler-Nichols. Este método se basa en la identificación empírica de los parámetros del controlador utilizando una respuesta en lazo cerrado del sistema. Se asegura de que el motor se encuentra en su punto de reposo y que no hay cargas externas aplicadas, por ende, se siguieron los siguientes pasos para la sintonización de los motores:

- Incrementar gradualmente la ganancia proporcional (K_p) hasta que el sistema empiece a oscilar de manera estable y continua por sí solo. Esto dará lugar a la banda proporcional (P) y al período de oscilación (T_u).
- Por último, a partir de P y T_u , se puede calcular la ganancia integral (K_i) y la ganancia derivativa (K_d), utilizando las siguientes fórmulas:

$$K_i = 0.5 * K_p / T_u \quad (20)$$

$$K_d = 0.125 * K_p * T_u \quad (21)$$

En cuanto a la generación de trayectorias, se sigue un proceso que se muestra en el siguiente diagrama.

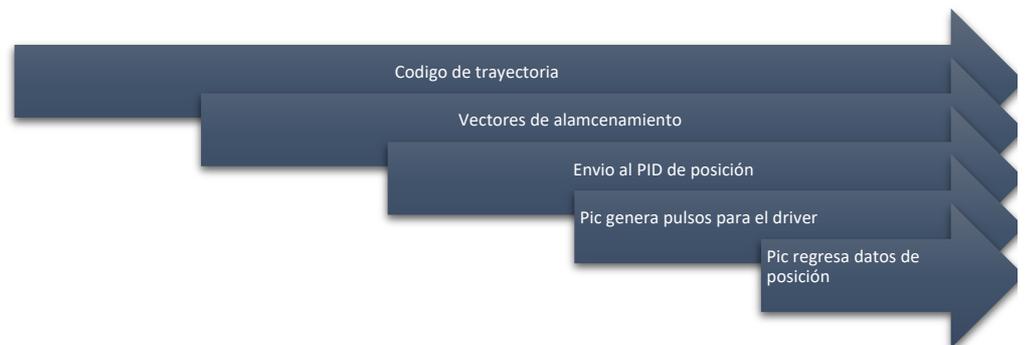


Fig. 16. Algoritmo que se sigue para la generación de trayectorias.

El Fig. 16 consiste en que primero se define la figura deseada mediante código, especificando el número de puntos que conformarán la trayectoria. Estos puntos se almacenan en un vector, y para que cada coordenada se genere y se envíe al Arduino, se implementa un bucle. La iteración de este bucle se extiende hasta alcanzar el número total de puntos requeridos para formar el círculo o la figura en cuestión. De esta manera, se logra generar secuencialmente cada coordenada y enviarla al Arduino, lo que permite que el robot siga la trayectoria establecida siguiendo la figura especificada. Cabe destacar que es fundamental considerar aspectos como la velocidad y aceleración del robot durante la ejecución de la trayectoria para garantizar movimientos suaves y precisos, así como la seguridad en su funcionamiento.

2.6. Interfaz gráfica de monitoreo y control

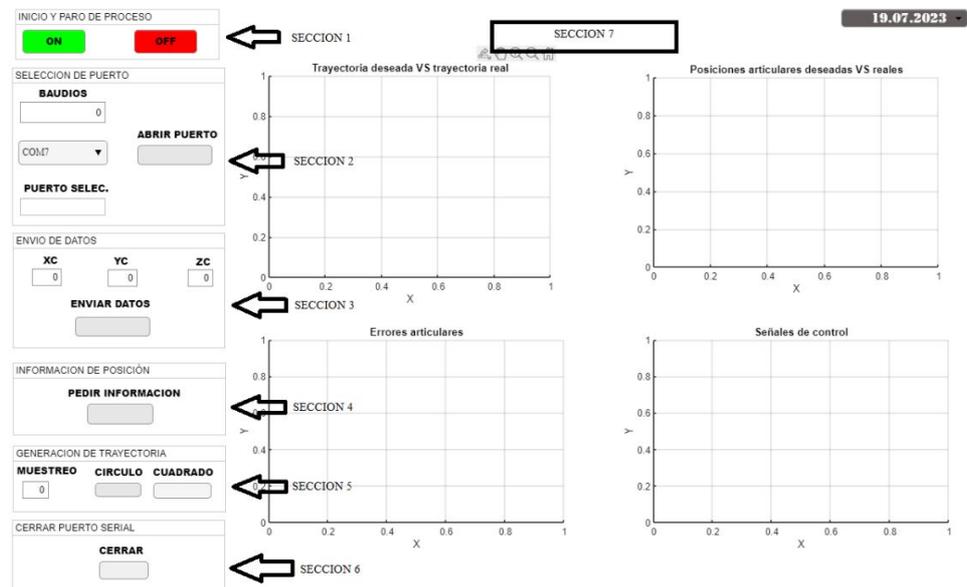


Fig. 17. Interfaz para el control del robot cilíndrico de 3 GDL.

La interfaz se desarrolló con la herramienta AppDesigner de MATAB, Fig. 17 esta interfaz proporciona una forma interactiva para que un usuario interactúe con el robot y controle sus funciones.

