

ESCUELA POLITÉCNICA NACIONAL

ESCUELA DE FORMACIÓN DE TECNÓLOGOS

SIMULACIÓN EN EL ENTORNO SIMULINK DE MATLAB DE UN BRAZO ROBÓTICO DE 4 GRADOS DE LIBERTAD (GDL)

TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE TECNÓLOGO EN ELECTROMECAÁNICA

Alex Humberto Guacho Tituaña

alex.guacho@epn.edu.ec

DIRECTORA: ING. CATALINA ELIZABETH ARMAS FREIRE, MSC.

catalina.armas@epn.edu.ec

CODIRECTOR: ING. CARLOS IVÁN ZAMBRANO OREJUELA, MSC.

ivan.zambrano@epn.edu.ec

Quito, junio 2021

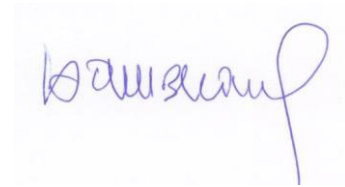
CERTIFICACIÓN

Certificamos que el presente trabajo fue desarrollado por el Sr Guacho Tituaña Alex Humberto como requerimiento parcial a la obtención del título de TECNÓLOGO EN ELECTROMECAÁNICA, bajo nuestra supervisión:



**Ing. Catalina Elizabeth
Armas Freire. MSC**

DIRECTORA DEL PROYECTO



**Ing. Oscar Iván Zambrano
Orejuela. MSC**

CODIRECTOR DEL PROYECTO

DECLARACIÓN

Yo, Alex Humberto Guacho Tituaña con CI: 172134732-4 declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y que he consultado las referencias bibliográficas que se incluyen en este documento.

Sin perjuicio de los derechos reconocidos en el primer párrafo del artículo 144 del Código Orgánico de la Economía Social de los Conocimientos, Creatividad e Innovación – COESC-, soy titular de la obra en mención y otorgó una licencia gratuita, intransferible y no exclusiva de uso con fines académicos a la Escuela Politécnica Nacional.

Entregó toda la información técnica pertinente, en caso de que hubiese una explotación comercial de la obra por parte de la EPN, se negociará los porcentajes de los beneficios conforme lo establece la normativa nacional vigente.



Alex Humberto Guacho Tituaña

CI: 172134732-4

Teléfono: 0960901979/02-2585138

Correo: alex.guacho@epn.edu.ec

DEDICATORIA

El siguiente proyecto de simulación busca brindar información a estudiantes o personas que comiencen a conocer sobre la robótica en el entorno Simulink de Matlab, desde cómo importar un modelo diseñado en 3D de un brazo robótico, hasta llegar a obtener el ensamblaje de este y simular su movimiento para conocer la posición de su efector final.

AGRADECIMIENTO

En la vida se presentan grandes retos y batallas, las mismas que Dios pone a los grandes guerreros. Agradecido con mis padres Humberto Ranulfo Guacho Guzman y Clara Magdalena Tituaña Criollo, por ser mi ejemplo de lucha y constancia para poder alcanzar lo que en la vida me proponga. Su apoyo ha sido uno de los factores que me ha permitido llegar a cumplir uno de mis grandes sueños, permitiéndome tener valores como personal y futuro profesional. También a mi hermano Alexis Damian Guacho Tituaña quien con su espíritu de niño alegrado cada uno de mis días y me ha permitido ser uno de sus ejemplos a seguir por la constancia y dedicación que tengo para superarme día a día.

Agradezco a mi abuelito Juan Manuel Tituaña Caiza quien en vida me enseñó que, a pesar de tener una enfermedad terminal, las personas no deben dejar de luchar por más difícil que sean las adversidades; que en la vida perdura quién eres como persona y no por los títulos que obtengas.

Mi gratitud inmensa a mi novia Génesis por brindarme su apoyo incondicional sin importar las adversidades que se presentaron, en los momentos buenos y malos fue ese refugio y apoyo para encontrar la fuerza para continuar adelante y no dejarme vencer.

Muchas gracias a cada miembro de mi familia que supo darme su apoyo en todo momento, pero en especial a mis padrinos: Ángel Padilla y Mariana Tituaña, como a mis tíos: Ángel Tituaña y Verónica Yugsi, quienes con una palabra de aliento y consejos me hicieron sentir que confiaban en mí y que estarían en todo momento prestos ayudarme.

Finalmente, mi gratitud a todos los ingenieros que en mi carrera universitaria me brindaron los conocimientos necesarios para formarme como profesional, como a mi directora la ingeniera Catalina Elizabeth Armar Freire; quien me brindó su apoyo en todo momento para realizar este trabajo de titulación.

ÍNDICE DE CONTENIDOS

1	Introducción.....	8
1.1	Objetivo general	9
1.2	Objetivos específicos	9
1.3	Fundamentos	9
	Brazo robótico	9
	Cinemática directa.....	10
2	Metodología	14
2.1	Descripción de la metodología usada	14
3	Resultados y Discusión	16
3.1	Características y especificaciones técnicas de diferentes brazos robóticos. .	16
3.2	Modelo del brazo robótico que permite la simulación en el entorno de trabajo de Simulink de Matlab.....	20
3.3	Ecuaciones de cinemática directa que describen el comportamiento del brazo robótico.....	35
3.4	Pruebas y análisis de resultados.....	55
	Comparación entre Simulink y Autodesk Inventor.	55
	Comparación entre el análisis de cinemática directa y Simulink.....	56
	Comparación del ángulo de giro ingresado	57
3.5	Manual de simulación del brazo robótico.	58
4	Conclusiones y Recomendaciones.....	59
4.1	Conclusiones.....	59
4.2	Recomendaciones	60
5	Referencias Bibliográficas	61
	ANEXOS	64
	Anexo 1: Certificado de Funcionamiento	i
	Anexo 2: Planos y Esquemas.....	iii
	Anexo 3: Códigos de programación.	vii
	Anexo 4: Pruebas de funcionamiento	xiii

ÍNDICE DE FIGURAS

Figura 1.1 Estructura de un brazo robótico	9
Figura 1.2 (a) Articulación rotacional (1 GDL) (b) Articulación prismática (1 GDL)	10
Figura 1.3 (a) Articulación cilíndrica (2 GDL) (b) Articulación planar (2 GDL) (c) Articulación esférica (3 GDL).....	10
Figura 1.4 Método geométrico	11
Figura 3.1 Brazo robótico Open manipulation.....	20
Figura 3.2 Descarga y selección del formato.....	20
Figura 3.3 Ventana de inicio de Autodesk Inventor.....	21
Figura 3.4 Ventana para abrir archivos.....	21
Figura 3.5 Importación del plano del brazo robótico a Autodesk Inventor.....	22
Figura 3.6 Eslabones y articulaciones del brazo robótico	22
Figura 3.7 Ventana para guardar cada eslabón.....	23
Figura 3.8 Nuevo archivo en Autodesk Inventor.	23
Figura 3.9 Ventana de trabajo de Autodesk Inventor - herramienta Insertar.....	24
Figura 3.10 Ventana para insertar los eslabones.....	24
Figura 3.11 Base del brazo robótico como eslabón fijo	25
Figura 3.12 Ventana de trabajo de Autodesk Inventor - herramienta Restringir.....	25
Figura 3.13 Ventana de la herramienta "Restringir"	26
Figura 3.14 (a) Eje del eslabón 0 (b) Centro del eslabón 1 (c) Resultado del ensamblaje	26
Figura 3.15 Eslabón 2 insertado a la venta de trabajo de Autodesk Inventor	27
Figura 3.16 (a) Eje del eslabón 1 (b) Centro del eslabón 2 (c) Resultado del ensamblaje	27
Figura 3.17 Eslabón 3 insertado a la venta de trabajo de Autodesk Inventor	28
Figura 3.18 (a) Eje del eslabón 2 (b) Centro del eslabón 3 (c) Resultado del ensamblaje	28
Figura 3.19 Eslabón 4 insertado a la venta de trabajo de Autodesk Inventor	29
Figura 3.20 (a) Eje del eslabón 3 (b) Centro del eslabón 4 (c) Resultado del ensamblaje	29
Figura 3.21 Ensamblaje final del brazo robótico	29
Figura 3.22 Librería Simscape Multibody Link y sus componentes.....	30
Figura 3.23 Descarga de smlink.2018b.win64.zip y install_addon.m	30
Figura 3.24 Ejecución de Matlab como administrador	31
Figura 3.25 Ejecución del comando addpath('foldername')	31

Figura 3.26	Ejecución del comando <code>install_addon('zipname')</code>	32
Figura 3.27	Instalación de la librería Simscape Multibody Link.....	32
Figura 3.28	Ejecución del comando <code>smlink_linkinv</code>	33
Figura 3.29	Pestaña complementos	33
Figura 3.30	Ejecución del comando <code>smimport</code>	34
Figura 3.31	Diagrama de bloques obtenido de la importación	34
Figura 3.32	Brazo robótico en 3D	35
Figura 3.33	Sistema de coordenadas en la base del brazo robótico.....	35
Figura 3.34	Eslabón 0 (a) Rotación alrededor del eje z (b) Desplazamiento d_1	36
Figura 3.35	Valor de la distancia d_1	36
Figura 3.36	Rotación alrededor del eje z	37
Figura 3.37	(a) Desplazamiento d_2 (b) Valor de la distancia d_2	37
Figura 3.38	(a) Desplazamiento a_1 (b) Valor de la distancia a_1	38
Figura 3.39	(a) Rotación q_3 y desplazamiento a_2 (b) Valor de la distancia a_2	38
Figura 3.40	(a) Rotación q_3 y desplazamiento a_3 (b) Valor de la distancia a_3	39
Figura 3.41	Sistema de coordenadas para cada articulación y distancias medidas ...	39
Figura 3.42	Panel de bibliotecas de Simulink	43
Figura 3.43	Bloque Matlab Function	44
Figura 3.44	Bloque From Workspace	44
Figura 3.45	Ventana de parámetros del Bloque From Workspace.....	45
Figura 3.46	Bloque To Workspace	45
Figura 3.47	Ventana de parámetros del Bloque To Workspace.....	46
Figura 3.48	Bloque Display.....	46
Figura 3.49	Bloque Simulink-PS Converter	46
Figura 3.50	Ventana de parámetros del Bloque Simulink-Ps Converter	47
Figura 3.51	Bloque PS-Simulink Converter	47
Figura 3.52	Ventana de parámetros del bloque PS-Simulink Converter	48
Figura 3.53	Bloque Transform Sensor.....	48
Figura 3.54	Ventana de parámetros del bloque Transform Sensor.....	49
Figura 3.55	Bloque Gain.....	49
Figura 3.56	Ventana de parámetros del bloque Gain	49
Figura 3.57	Asignación de las entradas, las salidas y el nombre de la función.....	50
Figura 3.58	Parámetros de Denavit Hartenberg d_i, a_i, α_i	50
Figura 3.59	Matriz de transformación en Matlab.....	51
Figura 3.60	Posición del efector final x, y, z	51
Figura 3.61	Ejecución del comando “guide”	52

Figura 3.62 Ventana GUIDE Quick Start	53
Figura 3.63 Interfaz del brazo robótico.	54
Figura 3.64 Uso de la interfaz del brazo robótico.....	54
Figura 3.65 Código QR de la simulación del brazo robótico	58
Figura A4. 1 Prueba N°1 (a) Valor obtenido en Autodesk Inventor (b) Valor obtenido en Simulink.....	xiv
Figura A4. 2 Prueba N°2 (a) Valor obtenido en Autodesk Inventor (b) Valor obtenido en Simulink.....	xiv
Figura A4. 3 Prueba N°3 (a) Valor obtenido en Autodesk Inventor (b) Valor obtenido en Simulink.....	xv
Figura A4. 4 Prueba N°4 (a) Valor obtenido en Autodesk Inventor (b) Valor obtenido en Simulink.....	xv
Figura A4. 5 Prueba N°1 (a) Posición x (b) Posición y (c) Posición z	xvi
Figura A4. 6 Prueba N°2 (a) Posición x (b) Posición y (c) Posición z	xvii
Figura A4. 7 Prueba N°3 (a) Posición x (b) Posición y (c) Posición z	xviii
Figura A4. 8 Prueba N°4 (a) Posición x (b) Posición y (c) Posición z	xix
Figura A4. 9 Valores giro ingresados en la Interfaz para la prueba N°1	xx
Figura A4. 10 Valores de giro medidos en Simulink referente a la prueba N°1	xx
Figura A4. 11 Valores giro ingresados en la Interfaz para la prueba N°2	xx
Figura A4. 12 Valores de giro medidos en Simulink referente a la prueba N°2	xxi
Figura A4. 13 Valores giro ingresados en la Interfaz para la prueba N°3	xxi
Figura A4. 14 Valores de giro medidos en Simulink referente a la prueba N°3	xxi
Figura A4. 15 Valores giro ingresados en la Interfaz para la prueba N°4	xxii
Figura A4. 16 Valores de giro medidos en Simulink referente a la prueba N°4	xxii

ÍNDICE DE TABLAS

Tabla 1.1 Ventajas y desventajas del método geométrico y metodología de Denavit Hartenberg	12
Tabla 3.1 Características de los brazos robóticos.....	18
Tabla 3.2 Criterios y ponderación para selección del brazo robótico	19
Tabla 3.3 Matriz para la selección del brazo robótico.	19
Tabla 3.4 Parámetros de Denavit-Hartenberg.....	39
Tabla 3.5 Parámetros de bloque From Workspace	44
Tabla 3.6 Parámetros del bloque To Workspace	45
Tabla 3.7 Parámetros del bloque Simulink-Ps Converter	47
Tabla 3.8 Parámetros del bloque Ps-Simulink Converter	48
Tabla 3.9 Parámetros del bloque Transform Sensor.....	48
Tabla 3.10 Componentes para crear la interfaz	53
Tabla 3.11 Prueba entre Autodesk Inventor y Simulink de Matlab	55
Tabla 3.12 Prueba en Simulink de Matlab.....	56
Tabla 3.13 Prueba del ángulo de giro ingresado a cada articulación	57

RESUMEN

En el presente trabajo de titulación se realiza la simulación de un brazo robótico con 4 grados de libertad (GDL), para conocer la posición de su efector final mediante el análisis de cinemática directa en el entorno Simulink de Matlab.

En el capítulo 1 se encuentran la introducción, el objetivo general y los objetivos específicos de este trabajo de titulación. También se presenta de manera general los conceptos fundamentales de brazo robótico, eslabón, articulación, cinemática directa, método geométrico y metodología de Denavit Hartenberg.

En el capítulo 2 se presenta la metodología utilizada donde se expone la selección del brazo robótico, así mismo el modelamiento y ensamblaje de éste. Además, se exhibe el análisis de cinemática directa y las pruebas de funcionamiento.

En el capítulo 3 se describe el análisis de mercado y los criterios selección para escoger el brazo robótico. Conjuntamente, se incluye el procedimiento para el ensamblaje de los 5 eslabones del brazo robótico, también se presentan los parámetros de Denavit Hartenberg que permitieron obtener 10 ecuaciones de transformación. Además, se describen las tres pruebas de funcionamiento de la simulación del brazo robótico en las cuales se obtuvo un error relativo porcentual máximo de 1.10 % entre las pruebas realizadas.

Finalmente, en el capítulo 4 se encuentran las conclusiones y recomendaciones derivadas del funcionamiento de la simulación del brazo robótico en Simulink.

PALABRAS CLAVE: Simulink, Matlab, Inventor, Brazo robótico, GDL.

ABSTRACT

In the present work, the simulation of a robotic arm with 4 degrees of freedom (GDL) is carried out in order to know the position of its end effector by means of direct kinematics analysis in the Simulink environment of Matlab.

Chapter 1 contains the introduction, the general objective and the specific objectives of this degree work. Also, the fundamental concepts of robotic arm, link, articulation, direct kinematics, geometric method and Denavit Hartenberg methodology are presented in a general way.

Chapter 2 presents the methodology used, where the selection of the robotic arm, as well as its modeling and assembly are presented. In addition, the direct kinematics analysis and the functional tests are shown.

Chapter 3 describes the market analysis and the selection criteria to choose the robotic arm. Together, the procedure for the assembly of the 5 links of the robotic arm is included, and the Denavit Hartenberg parameters that allowed obtaining 10 transformation equations are also presented. In addition, the three performance tests of the simulation of the robotic arm are described, in which a maximum relative percentage error of 1.10 % was obtained among the tests performed.

Finally, Chapter 4 contains the conclusions and recommendations derived from the operation of the simulation of the robotic arm in Simulink.

KEY WORDS: Simulink, Matlab, Inventor, robotic arm, GDL.

1 INTRODUCCIÓN

En la actualidad, la robótica es una de las áreas de mayor crecimiento en el mundo en cualquier tipo de industria, puesto que los robots o manipuladores son de suma importancia en las líneas de producción, ya que facilitan la fabricación de productos con características repetitivas; y, permiten incrementar la producción, calidad del producto y eficiencia de trabajo. El conocimiento de la robótica es necesario para profesionales en el área de electromecánica, debido a que la industria se encuentra muy desarrollada y robotizada. En la actualidad la programación de brazos robóticos se la realiza mediante herramientas de simulación que reproducen el comportamiento de un robot, para validar las decisiones del diseño conceptual y los movimientos programados, así como para capacitar al personal técnico que lo utiliza previo a su operación y mantenimiento [6].

Debido a lo anteriormente mencionado y tomando en cuenta el perfil laboral expuesto por la Escuela de Formación de Tecnólogos (ESFOT) de la Escuela Politécnica Nacional (EPN), donde se menciona que el profesional graduado debe contar con conocimientos, habilidades y capacidades técnicas, que le permitan desenvolverse en el campo industrial. Es importante que los futuros egresados de la carrera de Tecnología en Electromecánica de la ESFOT cuenten en su formación profesional con conocimientos relacionados a las aplicaciones de la Robótica; como por ejemplo: robots industriales, robots paralelos orientados a la rehabilitación física, robots para cirugías médicas, robots en la exploración espacial, etc. Es indiscutible la importancia de formar a los tecnólogos en la simulación y análisis de sistemas mecánicos, esto debido al desarrollo de la automatización, controladores e inteligencia artificial.

A través de este proyecto de titulación se complementa la formación académica de un tecnólogo, ya que se realizó un manual de simulación de un brazo robótico con 4 GDL, en el cual se redacta de forma detallada el proceso para la simulación del brazo robótico usando Simulink de Matlab.

En el Anexo 1 se adjunta el certificado de funcionamiento, el cual abala que este proyecto de titulación se encuentra operando de forma correcta.

1.1 Objetivo general

Simular en el entorno Simulink de Matlab un brazo robótico de 4 grados de libertad (GDL).

1.2 Objetivos específicos

- Obtener las características y especificaciones técnicas de diferentes brazos robóticos.
- Establecer el modelo del brazo robótico que permita la simulación en el entorno de trabajo de Simulink de Matlab.
- Obtener las ecuaciones de cinemática directa que describen el comportamiento del brazo robótico.
- Realizar pruebas de funcionamiento de la simulación del brazo robótico.
- Realizar un manual de la simulación del brazo robótico.

1.3 Fundamentos

Brazo robótico

Es un mecanismo serie o de cadena abierta que está formado por: una base, articulaciones, eslabones y su efector final (Figura 1.1).

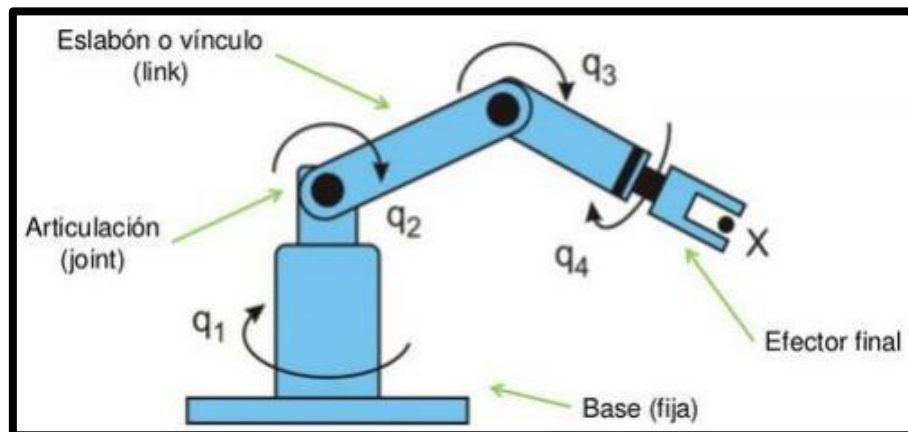


Figura 1.1 Estructura de un brazo robótico [22]

Eslabón: Es un cuerpo rígido que forma la estructura física de un brazo robótico, que se encuentran separados por articulaciones.

Articulación: Es un componente físico que une dos eslabones, permitiendo el movimiento relativo entre dichos eslabones [7]. En la Figura 1.2 se muestran dos tipos de articulaciones que generan un grado libertad:

- **Articulación prismática:** permite el desplazamiento de un eslabón con respecto a otro a lo largo del eje de la articulación [7].
- **Articulación rotacional:** gira entorno al eje de un eslabón con respecto a otro.

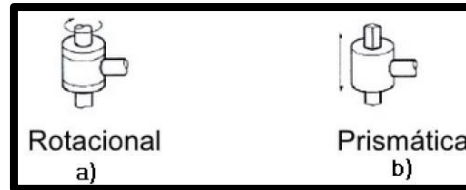


Figura 1.2 (a) Articulación rotacional (1 GDL) (b) Articulación prismática (1 GDL) [7]

Adicionalmente hay articulaciones que permiten tener dos grados de libertad de las cuales se puede resaltar las articulaciones: cilíndrica y planar (Figura 1.3). Finalmente se tiene una articulación que en una sola estructura física se puede generar tres movimientos, también conocida como articulación esférica [7].

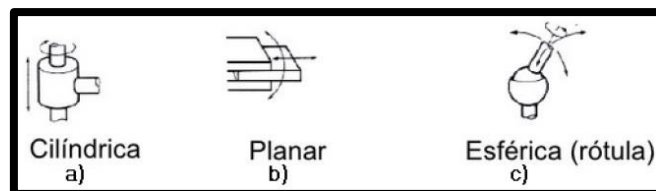


Figura 1.3 (a) Articulación cilíndrica (2 GDL) (b) Articulación planar (2 GDL) (c) Articulación esférica (3 GDL) [7]

Efecto final: Son dispositivos que interactúan directamente con el entorno del brazo robótico y son diseñados para un tipo de trabajo en particular. Según la ISO 8373, “es un objeto unido a la interfaz mecánica dispuesta en el extremo del brazo robótico, para dotar a éste de la funcionalidad necesaria para realizar su tarea”. Pueden ser: elementos de sujeción o herramientas [23].

Cinemática directa

La Cinemática directa estudia la posición y orientación del efecto final, con respecto a un sistema de coordenadas donde se toma como referencia los ángulos de las articulaciones y parámetros geométricos del brazo robótico [8].

Existen varios métodos que permiten resolver el análisis de cinemática directa, los cuales son:

Geométrico: Es un método no sistemático que utiliza relaciones geométricas que permiten conocer la posición del efecto final y la posición de sus articulaciones (Figura 1.4).

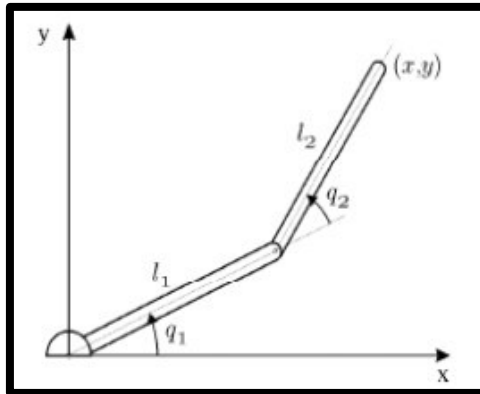


Figura 1.4 Método geométrico [22]

Metodología de Denavit Hartenberg (D-H): La representación de D-H, se aplica a robots de cadena cinemática abierta y consiste en una serie de reglas para colocar los sistemas de referencia de cada eslabón del robot [3].

Este método tiene un modelo de matriz estándar (T_n), que considera los siguientes parámetros: los ángulos de rotación, la distancia de los eslabones y los ejes de coordenadas colocado en cada uno de ellos, los grados de libertad, es importante conocer que el eje de coordenadas tomado como referencia se encuentra en la base, obteniendo de esta manera las denotaciones x, y, z . La matriz estándar está conformada por varias matrices de transformación: $Rot(Z_{n-1}, \theta_n), Trans(Z_{n-1}, d_n), Trans(X_n, a_n), Rot(X_n, \alpha_n)$.

La matriz estándar de Denavit Hartenberg tiene la siguiente estructura:

$$T_n = Rot(Z_{n-1}, \theta_n) \cdot Trans(Z_{n-1}, d_n) \cdot Trans(X_n, a_n) \cdot Rot(X_n, \alpha_n)$$

Ecuación 1.1 Matriz estándar de Denavit Hartenberg [5]

Donde:

$Rot(Z_{n-1}, \theta_n)$: Rotación en el eje z con un ángulo teta (θ).

$$Rot(Z_{n-1}, \theta_n) = \begin{bmatrix} \cos(\theta_n) & -\sin(\theta_n) & 0 & 0 \\ \sin(\theta_n) & \cos(\theta_n) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Ecuación 1.2 Matriz rotación ángulo teta [5]

$Trans(Z_{n-1}, d_n)$: Traslación de una distancia d en el eje z (d_n).

$$Trans(Z_{n-1}, d_n) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_n \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Ecuación 1.3 Matriz traslación distancia d [5]

$Trans(X_n, a_n)$: Traslación de una distancia a en el eje x (a_n).

$$Trans(x_n, a_n) = \begin{bmatrix} 1 & 0 & 0 & a_n \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_n \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Ecuación 1.4 Traslación distancia a [5]

$Rot(X_n, \alpha_n)$: Rotación en el eje x de un ángulo alfa (α_n).

$$Rot(X_n, \alpha_n) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\alpha_n) & \sin(\alpha_n) & 0 \\ 0 & \sin(\alpha_n) & \cos(\alpha_n) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Ecuación 1.5 Matriz rotación ángulo alfa [5]

La Ecuación 1.6 es la matriz resultante que contiene los parámetros de Denavit Hartenberg para el análisis del brazo robótico.

$$T_n = \begin{bmatrix} \cos(\theta_n) & -\sin(\theta_n) \cdot \cos(\alpha_n) & \sin(\theta_n) \cdot \cos(\alpha_n) & a_n \cdot \cos(\theta_n) \\ \sin(\theta_n) & \cos(\theta_n) \cdot \cos(\alpha_n) & -\cos(\theta_n) \cdot \sin(\alpha_n) & a_n \cdot \sin(\theta_n) \\ 0 & \sin(\alpha_n) & \cos(\alpha_n) & d_n \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Ecuación 1.6 Matriz estándar resultante de Denavit Hartenberg [5]

En la Tabla 1.1 se muestran las ventajas y desventajas de los métodos descritos anteriormente.

Tabla 1.1 Ventajas y desventajas del método geométrico y metodología de Denavit Hartenberg

Geométrico	Metodología de Denavit Hartenberg
Ventajas	Ventajas
<ul style="list-style-type: none"> • Se conoce la posición de cada una de las articulaciones del robot. • Se obtiene la posición y orientación del extremo del robot. 	<ul style="list-style-type: none"> • Se representa la traslación y rotación entre distintos eslabones, que se encuentren consecutivos. • Se obtienen los parámetros de manera directa, al conocer como actúan sus articulaciones. • Es un método sistemático bien desarrollado.

Geométrico	Metodología de Denavit Hartenberg
Desventajas	Desventajas
<ul style="list-style-type: none"> • No es un método sistemático. • Es adecuado para robots con pocos grados de libertad. 	<ul style="list-style-type: none"> • Las características constructivas del robot influyen al realizar el análisis.

De la Tabla 1.1, se aprecia que la Metodología de Denavit Hartenberg es la idónea a utilizarse en este proyecto, por el número de articulaciones y eslabones que tiene el brazo robótico.

2 METODOLOGÍA

2.1 Descripción de la metodología usada

Se realizó un análisis de mercado de los brazos robóticos en diferentes plataformas, para lo cual se tomó en cuenta características importantes como son: el tipo de actuadores, dimensiones de planos CAD, el valor comercial, etc. En base a las características y especificaciones técnicas se empleó el método de matriz de ponderación, la cual permitió seleccionar el brazo robótico Open manipulation modelo rm-x52.

Una vez seleccionado el mecanismo, se empleó el software de dibujo y diseño Autodesk Inventor para realizar cambios (color, textura, etc.) a los mecanismos del brazo robótico para una mejor apreciación visual. Se formaron cinco eslabones con cada mecanismo para facilitar el ensamblaje; además, se realizaron restricciones del tipo insertar que permitieron que el brazo robótico tenga un correcto movimiento en cualquiera de sus ejes.

Con el ensamblaje del brazo robótico finalizado, se inició la descarga de la librería Simscape Multibody Link, luego se realizó la vinculación de Autodesk Inventor con Simulink de Matlab, mediante esta vinculación se logró exportar el ensamblaje del brazo robótico, con el fin de obtener su modelamiento en el entorno de trabajo Simulink de Matlab.

Además, se analizó la cinemática directa del brazo robótico de 4 GDL, mediante la metodología de Denavit Hartenberg donde se utilizaron las matrices de rotación y traslación, con la finalidad de generar ecuaciones cinemáticas que al multiplicarlas permiten obtener la matriz estándar de Denavit Hartenberg la misma que indica cual es la posición del efector final y de esta manera conseguir la simulación propuesta.

Simulink proporciona un entorno gráfico al usuario que facilita el análisis y simulación de sistemas mecánicos incluyendo el análisis de cinemática. Los sistemas mecánicos y el análisis de cinemática directa del brazo robótico son representados mediante bloques en los cuales se emplea un lenguaje de programación propio (lenguaje M) el cual está basado en programas como Visual Basic o Visual C++, permitiendo crear un HMI para controlar la simulación del brazo robótico.

Se realizaron las pruebas de funcionamiento para simular el movimiento del brazo robótico y verificar la posición que alcanza el efector final. Con los datos obtenidos de

las pruebas se calculó el error relativo porcentual existente entre el valor medido y el valor calculado.

Finalmente, se presenta de forma detallada el proceso para simular el brazo robótico desde el HMI como es el ingresar los ángulos de giro y observar de manera gráfica la posición del efector final, permitiendo al usuario familiarizarse con el mecanismo modelado.

3 RESULTADOS Y DISCUSIÓN

En este proyecto se analizó la cinemática directa de un brazo robótico de 4 GDL, mediante la metodología de Denavit Hartenberg, con la finalidad de generar las ecuaciones cinemáticas que permitan realizar la simulación del movimiento del mecanismo y verificar las coordenadas x , y , y z que alcanza el efector final.







Finalmente, se realizó la comparación entre la posición del efector final del brazo robótico simulado y la obtenida del análisis cinemático.

3.1 Características y especificaciones técnicas de diferentes brazos robóticos.

En el mercado industrial existen diferentes compañías que se dedican a la comercialización de brazos robóticos a gran escala, los cuales tienen un costo elevado debido a sus aplicaciones como son el transporte de materiales, corte mecánico, manipulación de sustancias, etc. Esto es considerado un inconveniente para ser utilizados en empresas pequeñas o en temas de investigación, por este motivo se han abierto nuevos mercados que brindan un costo accesible.

En este proyecto se determina la posición del efector final de un brazo robótico de 4 GDL en un entorno virtual, para lo que se realizó un estudio de mercado para identificar diferentes estructuras de hardware abierto y comerciales de bajo costo. En la Tabla 3.1 se muestran 6 tipos de manipuladores con sus características.

Tabla 3.1 Características de los brazos robóticos.

Brazos Robóticos		PhantomX Pincher Robot Arm Kit Mark II - Turtlebot Arm	PincherX 100 Robot Arm	PhantomX Reactor Robot Arm Kit	PincherX 150 Robot Arm	DIY Arduino Robot Arm	Open manipulation
Características							
Actuadores	N°	5	5	7	8	3	4
	Nombre	AX-12A Dynamixel Actuators	PincherX 100 Robot Arm	PhantomX Reactor Robot Arm Kit	XL430-W250-T	Servomotor MG996R	XL430
	Precio	44.90	44.90	44.90	44.90	26.99	49.90
	Tipo	Dynamixel	Dynamixel	Dynamixel	Dynamixel	MG996R	Dynamixel
	Wattios	-	250	-	-	-	250
	RPM	59	57	59	59	-	57
	Fuerza	1.52	1.40	1.52	1.52	0.88 – 1.07	1.40
Dimensiones	Planos CAD	Si	Si	Si	Si	Si	Si
Precio total	Actuadores	224.50	249.50	314.30	399.20	80.97	-
	Brazo robótico	379.95	549.95	599.95	949.45	-	230.00 sin actuadores
Web	Catálogos	https://acortar.link/eufrM	https://acortar.link/VrLyF	https://acortar.link/mQDPH	https://acortar.link/mQDPH	https://acortar.link/cM4bf	https://acortar.link/85g2F
Imagen de referencia							

Se empleó el método de matriz de ponderación para seleccionar el brazo robótico para este proyecto.

En la Tabla 3.2 se muestran los criterios de selección con la ponderación que le corresponde a cada uno.

Tabla 3.2 Criterios y ponderación para selección del brazo robótico

Criterio	Descripción	Ponderado
1	Alcance máximo del manipulador robótico	0.20
2	Capacidad de carga máxima del manipulador	0.10
3	Sistema modular expansión y modificación	0.20
4	Grados de libertad	0.05
5	Material resistente y liviano	0.10
6	Información sobre dimensiones de partes y parámetros del brazo robótico	0.25
7	Costo de la plataforma robótica	0.10
Total		1.00

En la Tabla 3.3 se muestra la matriz de selección que considera cada criterio.

Tabla 3.3 Matriz para la selección del brazo robótico.

Brazo robótico	Criterio							Total
	1	2	3	4	5	6	7	
PhantomX Pincher Robot Arm Kit Mark II	0.60	0.30	0.60	0.15	0.40	1.25	0.40	3.70
PincherX 100 Robot Arm	1.00	0.50	0.00	0.25	0.50	1.25	0.00	3.50
PhantomX Reactor Robot Arm Kit	0.80	0.40	0.60	0.25	0.50	0.75	0.00	3.30
PincherX 150 Robot Arm	0.60	0.40	0.60	0.15	0.40	1.25	0.40	4.00
DIY Arduino Robot Arm	0.60	0.30	0.60	0.15	0.40	1.25	0.30	3.60
Open manipulation	0.80	0.30	1.00	0.15	0.40	1.25	0.30	4.20

De la Tabla 3.3 se aprecia que la mejor alternativa es la del brazo robótico Open manipulation del fabricante Robosavvy modelo rm-x52 (Figura 3.1).



Figura 3.1 Brazo robótico Open manipulation [1]

El brazo robótico Open manipulation tiene la siguiente estructura:

- 4 articulaciones rotaciones (GDL).
- Una base fija.
- 5 eslabones incluido el efector final.