SECCIÓN 1.- En esta primera sección se cuenta con dos botones, uno de inicio y el segundo de paro, la función principal es que al ser presionados ejecuten una acción; La acción del primer botón de inicio "ON" al ser presionado es que activa una variable global que desbloquea todos los comandos y acciones de la guide en pocas palabras enciende el sistema para así dar funcionamiento a la GUI y sus características; En el caso del segundo botón de paro "OFF" desactiva la variable global en cualquier momento del proceso así deteniendo cada acción que se esté ejecutando en ese momento.

SECCIÓN 2.- En esta sección se enfoca en la comunicación serial con el microcontrolador para así enviar y recibir los datos del proceso, en el apartado de baudios se elige la velocidad de comunicación, de igual manera se muestra la elección del puerto "COM" por el cual se va a comunicar; Se tiene un botón que al ser presionado ejecuta la acción de abrir el puerto serial para así dar comienzo a la comunicación mediante el puerto serial

SECCIÓN 3.- Se mantienen 3 campos numéricos para guardar el valor que se le agregue, estos son correspondientes a "XC" "YC" "ZC" del robot, que son posiciones a donde se desea que llegue el robot, estas posiciones se envían mediante el puerto serial al ser presionado el botón "ENVIAR DATOS".

SECCIÓN 4.- En esta sección se piden los datos de posición del efector final del robot, al presionarlo en cualquier momento del proceso dará las coordenadas correspondientes a la posición de nuestro efector final.

SECCIÓN 5.- En la sección de generación de trayectoria se muestra campo numérico donde se configura el tiempo de muestreo con que se pedirán los datos del robot para su posterior análisis una vez puesto el dato de muestreo se pueden observar dos botones, el primero genera una trayectoria de un círculo el segundo una trayectoria de un cuadrado, estos al ser presionados activan las gráficas y así mismo se muestran los datos específicos de cada gráfica.

SECCIÓN 7.- En esta sección se muestran la parte de los resultados, son 4 graficas donde se muestran la trayectoria deseada vs la trayectoria real, las posiciones articulares deseadas vs las reales, los errores articulares, y las señales de control.

3. Resultados

Se realizaron dos trayectorias para que el robot cilíndrico de 3 GDL ejecutara basándose en el siguiente proceso del Fig. 18.

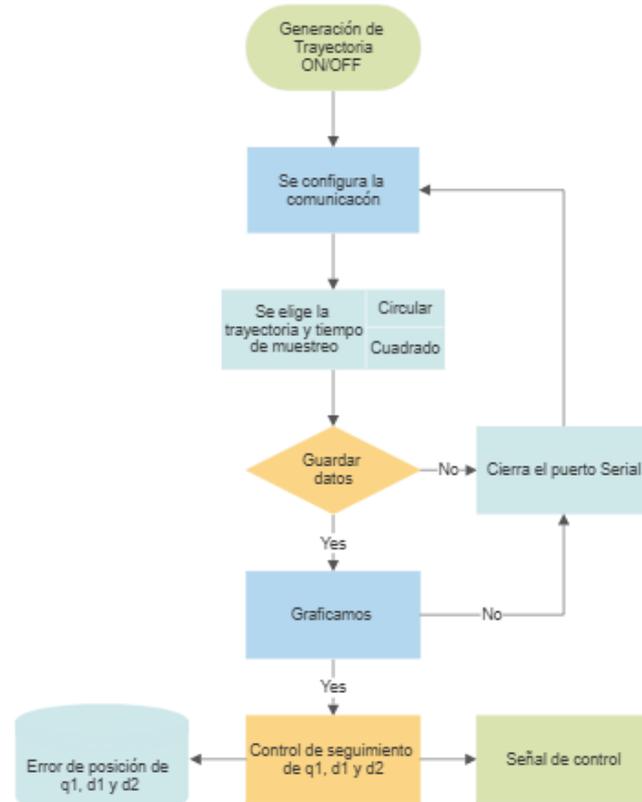


Fig. 18. Proceso de para la generación de una trayectoria en la interfaz.

3.1. Trayectoria del círculo

Las siguientes ecuaciones representan la trayectoria que se muestra es para la trayectoria de un círculo.

$$xr = xc + rc * \sin(a); \quad (22)$$

$$yr = yc + rc * \cos(a); \quad (23)$$

$$zr = zc + rc * \sin(a); \quad (24)$$

donde:

- a = es la función para generar el número de pasos.
- rc = es el radio del círculo.
- xc, yc, zc = definen el centro del círculo en 3 dimensiones.

En la Fig. 19 se observa el resultado final de la interfaz con la trayectoria del círculo donde se aprecian las cuatro gráficas que se analizarán más adelante con los datos que devuelve la interfaz.