3.2 Modelo del brazo robótico que permite la simulación en el entorno de trabajo de Simulink de Matlab

Descarga e importación del plano

Se realizó la descarga del plano en 3D del brazo robótico seleccionado, en un formato de archivo “Step” (Figura 3.2).

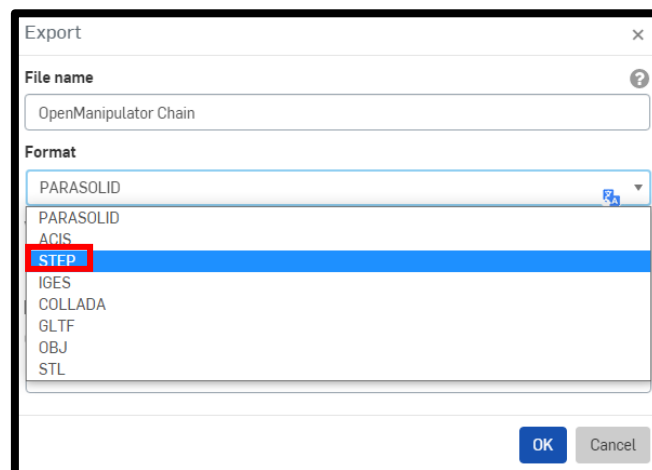


Figura 3.2 Descarga y selección del formato [1]

Finalizada la descarga, se abrió el programa Autodesk Inventor y en la ventana de inicio se escogió la opción “Abrir” (Figura 3.3).

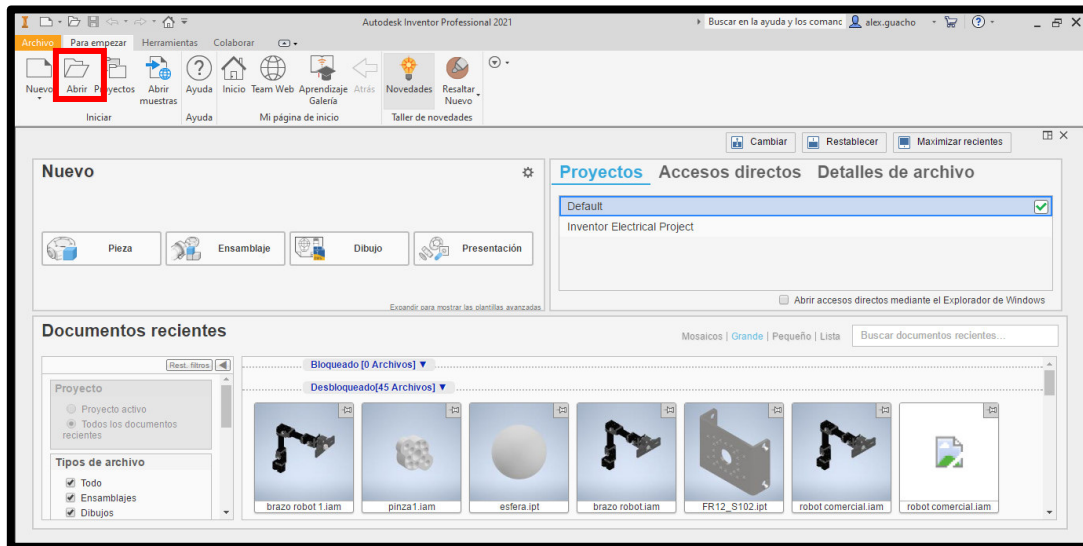


Figura 3.3 Ventana de inicio de Autodesk Inventor

Luego aparece una ventana para cambiar el tipo de archivo a formato “Step” y buscar el plano del brazo robótico descargado (Figura 3.4).

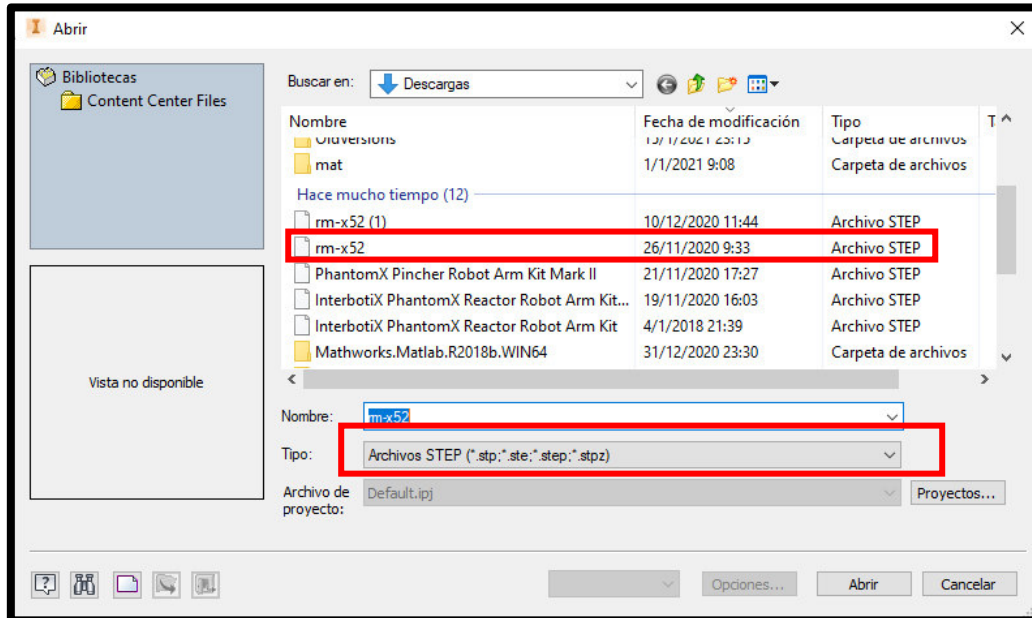


Figura 3.4 Ventana para abrir archivos

Finalmente, en la ventana de trabajo de Autodesk Inventor se observa la estructura del brazo robótico (Figura 3.5).

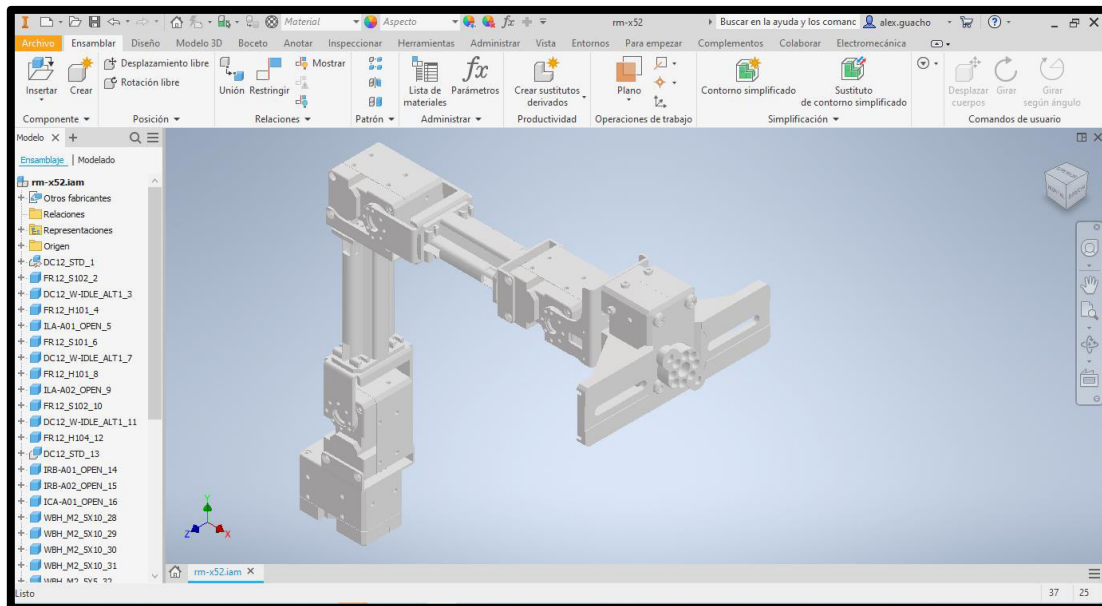


Figura 3.5 Importación del plano del brazo robótico a Autodesk Inventor

Conformación de eslabones.

En la Figura 3.6 se muestran los eslabones y articulaciones del brazo robótico; además, se realizaron cambios de color y de estructura del brazo robótico, con el fin de mejorar la apreciación visual.

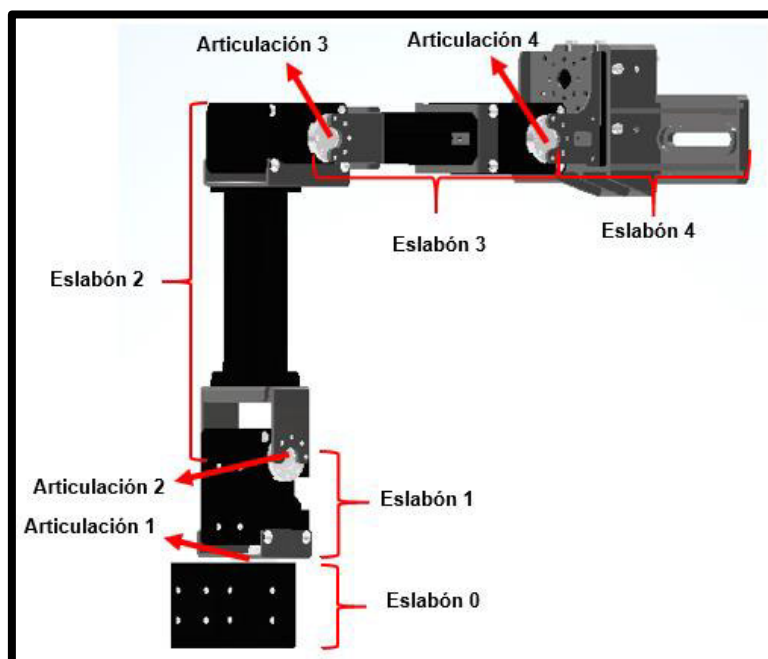


Figura 3.6 Eslabones y articulaciones del brazo robótico

Luego se guardaron los eslabones como piezas individuales (Figura 3.7).

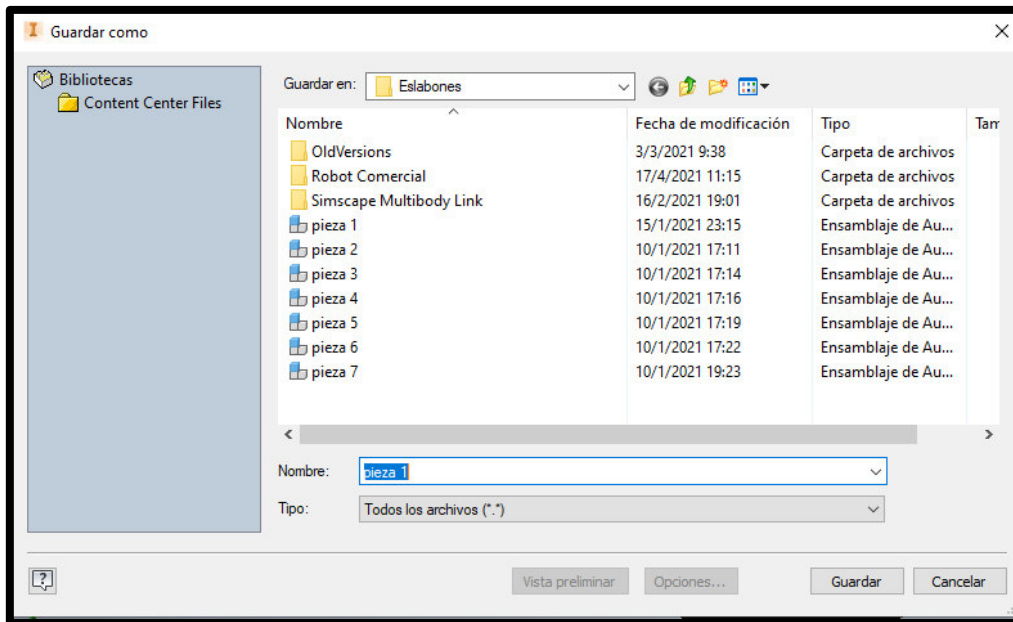


Figura 3.7 Ventana para guardar cada eslabón

Importación de los eslabones.

Se abrió un nuevo archivo en Autodesk Inventor, se escoge la opción “Normal.iam” (Figura 3.8).

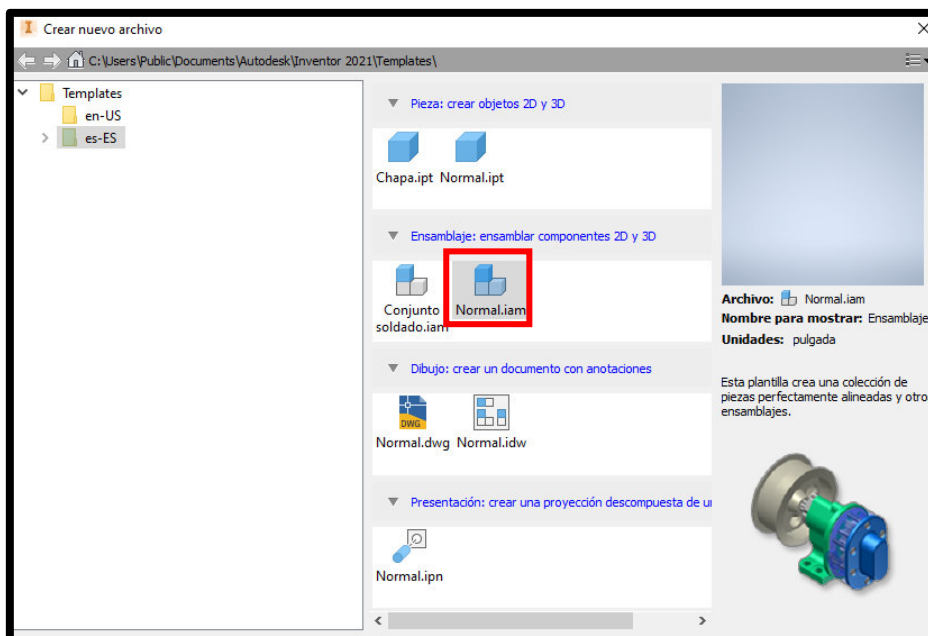


Figura 3.8 Nuevo archivo en Autodesk Inventor.

Después en la ventana de trabajo de Autodesk Inventor, en la pestaña “Ensamblar” se selecciona la herramienta “Insertar” (Figura 3.9).

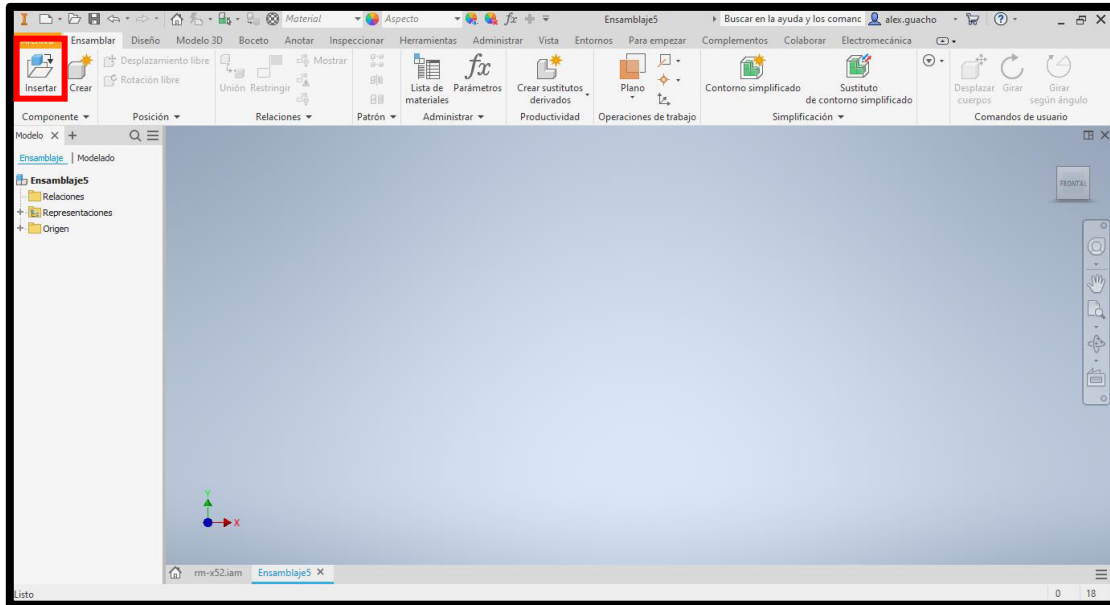


Figura 3.9 Ventana de trabajo de Autodesk Inventor - herramienta Insertar

A continuación, aparece una ventana para seleccionar el eslabón a insertar (Figura 3.10).

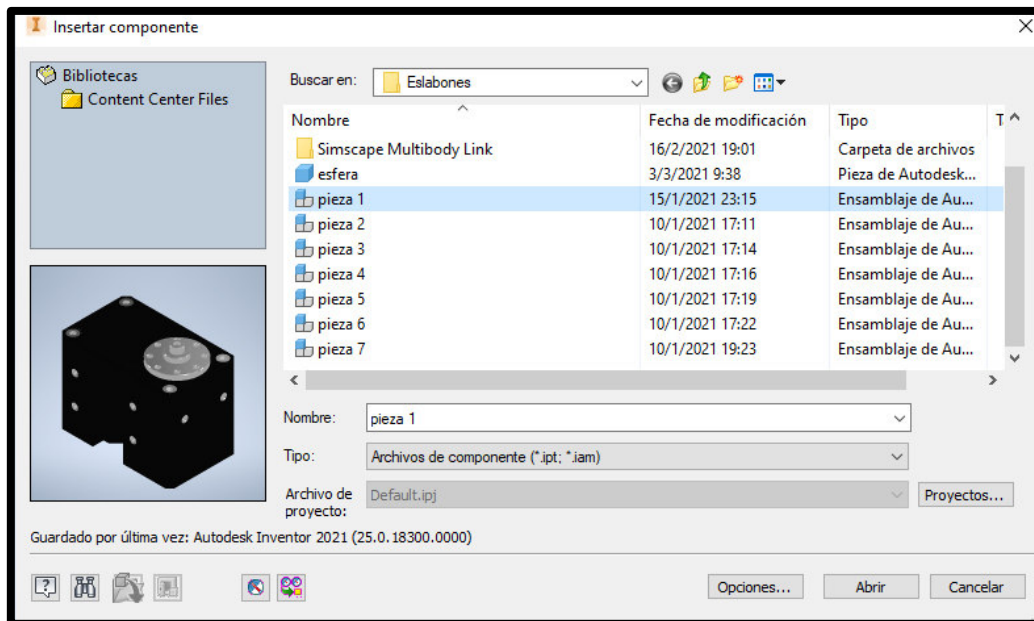


Figura 3.10 Ventana para insertar los eslabones

Ensamblaje del brazo robótico.

Se consideró la base del brazo robótico como el eslabón 0, por lo que se procedió a fijarla, haciendo clic derecho sobre el eslabón y luego seleccionando la opción "Fijo" (Figura 3.11).

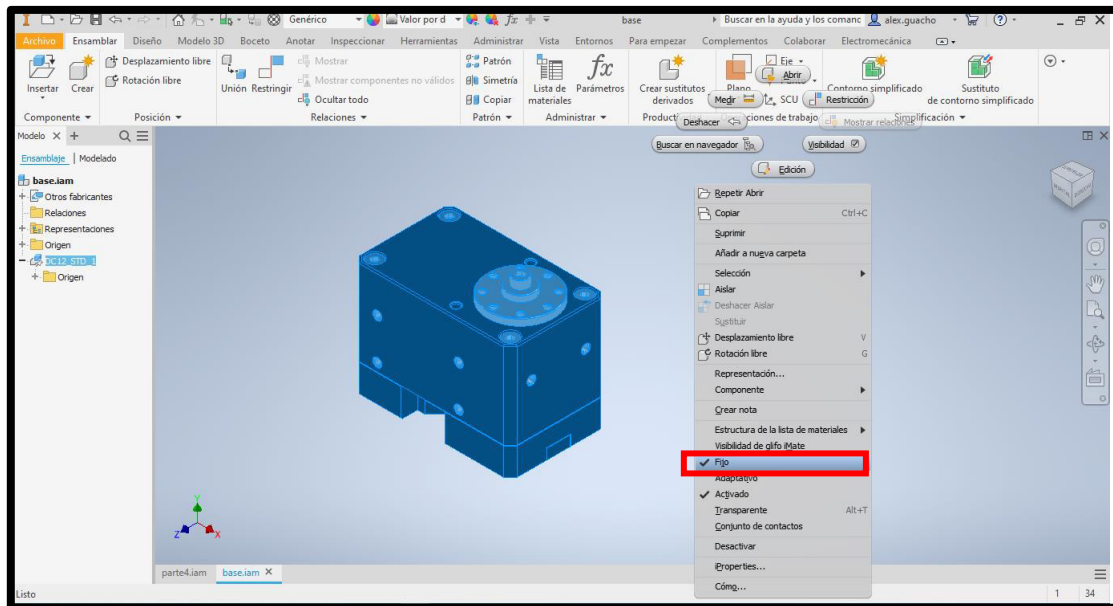


Figura 3.11 Base del brazo robótico como eslabón fijo

Ensamblaje 1

Se insertó el eslabón 1, luego en la pestaña “Ensamblar” se escogió la herramienta “Restringir” (Figura 3.12).

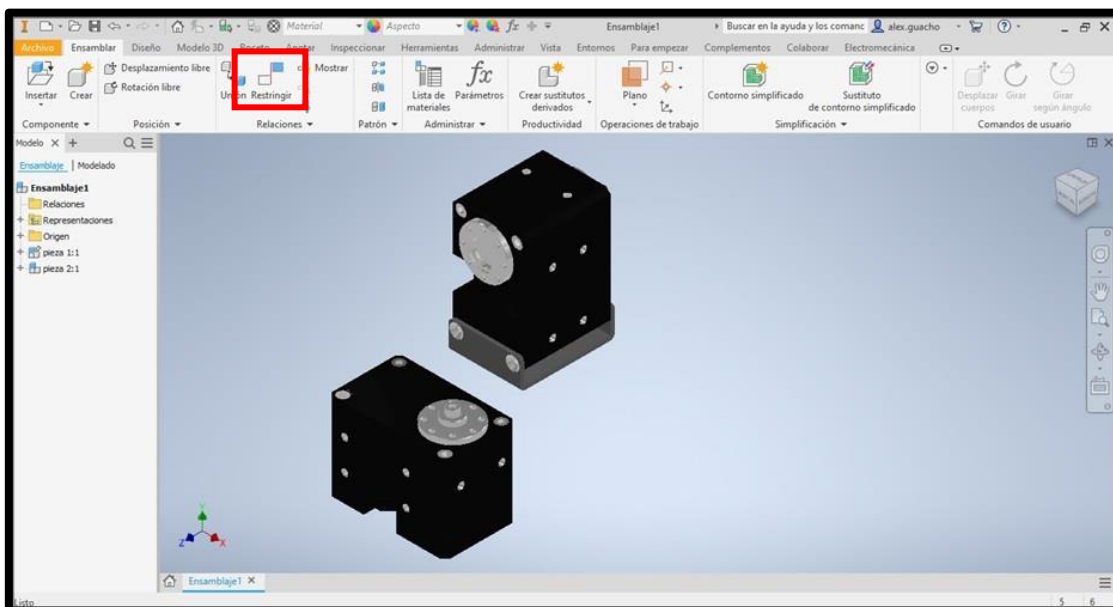


Figura 3.12 Ventana de trabajo de Autodesk Inventor - herramienta Restringir

Aparece la ventana “Añadir restricción”, de la cual se escogió la opción “Tipo – Insertar” (Figura 3.13). Esta restricción se la utilizó para ensamblar todos los eslabones del brazo robótico.

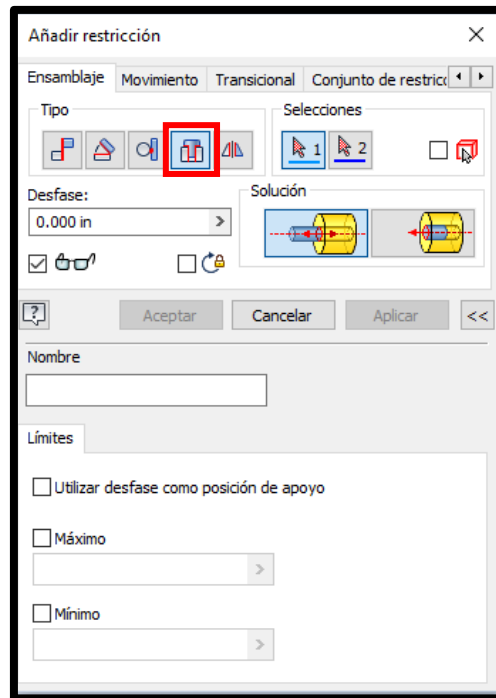


Figura 3.13 Ventana de la herramienta "Restringir"

A continuación, se seleccionó el eje del eslabón 0 y la base del eslabón 1 (Figura 3.14).

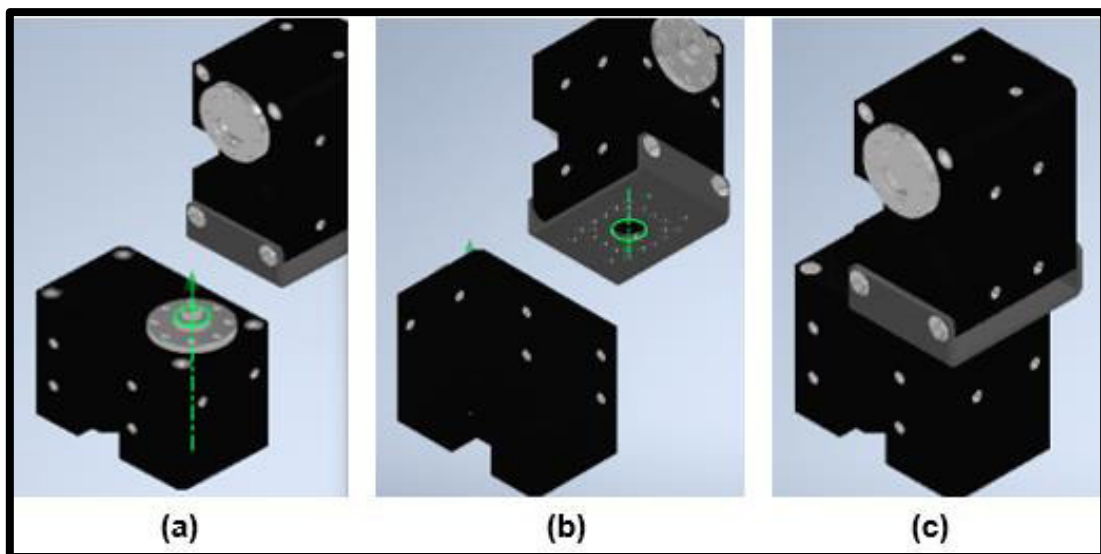


Figura 3.14 (a) Eje del eslabón 0 (b) Centro del eslabón 1 (c) Resultado del ensamblaje

Ensamblaje 2

Se insertó el eslabón 2, luego se utilizó la herramienta "Restringir" y se escogió la opción "Tipo - Insertar" (Figura 3.15).

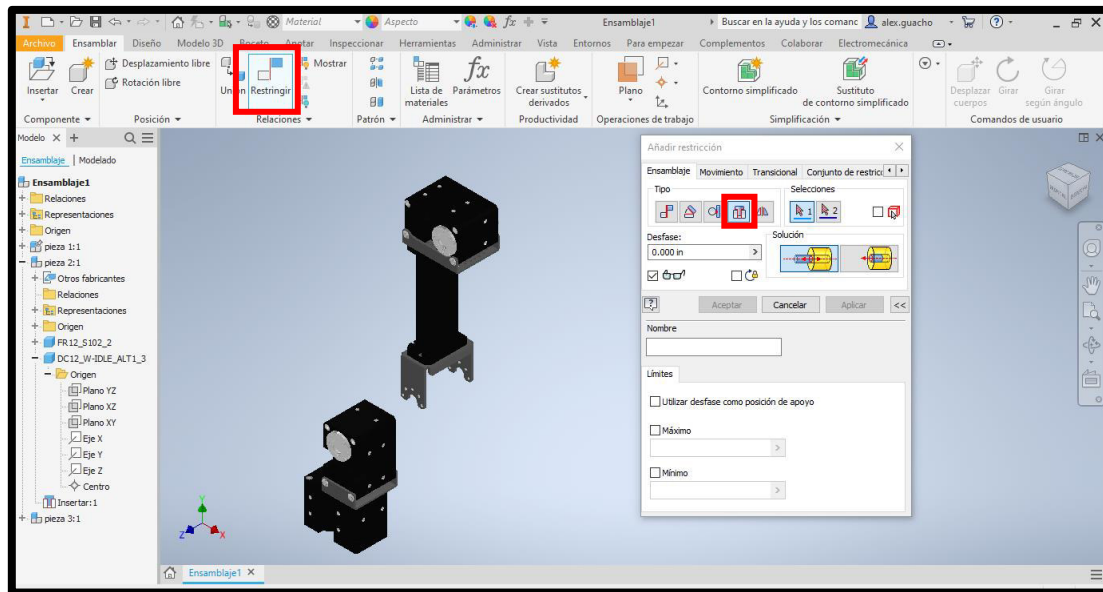


Figura 3.15 Eslabón 2 insertado a la venta de trabajo de Autodesk Inventor

Después se seleccionó el eje del eslabón 1 y el centro del eslabón 2 (Figura 3.16).

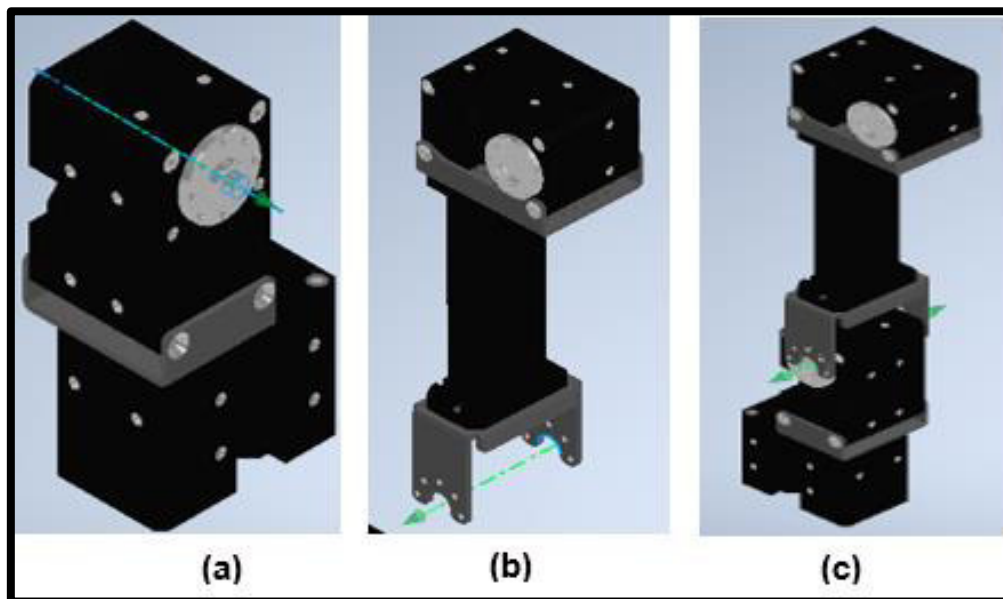


Figura 3.16 (a) Eje del eslabón 1 (b) Centro del eslabón 2 (c) Resultado del ensamblaje

Ensamblaje 3

Se insertó el eslabón 3, luego se utilizó la herramienta “Restringir” y se escogió la opción “Tipo - Insertar” (Figura 3.17).

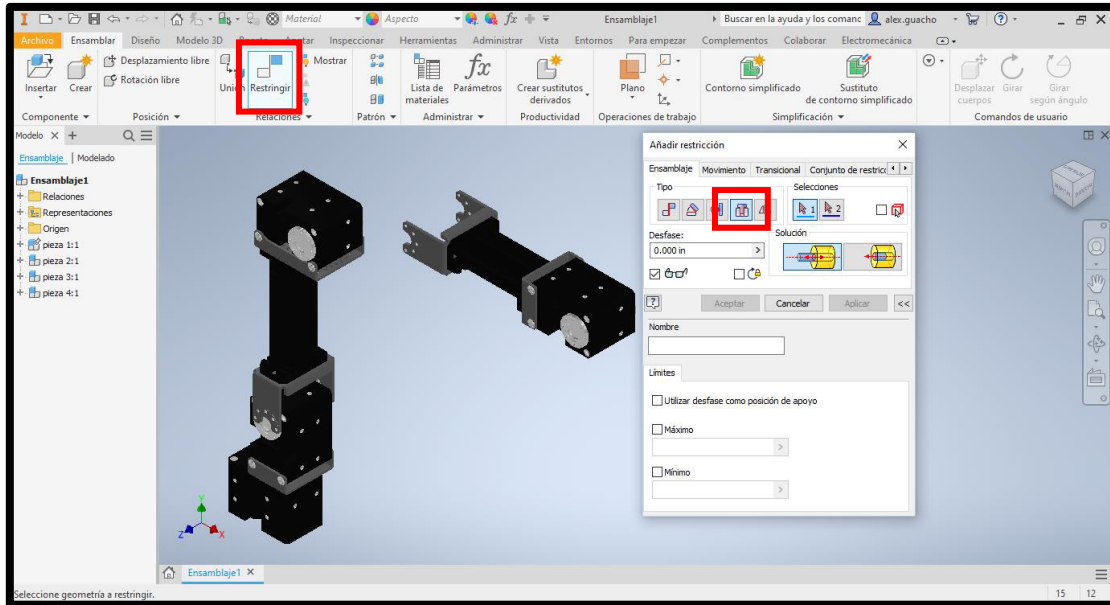


Figura 3.17 Eslabón 3 insertado a la venta de trabajo de Autodesk Inventor

Después se seleccionó el eje del eslabón 2 y el centro del eslabón 3 (Figura 3.18).