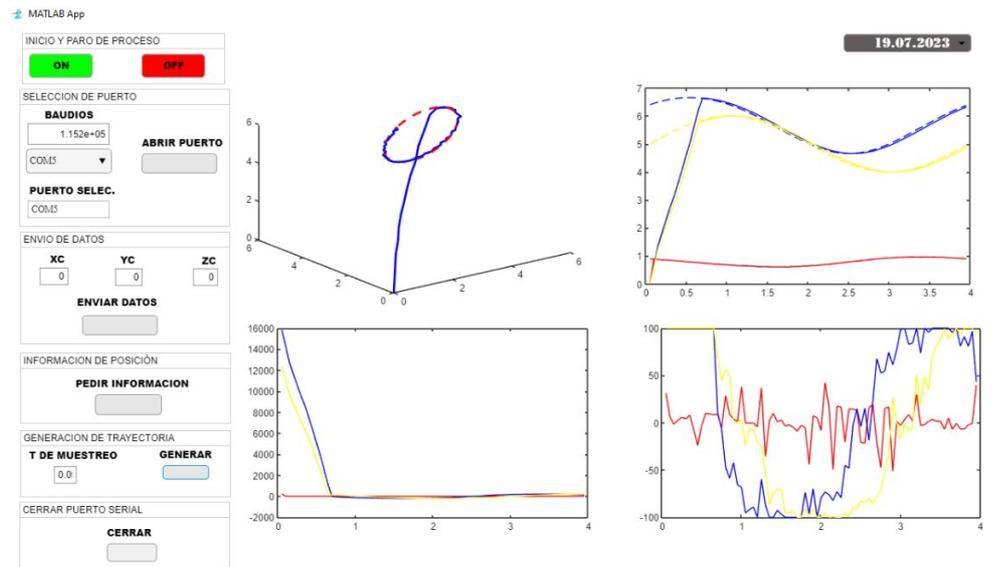


Fig. 19. Interfaz gráfica con seguimiento de trayectoria del círculo.

Con los datos que se obtienen de la interfaz se analiza más a profundidad los errores de posición que son de q_1 , d_1 y d_2 que son tres gráficas, en la Fig. 20 se describe el error en tiempo real de cada posición de las articulaciones del robot cilíndrico. Se observa que, al inicio del sistema, los errores son muy cercanos a cero, alrededor de 0.1 radianes en el caso de q_1 y en centímetros para d_1 y d_2 . Además, se observa que los errores oscilan alrededor del rango de -0.1 a 0.1, pero nunca sobrepasan estos valores. Esto demuestra la estabilidad del sistema de control y la capacidad del controlador PID para mantener el robot dentro de márgenes de error aceptables durante el movimiento.

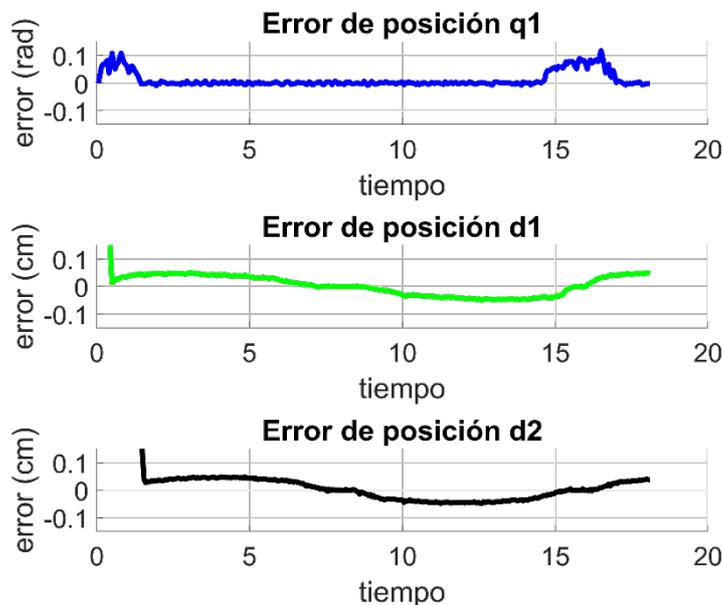


Fig. 20. Errores de posiciones de q_1 , d_1 y d_2 de la trayectoria del círculo.

De acuerdo a lo que se analizó en la Fig. 20 que son los errores de posición de las articulaciones, se puede observar en la Fig. 21 la trayectoria deseada y la real que es la que está generando el robo cilíndrico en físico, que parte desde su origen del robot y por consiguiente se posiciona en el círculo y después sigue sus trayectorias, donde se puede apreciar que si genera el círculo deseado.

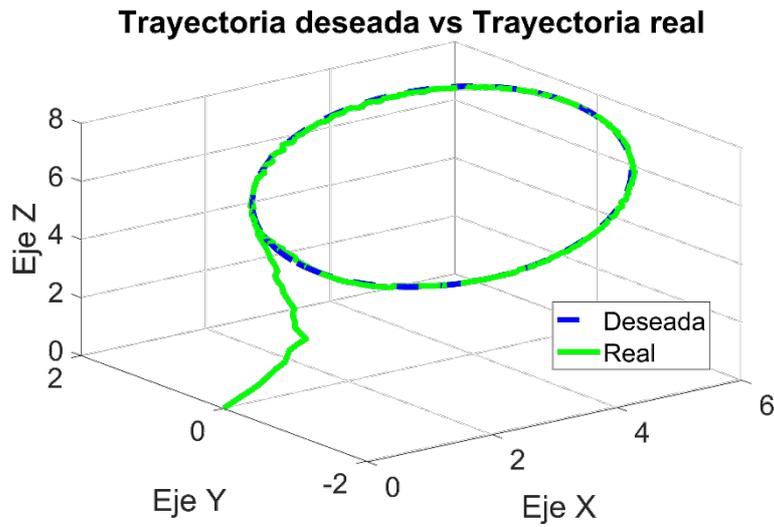


Fig. 21. Trayectoria Deseada contra la real.