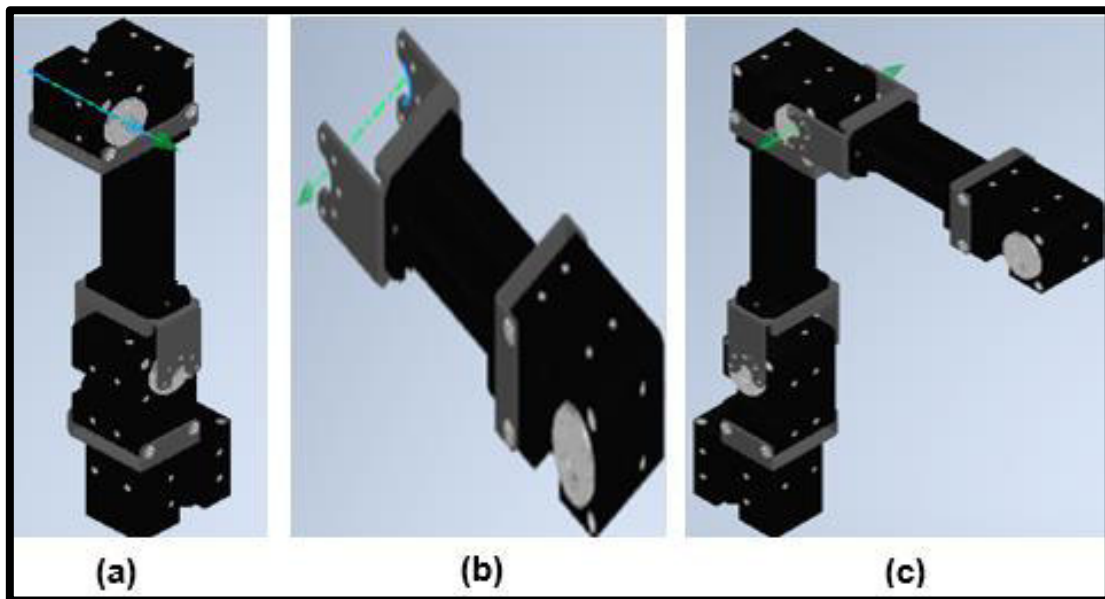


Figura 3.18 (a) Eje del eslabón 2 (b) Centro del eslabón 3 (c) Resultado del ensamblaje

Ensamblaje 4

Se insertó el eslabón 4, luego se utilizó la herramienta “Restringir” y se seleccionó la opción “Tipo - Insertar” (Figura 3.19).

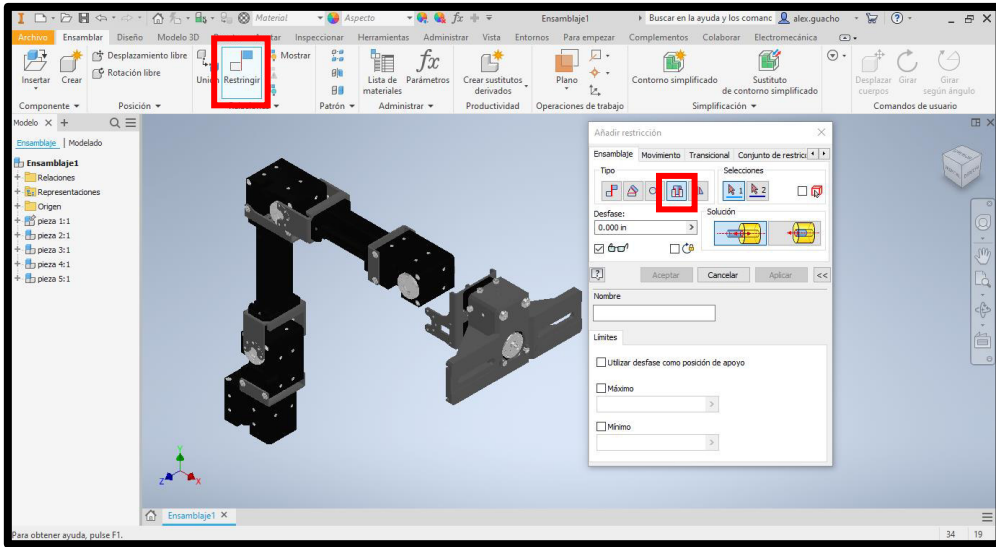


Figura 3.19 Eslabón 4 insertado a la venta de trabajo de Autodesk Inventor

Después se seleccionó el eje del eslabón 3 y el centro del eslabón 4 (Figura 3.20).

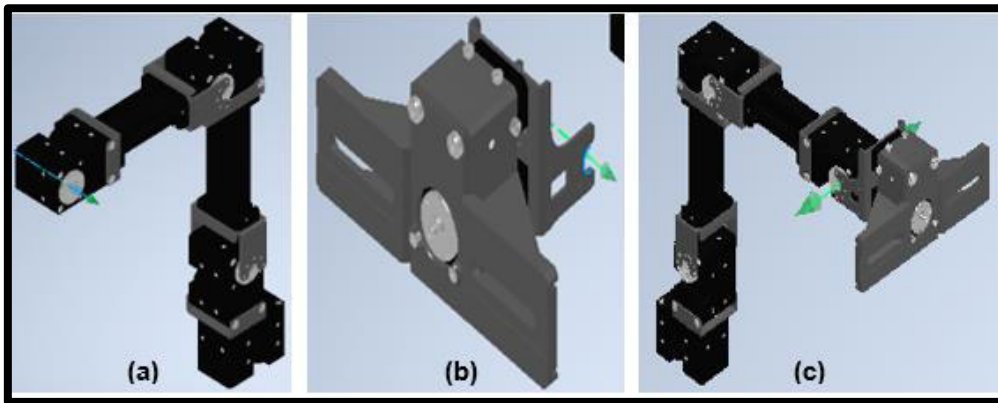


Figura 3.20 (a) Eje del eslabón 3 (b) Centro del eslabón 4 (c) Resultado del ensamblaje

Finalmente, se obtuvo el ensamblaje del brazo robótico en 3D (Figura 3.21).

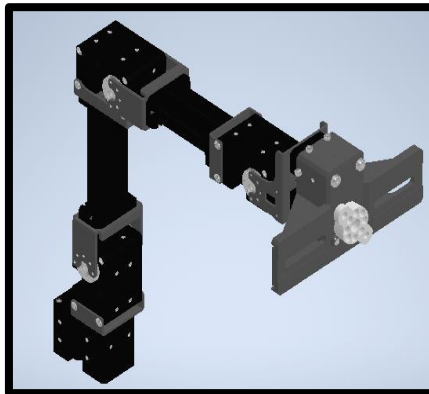


Figura 3.21 Ensamblaje final del brazo robótico

Descarga de Simscape Multibody Link

Se ingresó al enlace <http://bit.ly/SimscapeMultibodyLink>, para descargar la librería Simscape Multibody Link y sus componentes.

Se seleccionó “Simscape Multibody Link 6.0 – Release 2018b (Simscape Multibody 6.0)” debido a la versión de Matlab que se utilizó para este proyecto (Figura 3.22)

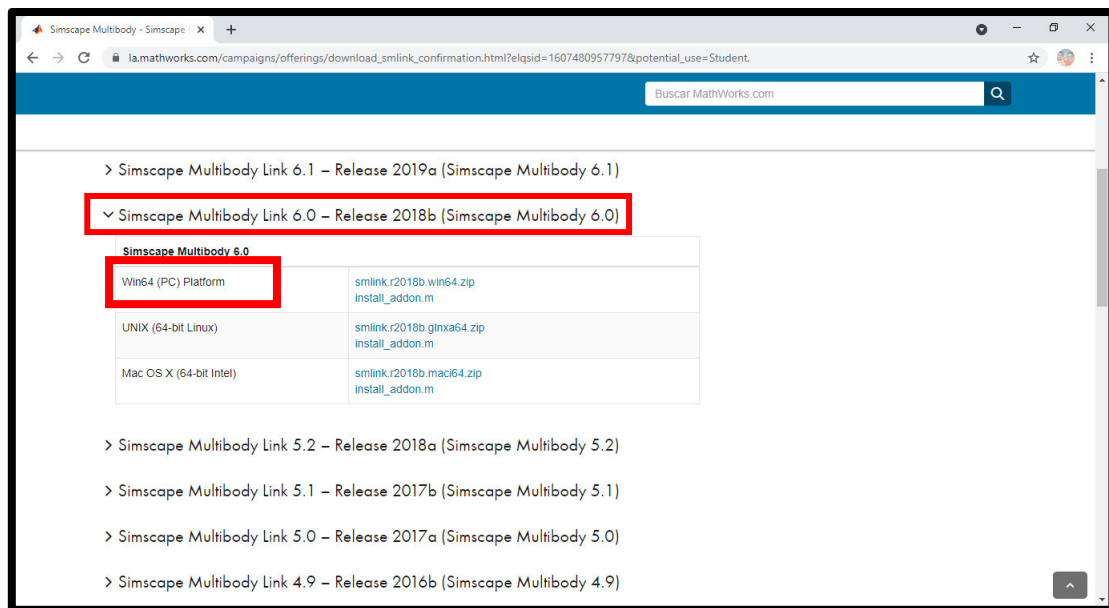


Figura 3.22 Librería Simscape Multibody Link y sus componentes [2]

A continuación, se descargó “smlink.2018b.win64.zip” y “install_addon.m” (Figura 3.23).

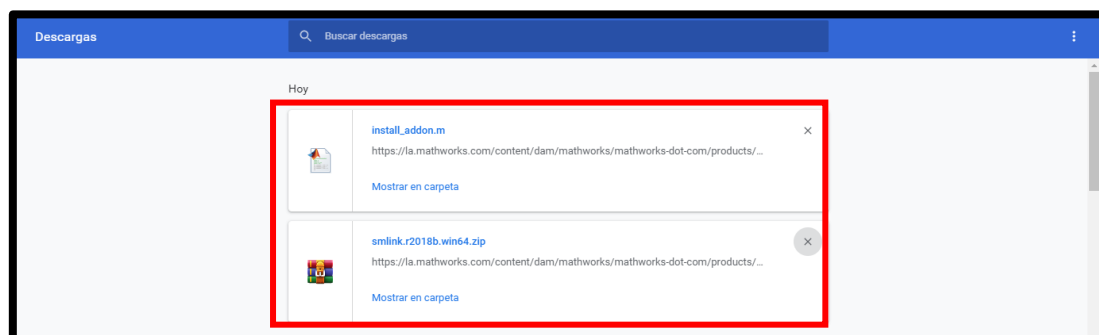


Figura 3.23 Descarga de smlink.2018b.win64.zip y install_addon.m

Vinculación entre Matlab y Autodesk Inventor.

Se inició Matlab como “Administrador” (Figura 3.24).

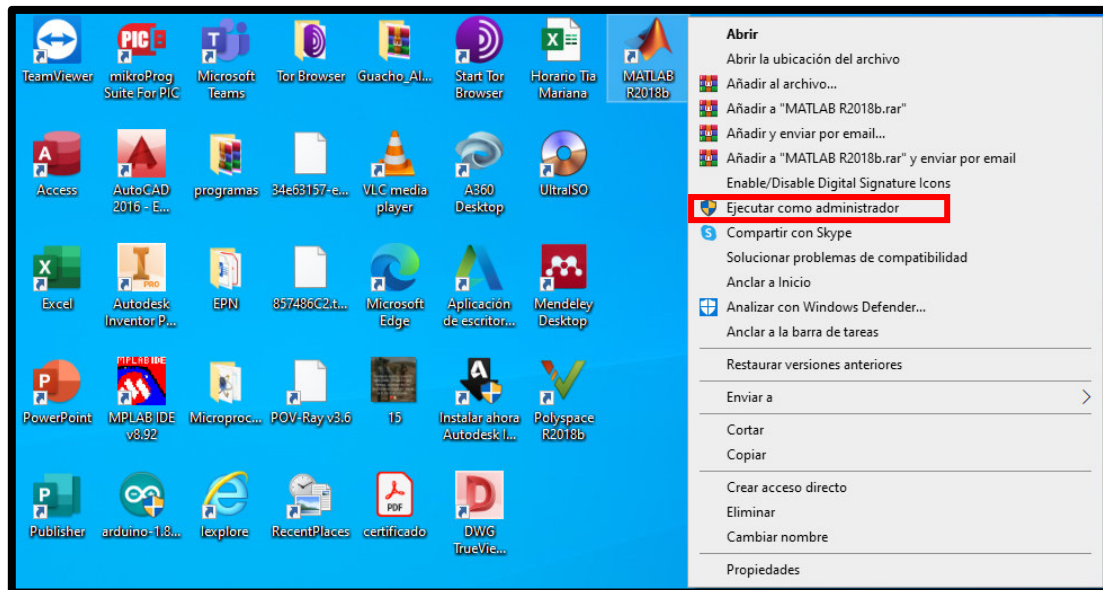


Figura 3.24 Ejecución de Matlab como administrador

En la ventana “Command Window” de Matlab se ejecutó el comando “`addpath('foldername')`”, en la parte “`foldername`” se escribió la ubicación de los archivos “`smlink.2018b.win64.zip`” y “`install_addon.m`” (Figura 3.25).

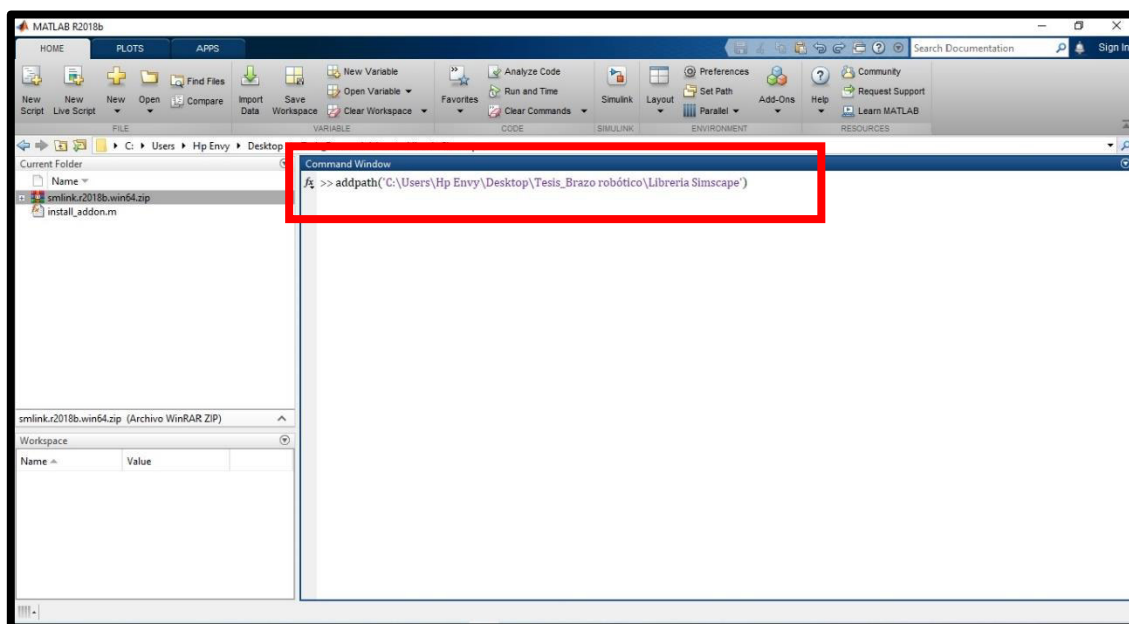


Figura 3.25 Ejecución del comando `addpath('foldername')`

Luego se ejecutó el comando “`install_addon('zipname')`”, en la parte de “`zipname`” se escribió “`smlink.2018b.win64.zip`” (Figura 3.26).

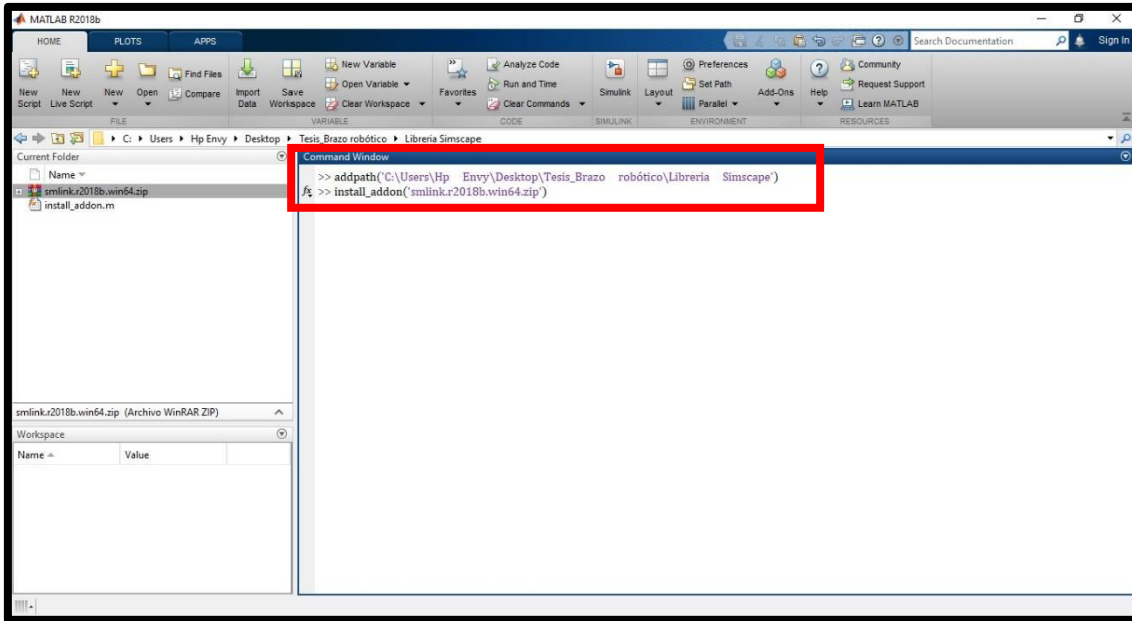


Figura 3.26 Ejecución del comando `install_addon('zipname')`

Como resultado de la ejecución del comando “`install_addon('zipname')`” se instala la librería Simscape Multibody Link (Figura 3.27).

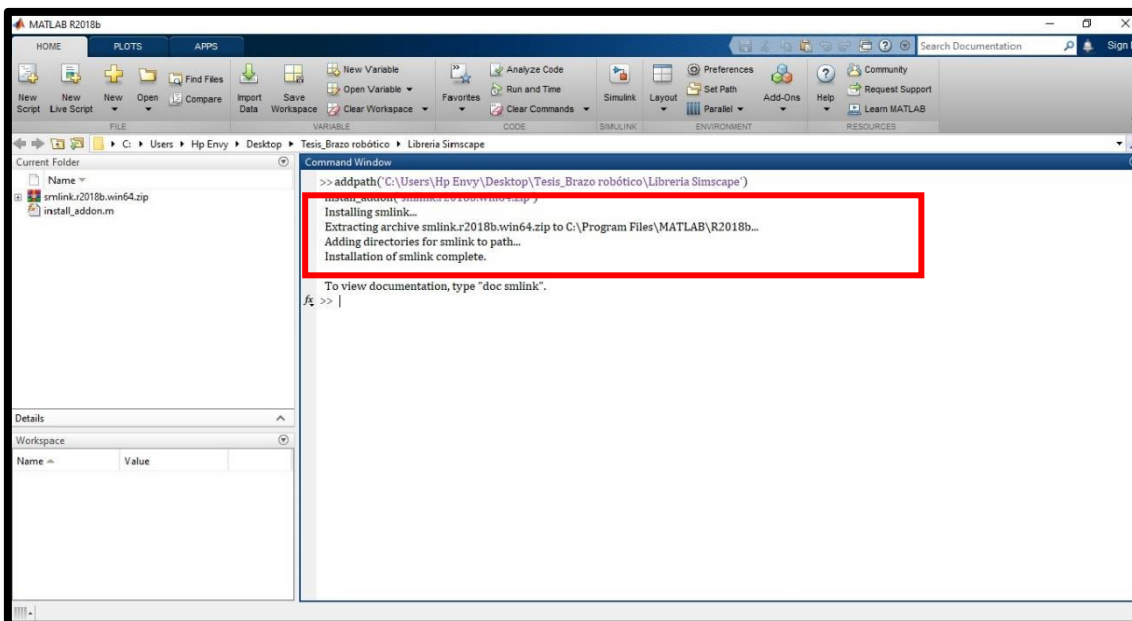


Figura 3.27 Instalación de la librería Simscape Multibody Link

Para vincular Matlab y Autodesk Inventor, en la ventana Command Window se ejecutó el comando “`smlink_linkinv`” (Figura 3.28).

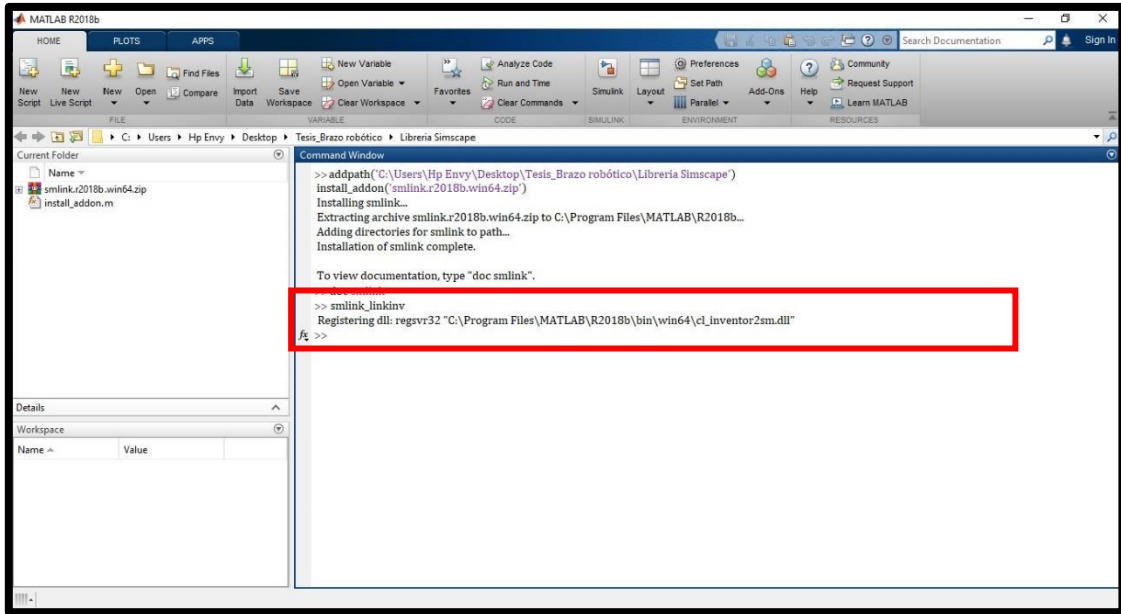


Figura 3.28 Ejecución del comando `smlink_linkinv`

Exportación e importación del ensamblaje del brazo robótico.

Para la exportación del brazo robótico desde Autodesk Inventor, en la pestaña “Complementos” se escoge la opción “Export Simscape Multibody” (Figura 3.29).

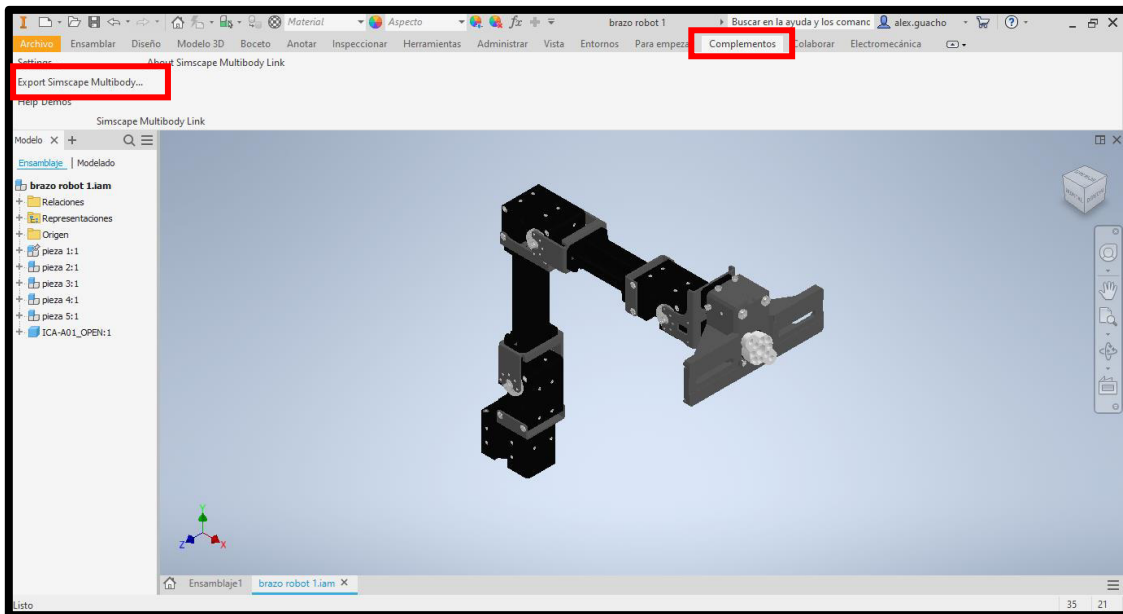


Figura 3.29 Pestaña complementos

Para la importación del brazo robótico hacia Simulink, en la ventana Command Window de Matlab se ejecutó el comando “`smimport('')`”; dentro del paréntesis se escribió el nombre con el que se guardó la exportación del brazo robótico (Figura 3.30).

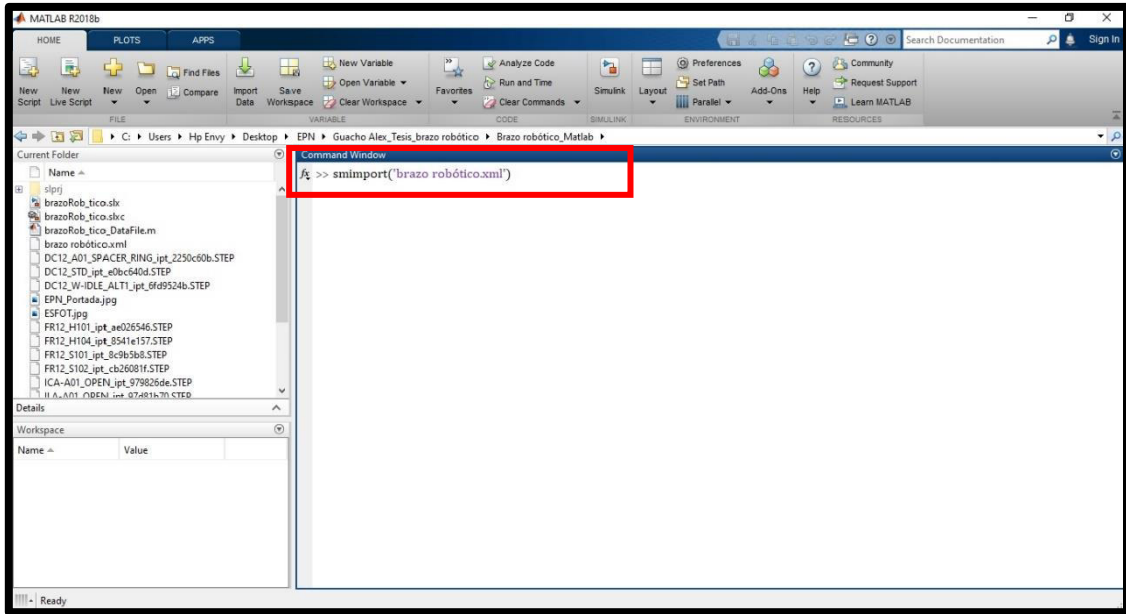


Figura 3.30 Ejecución del comando smimport

Como resultado de la importación, se despliega el diagrama de bloques del brazo robótico en Simulink (Figura 3.31). Además, haciendo clic en el botón “play” de la barra de herramientas de Simulink aparece la ventana “Mechanics Explorers” que muestra la simulación del brazo robótico en 3D (Figura 3.32).

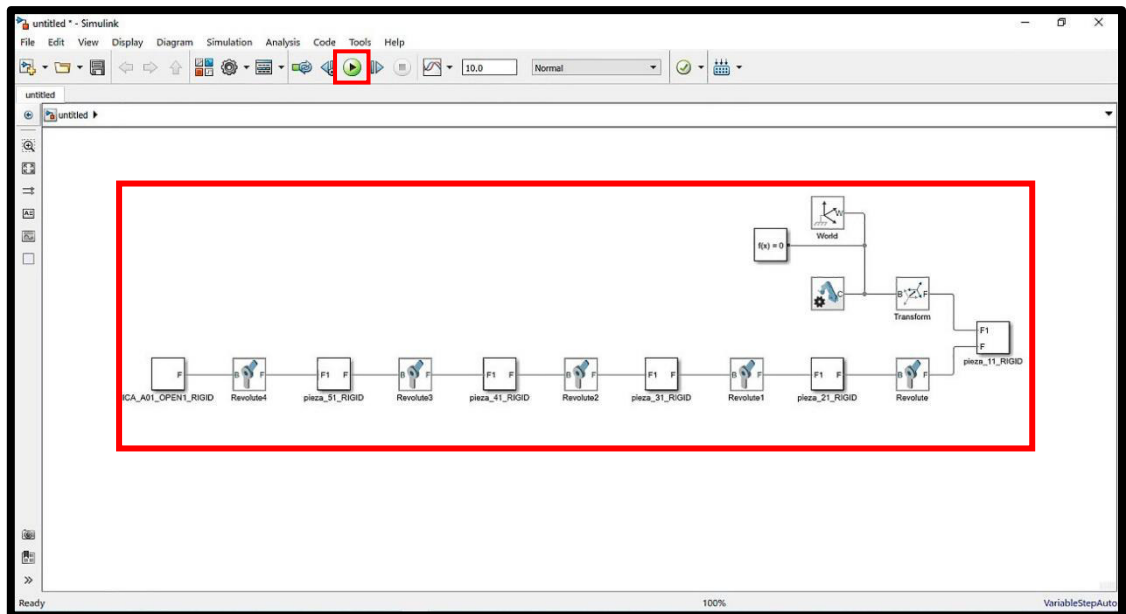


Figura 3.31 Diagrama de bloques obtenido de la importación

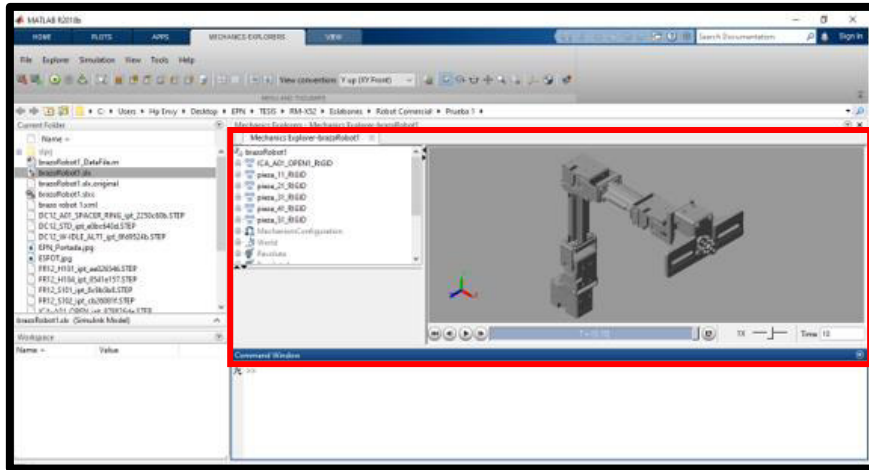


Figura 3.32 Brazo robótico en 3D

3.3 Ecuaciones de cinemática directa que describen el comportamiento del brazo robótico.

Para obtener las ecuaciones de cinemática directa se empleó la metodología de Denavit Hartenberg, descrita en la sección 1.3.

Análisis de Cinemática Directa.

Se estableció el sistema de coordenadas a emplearse en este análisis, el cual se ubicó en el centro de la base del brazo robótico (Figura 3.33).

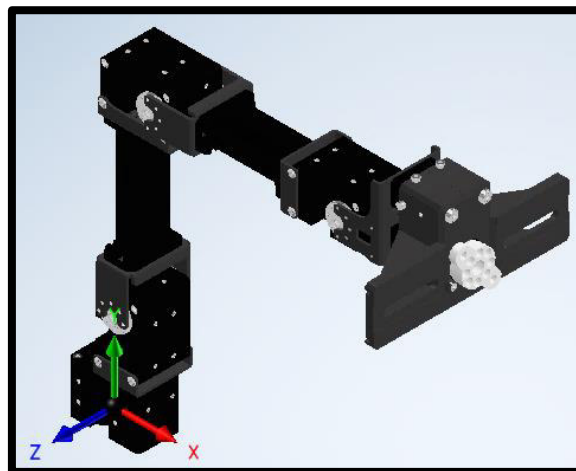


Figura 3.33 Sistema de coordenadas en la base del brazo robótico

Eslabón 0.

En la Figura 3.34 se muestra que el sistema de coordenadas ha girado 270 grados ($^{\circ}$) alrededor del eje x con el objetivo de determinar lo siguiente:

q_1 : rotación alrededor del eje z.

d_1 : 5.95 (cm) distancia a lo largo del eje z medida desde el centro del eslabón 0 hasta la articulación 1 (Figura 3.35).

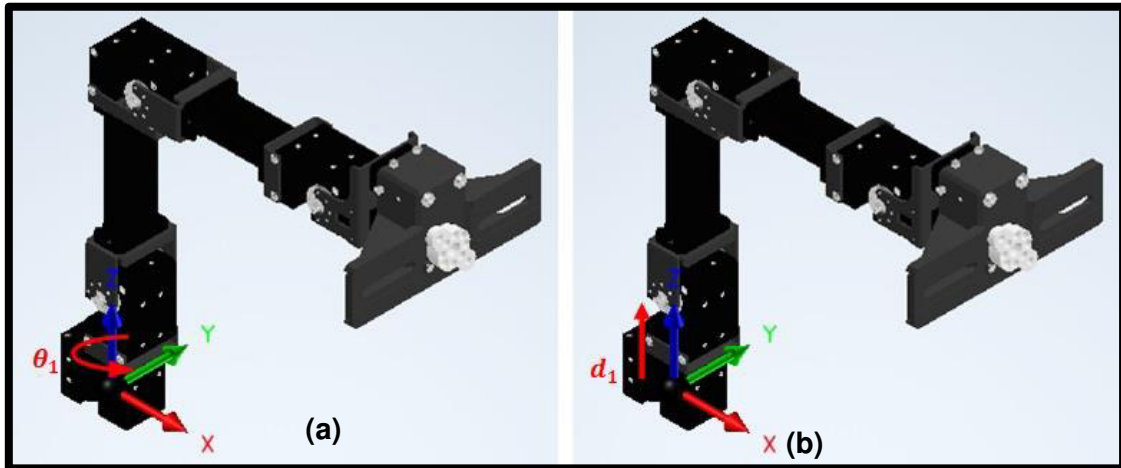


Figura 3.34 Eslabón 0 (a) Rotación alrededor del eje z (b) Desplazamiento d_1

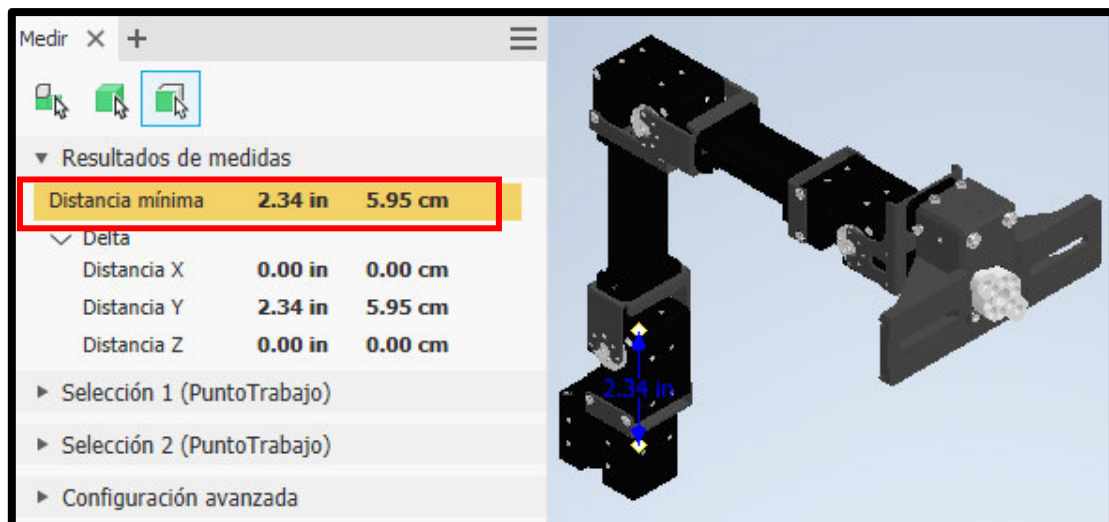


Figura 3.35 Valor de la distancia d_1

Eslabón 1

En la Figura 3.36 se muestra que el sistema de coordenadas se ha desplazado a la articulación 2; además, ha girado 90 grados ($^{\circ}$) alrededor del eje x con el objetivo de determinar lo siguiente:

q_2 : rotación alrededor del eje z.