En la Fig. 22 se presentan las señales de control tanto de q_1 , d_1 y d_2 se analiza cómo reacciona el controlador PID, se aprecia como la señal de control oscila desde el -100 al 100 %, las señales negativas indican el cambio de giro de la articulación. Además, esta figura indica que el modelo cinemático, tanto directo como inverso, está funcionando correctamente.

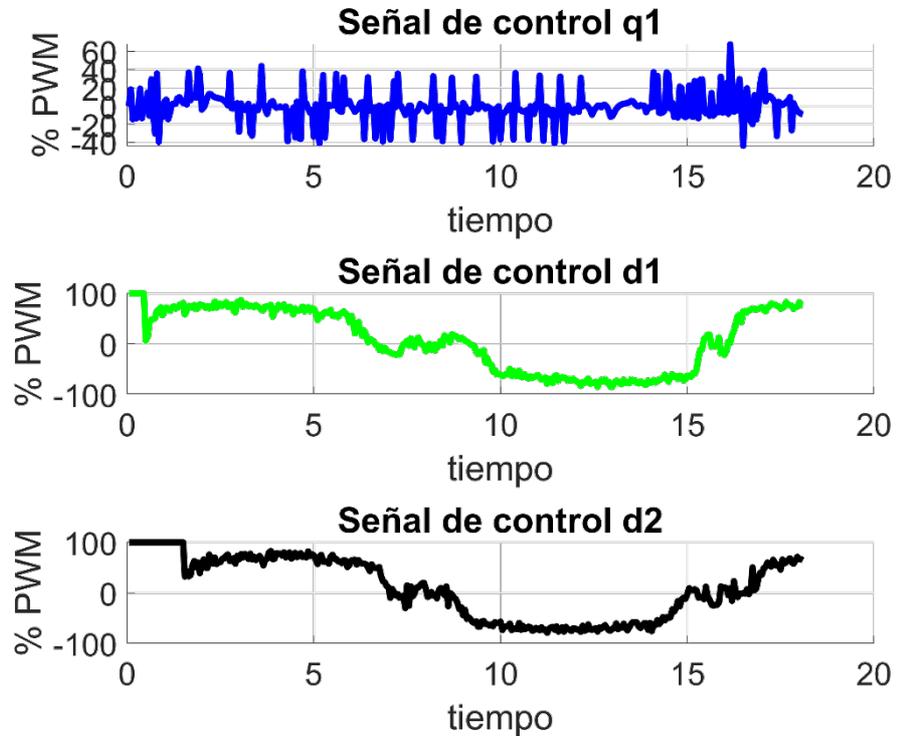


Fig. 22. Señales de control de q_1 , d_1 y d_2 .

En las siguientes figuras 23, 24 y 25 se observará el control de seguimiento de las posiciones q_1 , d_1 y d_2 deseadas contra las reales donde se aprecia que el controlador sigue a la referencia con un error mínimo.



Fig. 23. Control de seguimiento de la trayectoria en la posición lineal d_1 .

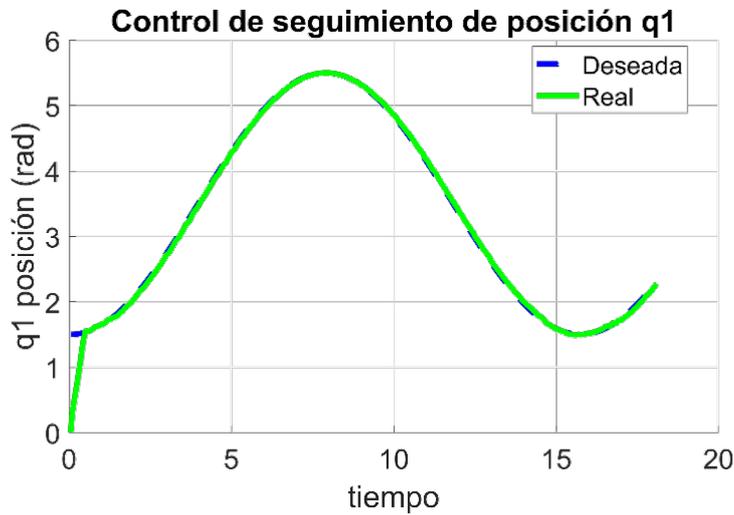


Fig. 24. Control de seguimiento de la trayectoria en la posición lineal q_1 .



Fig. 25. Control de seguimiento de la trayectoria en la posición lineal d_2 .

3.2. Trayectoria de cuadrado

Ahora se ahora la prueba con una trayectoria que genera un cuadrado de esta forma para analizar qué tan derecho se puede ir el robot desarrollado, a continuación, se muestra la metodología para generar un cuadrado. Supongamos que el cuadrado tiene un centro (x_0, y_0) , por consiguiente, se definen los cuatro vértices del cuadrado como se muestra a continuación.

- Vértice superior izquierda:

$$\left(x_0 - \frac{L}{2}, y_0 + \frac{L}{2}\right) \tag{25}$$

- Vértice superior derecho:

$$\left(x_0 + \frac{L}{2}, y_0 + \frac{L}{2}\right) \tag{26}$$

- Vértice inferior izquierda:

$$\left(x_0 - \frac{L}{2}, y_0 - \frac{L}{2}\right) \tag{27}$$

- Vértice inferior derecho:

$$\left(x_0 + \frac{L}{2}, y_0 - \frac{L}{2}\right) \tag{28}$$

Para poder generar la trayectoria cerrada del cuadrado hacemos que el centro del cuadrado se mueva en un círculo con radio R , describiendo la ecuación paramétrica 29.

$$x(t) = x_0 + R \cdot \cos(2\pi \cdot t) \tag{29}$$

$$y(t) = y_0 + R \cdot \sin(2\pi \cdot t) \tag{30}$$

En la Fig. 26 se observa la interfaz ejecutando la trayectoria del cuadrado devolviéndonos varias gráficas que se analizarán más adelante con los datos que devuelve la interfaz.

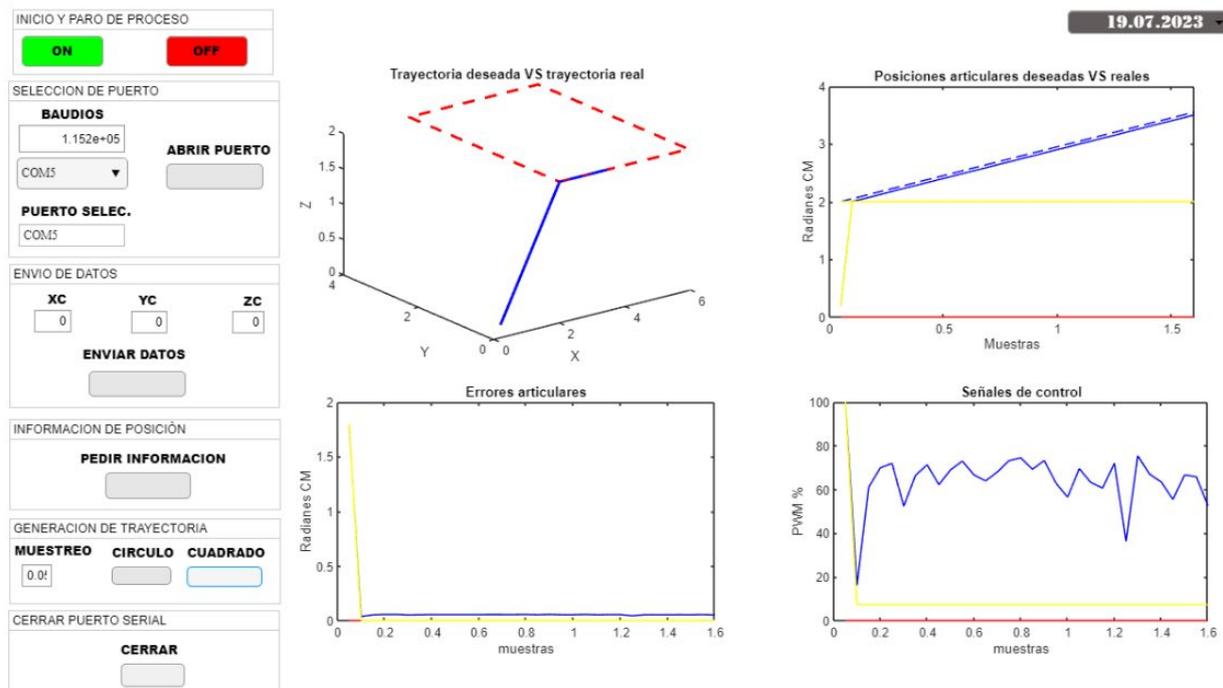


Fig. 26. Simulación de la trayectoria cuadrada.

Con los datos que se obtienen de la interfaz se graficaron los errores de q_1 que se puede observar cómo el error en radianes solo varía en 0, en donde en un momento a veces crece su error pero solo es un error de 0.1, mientras en los valores de d_1 varía el error entre 0.1 y -0.1 cm y por último el error de posición en d_2 es de solo en el inicio en un instante su error es de 0.1 cm, pero después de ahí todo siempre se mantiene en 0 como se muestra en la Fig. 27.