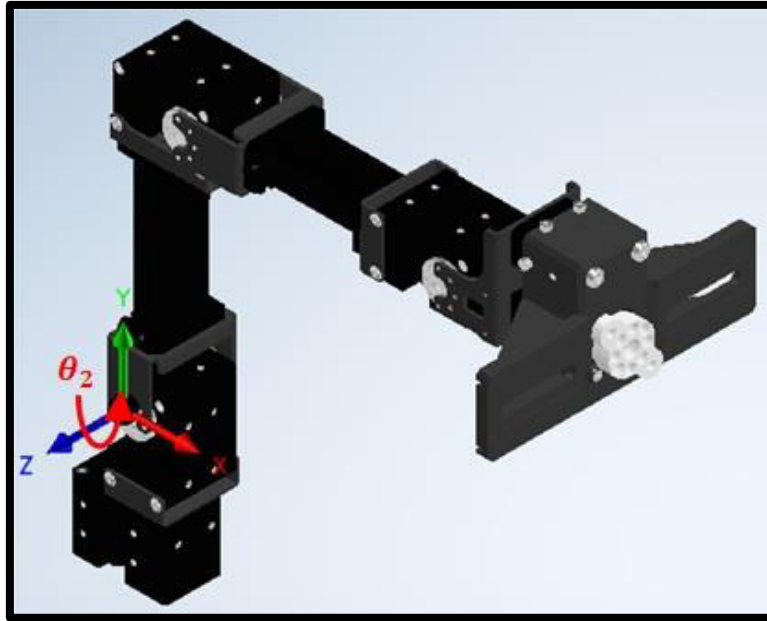


Figura 3.36 Rotación alrededor del eje z

En la Figura 3.37 se muestra que el sistema de coordenadas ha girado 270 grados ($^{\circ}$) alrededor del eje x con el objetivo de determinar lo siguiente:

d_2 : 12.80 (cm) distancia a lo largo del eje z medida desde la articulación 1 hasta el centro del servomotor del eslabón 2.

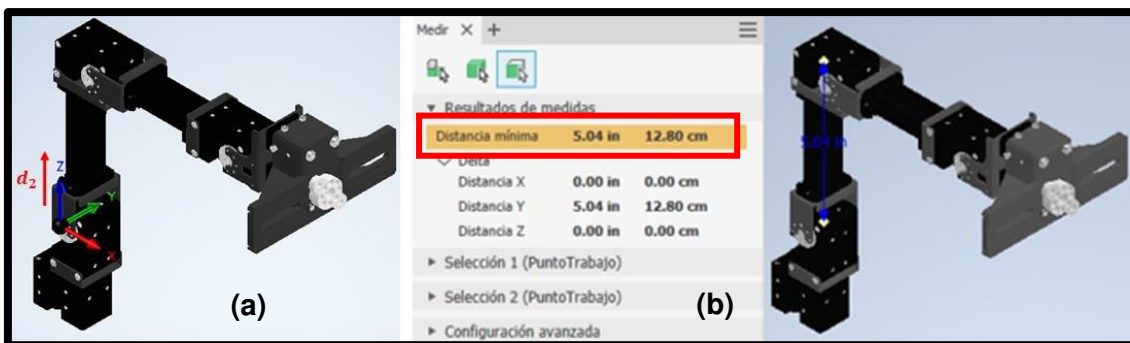


Figura 3.37 (a) Desplazamiento d_2 (b) Valor de la distancia d_2

Eslabón 2.

En la Figura 3.38 se muestra que el sistema de coordenadas se ha desplazado al centro del servomotor del eslabón 2; además, ha girado 90 grados ($^{\circ}$) alrededor del eje x con el objetivo de determinar lo siguiente:

a_1 : 2.40 (cm) distancia a lo largo del eje x medida desde el centro del servomotor del eslabón 2 hasta la articulación 3.

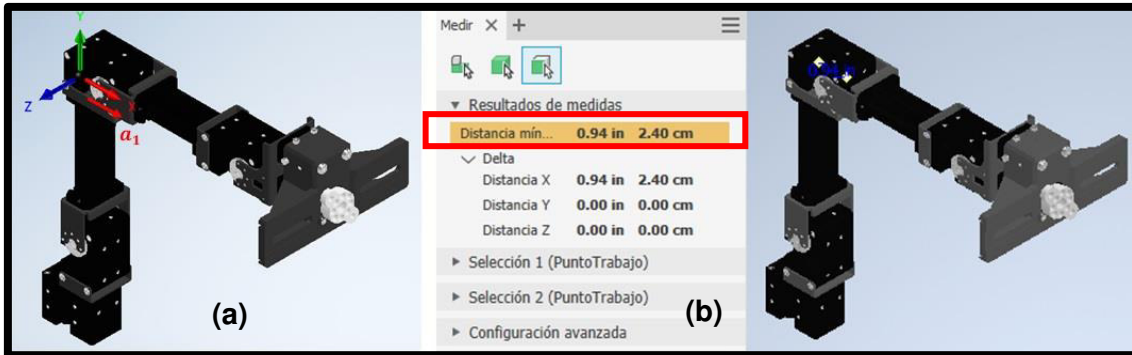


Figura 3.38 (a) Desplazamiento a_1 (b) Valor de la distancia a_1

En la Figura 3.39 se muestra que el sistema de coordenadas se ha desplazado a la articulación 3 con el objetivo de determinar lo siguiente:

- q_3 : rotación alrededor del eje z .
- a_2 : 12.40 (cm) distancia a lo largo del eje x medida desde la articulación 3 hasta la articulación 4.

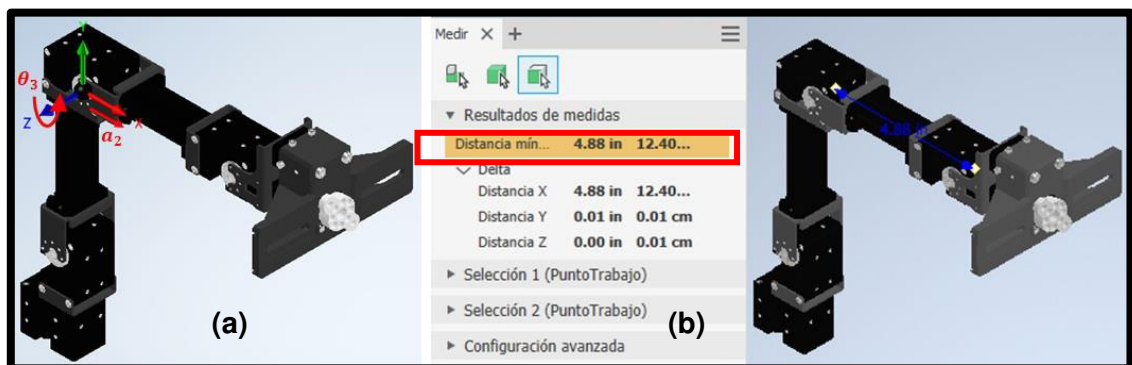


Figura 3.39 (a) Rotación q_3 y desplazamiento a_2 (b) Valor de la distancia a_2

Eslabón 3.

En la Figura 3.40 se muestra que el sistema de coordenadas se ha desplazado a la articulación 4 con el objetivo de determinar lo siguiente:

- q_4 : rotación alrededor del eje z .
- a_3 : 6.75 (cm) distancia a lo largo del eje x medida desde la articulación 4 hasta el efector final del brazo robótico.

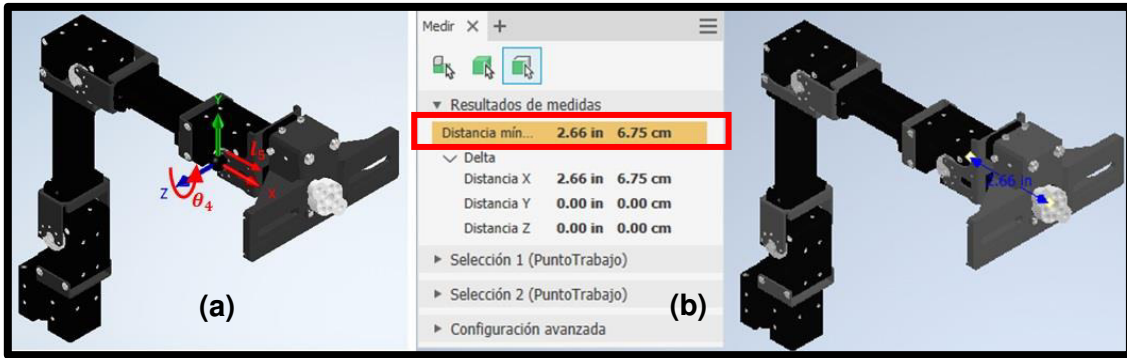


Figura 3.40 (a) Rotación q_3 y desplazamiento a_3 (b) Valor de la distancia a_3

En la Figura 3.41 se muestran todos los valores determinados anteriormente.

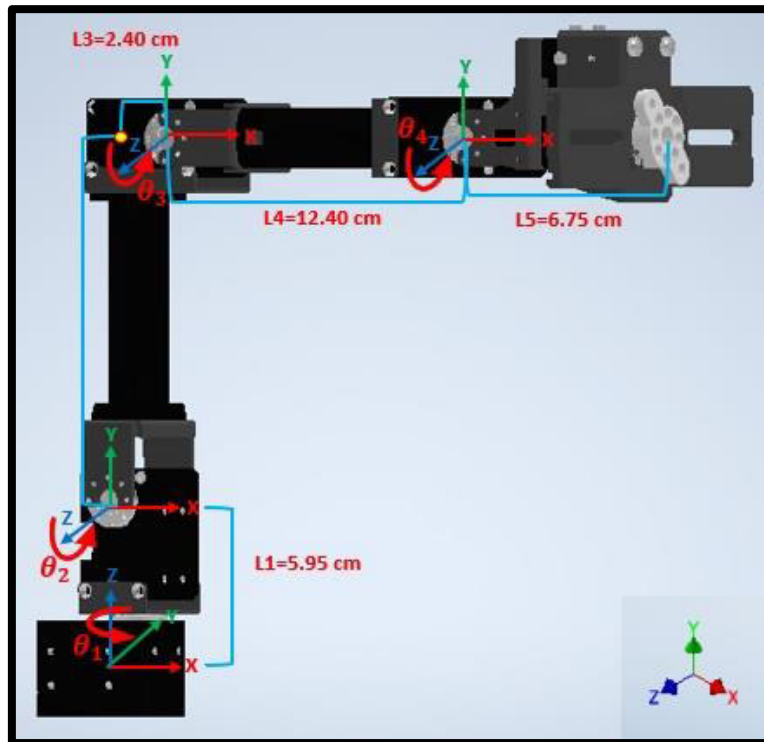


Figura 3.41 Sistema de coordenadas para cada articulación y distancias medidas

En la Tabla 3.4 se muestran los parámetros de Denavit Hartenberg, descritos en la sección 1.3, que se obtuvieron al realizar el análisis anterior.

Tabla 3.4 Parámetros de Denavit-Hartenberg

Parámetros Pasos	a_i (cm)	α_i (°)	d_i (cm)	θ_i (°)
1	-	270	-	-
2	-	-	5.95	q_1

Parámetros Pasos	a_i (cm)	α_i (°)	d_i (cm)	θ_i (°)
3	-	90	-	-
4	-	-	-	q_2
5	-	270	-	-
6	-	-	12.80	-
7	-	90	-	-
8	2.40	-	-	-
9	12.40	-	-	q_3
10	6.75	-	-	q_4

Con base en la Tabla 3.4 y las matrices de rotación y traslación descritas en la sección 1.3, se desarrolla la cinemática.

Usando la Ecuación 1.5 se obtiene:

$$Rot(\alpha_1) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\alpha_1) & -\sin(\alpha_1) & 0 \\ 0 & \sin(\alpha_1) & \cos(\alpha_1) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Ecuación 3.1 Matriz de rotación α_1

Donde:

α_1 : 270 (°) grados.

Usando la Ecuación 1.2 se obtiene:

$$Rot(d_1, \theta_2) = \begin{bmatrix} \cos(\theta_2) & -\sin(\theta_2) & 0 & 0 \\ \sin(\theta_2) & \cos(\theta_2) & 0 & 0 \\ 0 & 0 & 1 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Ecuación 3.2 Matriz de rotación θ_2 y traslación d_1

Donde:

θ_2 : q_1 (°) grados.

d_1 : 5.95 (cm) centímetros.

Usando la Ecuación 1.5 se obtiene:

$$Rot(\alpha_3) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\alpha_3) & -\sin(\alpha_3) & 0 \\ 0 & \sin(\alpha_3) & \cos(\alpha_3) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Ecuación 3.3 Matriz de rotación α_3

Donde:

α_3 : 90 (°) grados.

Usando la Ecuación 1.2 se obtiene:

$$Rot(\theta_4) = \begin{bmatrix} \cos(\theta_4) & -\sin(\theta_4) & 0 & 0 \\ \sin(\theta_4) & \cos(\theta_4) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Ecuación 3.4 Matriz de rotación θ_4

Donde:

θ_4 : q_2 (°) grados.

Usando la Ecuación 1.5 se obtiene:

$$Rot(\alpha_5) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\alpha_5) & -\sin(\alpha_5) & 0 \\ 0 & \sin(\alpha_5) & \cos(\alpha_5) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Ecuación 3.5 Matriz de rotación α_5

Donde:

α_5 : 270 (°) grados.

Usando la Ecuación 1.2 se obtiene:

$$Rot(d_2, \theta_6) = \begin{bmatrix} \cos(\theta_6) & -\sin(\theta_6) & 0 & 0 \\ \sin(\theta_6) & \cos(\theta_6) & 0 & 0 \\ 0 & 0 & 1 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Ecuación 3.6 Matriz de rotación θ_6 y traslación d_2

Donde:

θ_6 : 0 (°) grados.

d_2 : 12.80 (cm) centímetros.

Usando la Ecuación 1.5 se obtiene:

$$Rot(\alpha_7) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\alpha_7) & -\sin(\alpha_7) & 0 \\ 0 & \sin(\alpha_7) & \cos(\alpha_7) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Ecuación 3.7 Matriz de rotación α_7

Donde:

α_7 : 90 (°) grados.

Usando la Ecuación 1.2 y la Ecuación 1.4 se obtiene:

$$Trans(\theta_8, a_1) = \begin{bmatrix} \cos(\theta_8) & -\sin(\theta_8) & 0 & a_1 * \cos(\theta_8) \\ \sin(\theta_8) & \cos(\theta_8) & 0 & a_1 * \sin(\theta_8) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Ecuación 3.8 Matriz de traslación a_1

Donde:

θ_8 : 0 (°) grados.

a_1 : 2.40 (cm) centímetros.

Usando la Ecuación 1.2 y la Ecuación 1.4 se obtiene:

$$Rot, Trans(\theta_9, a_2) = \begin{bmatrix} \cos(\theta_9) & -\sin(\theta_9) & 0 & a_2 * \cos(\theta_9) \\ \sin(\theta_9) & \cos(\theta_9) & 0 & a_2 * \sin(\theta_9) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Ecuación 3.9 Matriz de rotación θ_9 y traslación a_2

Donde:

θ_9 : q_3 (°) grados.

a_2 : 12.40 (cm) centímetros.

Usando la Ecuación 1.2 y la Ecuación 1.4 se obtiene:

$$Rot, Trans(\theta_{10}, a_3) = \begin{bmatrix} \cos(\theta_{10}) & -\text{sen}(\theta_{10}) & 0 & a_3 * \cos(\theta_{10}) \\ \text{sen}(\theta_{10}) & \cos(\theta_{10}) & 0 & a_3 * \text{sen}(\theta_{10}) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Ecuación 3.10 Matriz de rotación θ_{10} y traslación a_3

Donde:

θ_{10} : q_4 (°) grados.

a_3 : 6.75 (cm) centímetros.

Simulink

Con base en la importación del brazo robótico a Simulink descrita en la sección 3.2, se utilizó el panel de bibliotecas para encontrar los bloques necesarios para la simulación del brazo robótico, haciendo clic sobre el botón “Library Browser” aparece el panel que muestra de manera detallada y clasificada las bibliotecas que contiene Simulink (Figura 3.42).