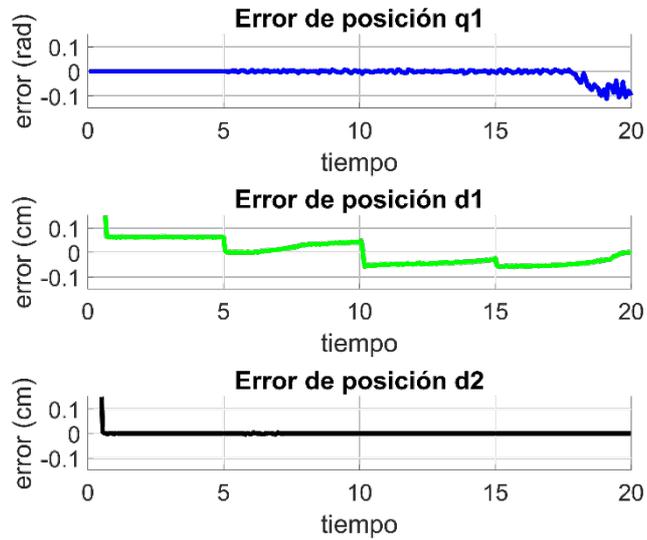


Fig. 27. Errores de posiciones en q_1 , d_1 y d_2 en la trayectoria del cuadrado.

En la Fig. 28 se analiza la trayectoria deseada y la real que es la que va siguiendo el robot cilíndrico, donde se analiza como primero parte del origen hasta llegar a una esquina del cuadrado y después sigue todo el cuadrado creando una copia del deseado en las tres dimensiones.

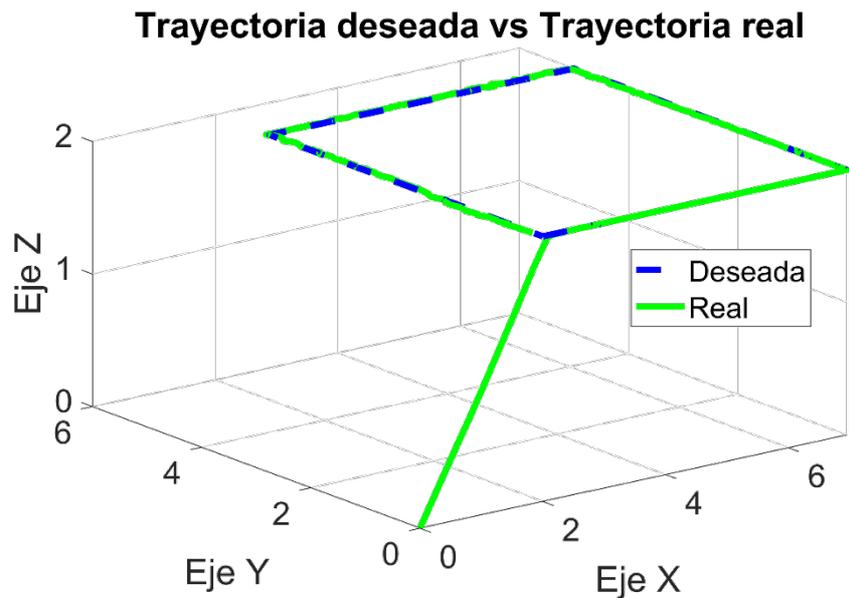


Fig. 28. Cuadrado deseado y real con los datos obtenidos de la interfaz.

En las Figuras 29, 30 y 31 se aprecia como el control de seguimiento es intacto, es decir el deseado con el real se enciman las líneas dando por sentado que el control de posición logra llevar al robot a las posiciones deseadas con esto se refirman que los modelos cinemáticos directo e inverso son correctos.

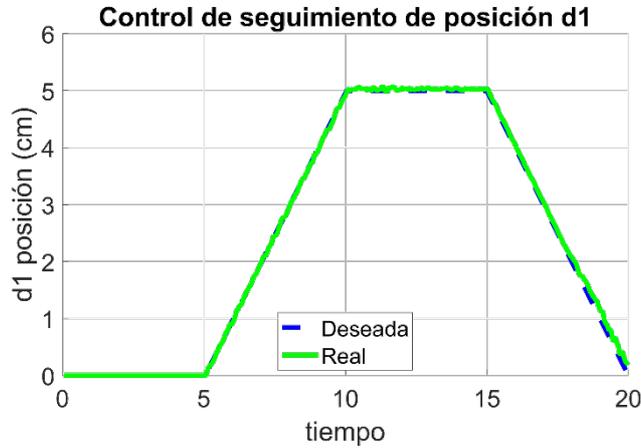


Fig. 29. Control de seguimiento de la trayectoria en la posición lineal d1.

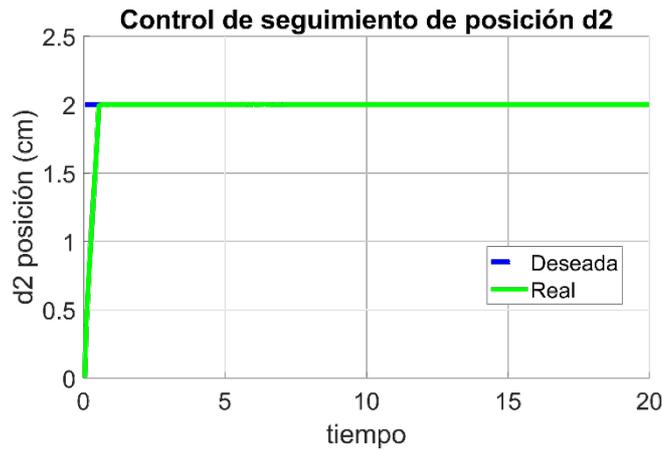


Fig. 30. Control de seguimiento de la trayectoria en la posición lineal d2.

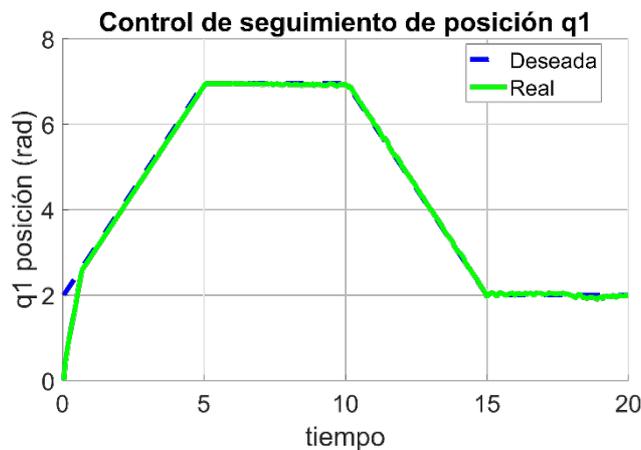


Fig. 31. Control de seguimiento de la trayectoria o en la posición angular q1.

Al analizar los datos obtenidos en tiempo real de las posiciones en cada eje de cada trayectoria con la siguiente ecuación.

$$ECM = \sqrt{\frac{\sum_{i=1}^n (v_i - \bar{v}_i)^2}{n}} \quad (31)$$

Se obtuvieron los errores cuadráticos medios y la tasa de error del plano cartesiano con respecto al área de trabajo donde se muestran en la Tabla 3.

Tabla 3. Error medio cuadrado y la tasa de error en XYZ.

	Espacio de trabajo	Círculo		Cuadrado	
		RMSE	% Error	RMSE	% Error
X	9.5 cm	0.0304	0.3195	0.0231	0.2434
Y	9.5 cm	0.1407	1.4810	0.2027	2.1339
Z	13 cm	0.6351	4.8851	0.1789	1.3759

4. Discusión

Se presentó el diseño y desarrollo de un robot cilíndrico de tres grados de libertad para el control de seguimiento de trayectorias. El sistema de control de trayectorias demostró acercarse bastante a la trayectoria deseada. Esto se puede ver en los resultados obtenidos y en las gráficas, el controlador de posición resultó tener un rendimiento en cuanto a precisión y cercanía. Aunque el robot cilíndrico mostró resultados prometedores, también se identificaron algunas limitaciones que deben abordarse en futuras iteraciones. Una de ellas es el tipo de control articular empleado, que, aunque demostró acercarse a la trayectoria deseada no se alcanzaba con error 0. Para aplicaciones más precisas será necesario aplicar otro tipo de controlador como lo es un control PID de trayectorias el cual permite alcanzar las coordenadas deseadas.

5. Conclusiones

Este trabajo detalla el proceso de diseño y construcción de un robot manipulador con tres grados de libertad en un diseño cilíndrico. Esto en sí mismo es un aporte significativo, ya que la creación de un manipulador con estas características implica consideraciones específicas de diseño, fabricación, selección de materiales y análisis cinemático, entre otros aspectos técnicos.

La inclusión de una interfaz gráfica de usuario para el seguimiento de trayectorias es otra contribución valiosa. Esta interfaz permite a los usuarios interactuar fácilmente con el robot, definir y modificar trayectorias, visualizar resultados e incluso facilita el control del robot de manera intuitiva.

Es interesante notar que en el estudio del robot cilíndrico se lograron obtener los modelos cinemáticos directo e inverso de posición, lo que permitió realizar pruebas de trayectorias utilizando el controlador PID de posición. A través de estas pruebas, se pudo observar que el controlador se aproxima a las coordenadas deseadas con errores menores al 5%, lo que demuestra el correcto seguimiento de la trayectoria deseada. Sin embargo, se identificó que el controlador PID de posición tiene una limitación en estado estacionario, lo que significa que, aunque se acerca a las coordenadas deseadas, aún permanece un pequeño error constante en la posición final. Este error en estado estacionario puede deberse a diversas razones, como la no linealidad del sistema, inexactitudes en el modelo cinemático, fricciones o inercias en las articulaciones, entre otros factores. Para reducir este error en estado estacionario, se menciona que se pueden aplicar otro tipo de controladores. Esto se refiere a utilizar controladores más avanzados y sofisticados que superen las limitaciones del PID estándar. Algunos ejemplos de controladores más capaces incluyen controladores de tipo adaptativo, controladores

robustos, controladores fuzzy o controladores basados en redes neuronales. La selección del controlador más adecuado dependerá de las características y requerimientos específicos del robot cilíndrico y las tareas que deba realizar. Utilizando controladores más avanzados, es posible mejorar aún más el rendimiento del robot, reducir el error en estado estacionario y lograr un seguimiento más preciso de las trayectorias deseadas. Esto conduce a un funcionamiento más eficiente y confiable del robot en diversas aplicaciones industriales y de servicio.

Referencias

1. J. M. T. Peraza, «Diseño y construcción de un prototipo de robot con tres grados de libertad para posicionamiento de objetos,» ITCA Editores, Santa Tecla, 2018.
2. M. H. McCloy, Robotics, Springer Science & Business Media, 2013.
3. A. O. Baturone, *Robótica Manipuladores y robots móviles*, Barcelona: Marcombo, 2001.
4. A. Barrientos, *Fundamentos de Robótica*, Madrid: McGraw Hill, 2007.
5. P. A. B. Shah R, «Concept for Automated Sorting Robotic Arm,» *Procedia Manufacturing*, vol. 20, pp. 400-405, 2018.
6. D. C. I, «Calibración ojo a mano de un brazo robótico industrial con cámaras 3D de luz estructurada,» *Iberoamericana de Automática e Informática Industrial*, vol. XIX, nº 19, pp. 154-163, 2022.
7. C. M. Ramos, «Robótica y cirugía laparoscópica,» *Cirugía Española*, vol. 80, nº 4, pp. 189-194, 2006.
8. D. N. C. C., «Diseño y construcción de un brazo robótico de 6 GDL,» *Tekhnê*, vol. 13, nº 1, pp. 65-70, 2016.
9. C. H. Kevin Morales, «Diseño y optimización de la estructura mecánica de un brazo robótico antropomórfico desarrollado con fines educativos,» *UIS INGENIERÍAS*, vol. 1, nº 12, p. 23, 2019.
10. J. K. K. S. S. Atirav Seth, «Design and simulation of 6-DOF cylindrical robotic manipulator using finite element analysis,» *Materials Today: Proceedings*, vol. 62, nº 3, pp. 1521-1525, 2022.
11. D. P. Cando Garzón, «Diseño y construcción de un brazo robótico para un sistema de clasificación de objetos de forma geométrica mediante el reconocimiento de imágenes por medio de una cámara de video,» Quito: Universidad Israel, Israel, 2016.
12. F. A. Alamsyah, «The Kinematics Analysis of Robotic Arm manipulators Cylindrical Robot RPP Type for FFF 3D Print using Scilab,» *IOP Conference Series: Materials Science and Engineering*, vol. 494, 2019.
13. C. T. C. A. Jose de Jesus Rubio, «OPTIMAL CONTROL BASED IN A MATHEMATICAL MODEL APPLIED TO ROBOTIC ARMS,» *International Journal of Innovative Computing, Information and Control*, vol. 7, nº 8, pp. 5045-5062, 2011.
14. R. López, «Análisis cinemático y dinámico de un manipulador robótico con tres grados de libertad,» Repositorio Dspace, 2018.
15. E. A. M. Cruz, «Metodología para Generación de Trayectorias de Manipuladores Roboticos, Su Cinematica y Dinámica,» IPN, México, 2000.
16. O. H. H. Ş. Rüştü GÜNTÜRKÜN, «Design and Application of Plc Controlled Robotic Arm Choosing Objects According to Their Color,» *Electronic Letters on Science & Engineering*, vol. 16, nº 2, pp. 50-62, 2020.
17. V. L. Svitlana Sotnik, «Modern Industrial Robotics Industry,» *International Journal of Academic Engineering Research (IJAER)*, vol. 6, nº 1, pp. 37-46, 2022.
18. J. H. C. Rojas, «Metodología de diseño mecatrónico de robots,» *Épsilon*, vol. 1, nº 25, 2005.
19. K. L. A. C., «Influencia del porcentaje de relleno en la resistencia mecánica en impresión 3D, por medio del método de Modelado por Deposición Fundida (FDM),» *Ingeniare*, vol. 24, nº 12, 2016.
20. M. O. Gómez, «Manejo de motores DC con puente H y control por modulación de pulso PWM Sigma con aplicaciones en simuladores de vuelo, caso de la reparación del simulador de vuelo del CIMA,» *REDICCES*, vol. 1, nº 13, 2015.
21. M. A. B. P. C. R. C. Z. S. B. R. R. J. H. A. A. E. A. M. R. Jack Lut Aguirre Valverde, «CONTROL ÓPTIMO LQR PARA ORIENTAR LA PLATAFORMA MÓVIL DE UN ROBOT PARALELO DE 2GDL EXPERIMENTAL,» *Investigaciones*, vol. 10, nº 4, pp. 360-369, 2021.
22. A. Barrientos, *Fundamentos de Robótica*, España: McGraw-Hill/Interamericana.
23. P. B. P. G. M. K. S. A. J. O. Á. M. Molina Cárdenas M, «Diseño y Construcción del Prototipo de un Brazo Robótico con Tres Grados de Libertad, como Objeto de Estudio,» *Ingeniare*, vol. 18, nº 7, pp. 87-94, 2015.