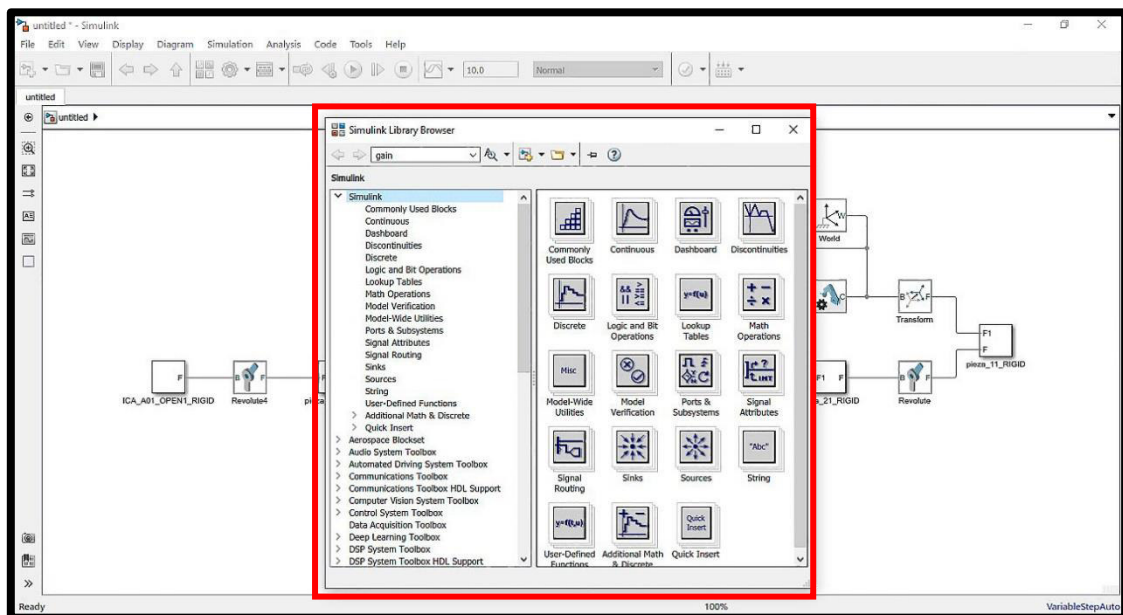


Figura 3.42 Panel de bibliotecas de Simulink

A continuación, se describen los bloques que se utilizaron para el desarrollo de este proyecto.

MATLAB Function.

El bloque MATLAB Function (Figura 3.43) crea funciones que contienen uno o más comandos secuenciales para escribir un programa con varias líneas de código. Además, permite declarar el nombre de la función, las entradas y las salidas [13].

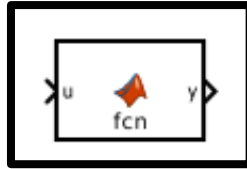


Figura 3.43 Bloque Matlab Function

From Workspace

El bloque From Workspace (Figura 3.44) lee los valores de datos especificados en series de tiempo, matriz o formato de estructura desde el espacio de trabajo de MATLAB [14].

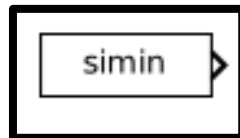


Figura 3.44 Bloque From Workspace

La configuración del bloque requiere los siguientes parámetros:

- Data (Nombre de la variable).
- Sample Time (Tiempo de muestra).

En la Tabla 3.5 se muestra la configuración de parámetros, que se realizó para este bloque.

Tabla 3.5 Parámetros de bloque From Workspace

Parámetro	Valor utilizado	Unidades	Justificación de los valores utilizados
Nombre de la variable	t q#_s	–	Nombre debido a que es la articulación del brazo robótico.
Tiempo de muestreo	0.01	ms	Valor debido a que es el tiempo de simulación usado en el entorno Simulink.

En la Figura 3.45 se muestran los parámetros configurados.

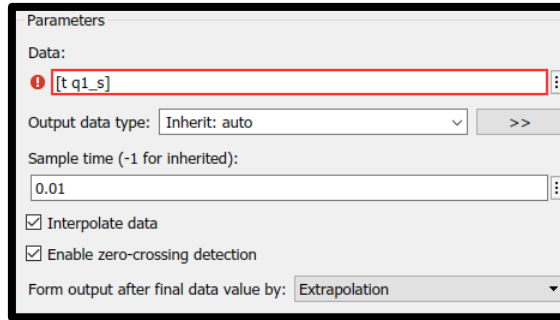


Figura 3.45 Ventana de parámetros del Bloque From Workspace

To Workspace

El bloque To Workspace (Figura 3.46) escribe datos de señales de entrada en un espacio de trabajo. Durante la simulación, el bloque escribe datos en un búfer interno [15].

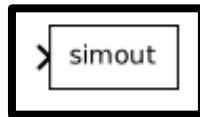


Figura 3.46 Bloque To Workspace

La configuración del bloque requiere los siguientes parámetros:

- Variable name (Nombre de la variable).
- Save format (Guardar formato).
- Sample time (Tiempo de muestra).

En la Tabla 3.6 se muestra la configuración de parámetros, que se realizó para este bloque.

Tabla 3.6 Parámetros del bloque To Workspace

Parámetro	Valor utilizado	Unidades	Justificación de los valores utilizados
Nombre de la variable	posición_cal	–	Nombre asignado debido a posición del efector final del brazo robótico.
Guardar formato	Array	–	Permite guardar señales en 3D.
Tiempo de muestreo	0.01	ms	Valor debido a que es el tiempo de simulación usado en el entorno Simulink.

En la Figura 3.47 se muestran los parámetros configurados.

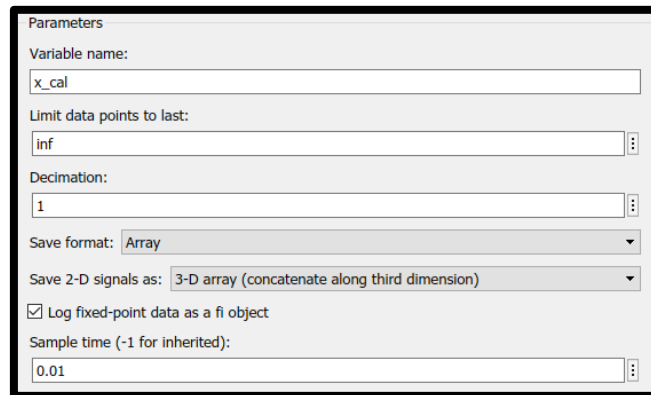


Figura 3.47 Ventana de parámetros del Bloque To Workspace

Display

El bloque Display (Figura 3.48) muestra el valor de la señal durante la simulación [16].



Figura 3.48 Bloque Display

Simulink-PS Converter

El bloque Simulink-PS Converter (Figura 3.49) convierte una señal de entrada de Simulink en una señal física [17].

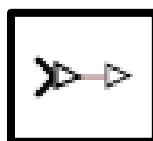


Figura 3.49 Bloque Simulink-PS Converter

La configuración del bloque requiere los siguientes parámetros:

- Filtering and derivatives (Filtrado y derivados).
- Provided signals (Señales proporcionales).
- Input filtering time constant (in seconds) (Constante de tiempo).

En la Tabla 3.7 se muestra la configuración de parámetros, que se realizó para este bloque.

Tabla 3.7 Parámetros del bloque Simulink-Ps Converter

Parámetro	Valor utilizado	Unidades	Justificación de los valores utilizados
Filtrado y derivados	Filter input, derivatives calculated	–	Permite filtrar y derivar valores calculados.
Señales proporcionales	Second-order filtering	–	Permite filtrar señales de segundo orden.
Constante de tiempo	0.01	ms	Valor debido a que es el tiempo de simulación usado en el entorno Simulink.

En la Figura 3.50 se muestran los parámetros configurados.

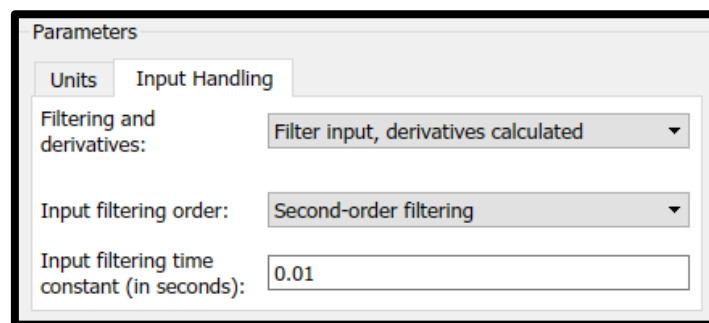


Figura 3.50 Ventana de parámetros del Bloque Simulink-Ps Converter

PS-Simulink Converter

El bloque PS-Simulink Converter (Figura 3.51) convierte una señal física en una señal de salida de Simulink [18].

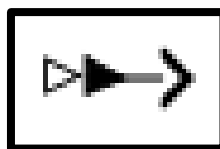


Figura 3.51 Bloque PS-Simulink Converter

La configuración del bloque requiere los siguientes parámetros:

- Output signal unit (unidad de señal de salida).

En la Tabla 3.8 se muestra la configuración de parámetros, que se realizó para este bloque.

Tabla 3.8 Parámetros del bloque Ps-Simulink Converter

Parámetro	Valor utilizado	Unidades	Justificación de los valores utilizados
Unidad de señal de salida	-	m	Valor proporcional para conocer la posición del efector final.

En la Figura 3.52 se muestran los parámetros configurados.

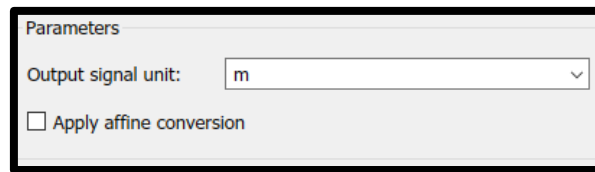


Figura 3.52 Ventana de parámetros del bloque PS-Simulink Converter

Transform Sensor

El bloque Transform Sensor (Figura 3.53) mide la relación espacial relativa entre dos fotogramas arbitrarios en un modelo [19].



Figura 3.53 Bloque Transform Sensor

La configuración del bloque requiere los siguientes parámetros:

- Translation (traslado).

En la Tabla 3.9 se muestra la configuración de parámetros, que se realizó para este bloque.

Tabla 3.9 Parámetros del bloque Transform Sensor

Parámetro	Valor utilizado	Unidades	Justificación de los valores utilizados
Traslado	x, y, z	-	Permite conocer la posición por medio del sistema de coordenadas.

En la Figura 3.54 se muestran los parámetros configurados.

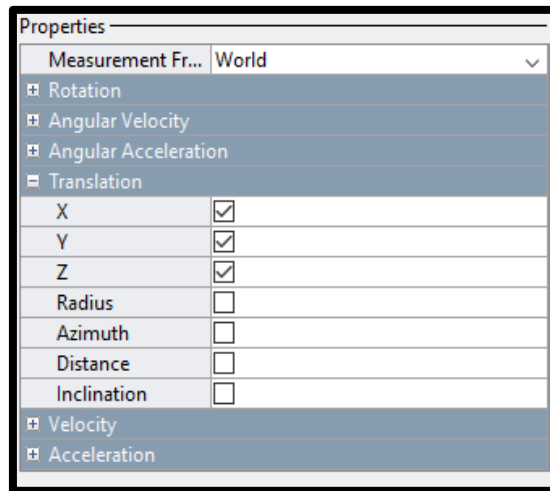


Figura 3.54 Ventana de parámetros del bloque Transform Sensor

Gain

El bloque Gain (Figura 3.55) multiplica la entrada por un valor constante (ganancia). La entrada y la ganancia pueden ser cada una un escalar, un vector o una matriz.

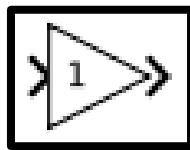


Figura 3.55 Bloque Gain

La configuración del bloque requiere el valor constante de ganancia (Figura 3.56).

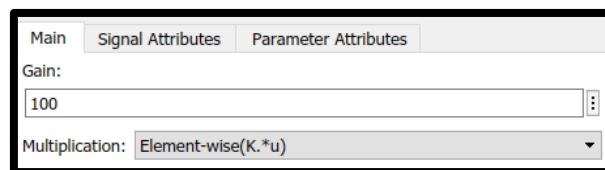


Figura 3.56 Ventana de parámetros del bloque Gain

Simulación del análisis de cinemática directa en el diagrama de bloques del brazo robótico.

Para resolver la cinemática directa del brazo robótico en Simulink se utilizó el bloque Matlab Function, para lo que se asignaron las entradas, las salidas y el nombre de la función (Figura 3.57).

Donde:

- q_n : Entradas - Articulaciones del brazo robótico (°).
- X, Y, Z : Salidas (cm).
- CD4GDL : Nombre de la función (Cinemática directa 4 grados de libertad).

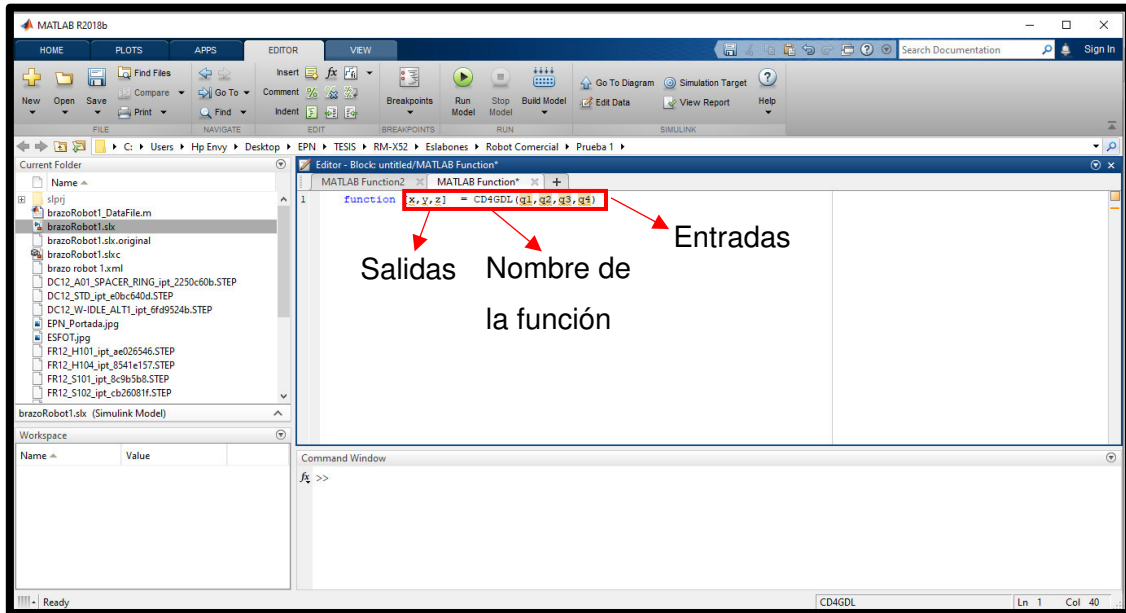


Figura 3.57 Asignación de las entradas, las salidas y el nombre de la función

Luego se escribieron los parámetros de la Tabla 3.4 (d_i, a_i, α_i) obtenidos en la sección 3.3 (Figura 3.58).

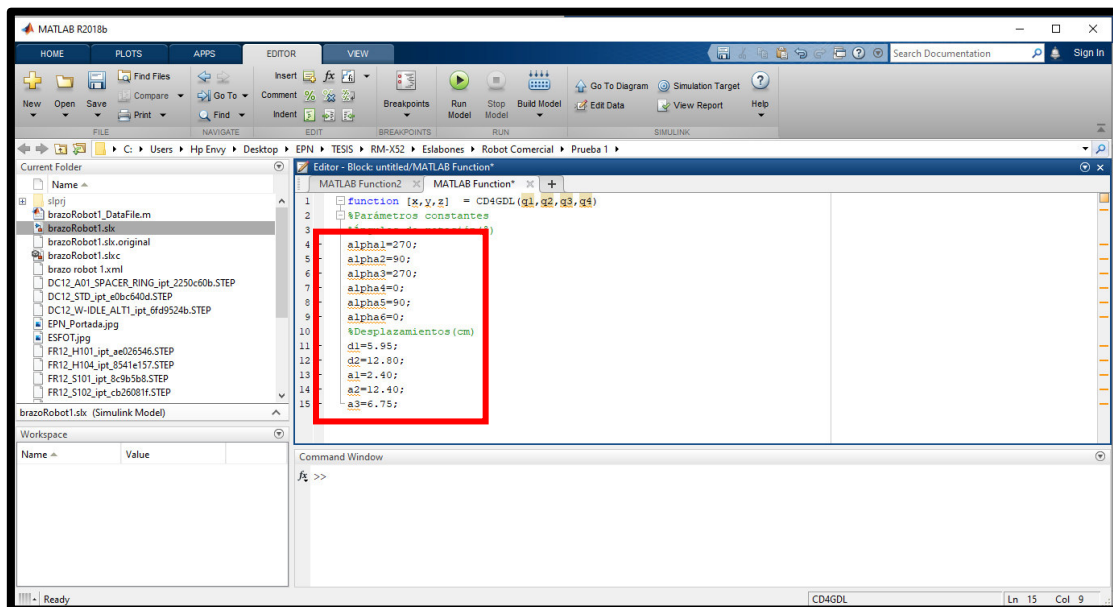


Figura 3.58 Parámetros de Denavit Hartenberg d_i, a_i, α_i

A continuación, se escribieron las matrices de transformación obtenidas en la sección 3.3. En la Figura 3.59 se muestra una de las matrices de transformación.

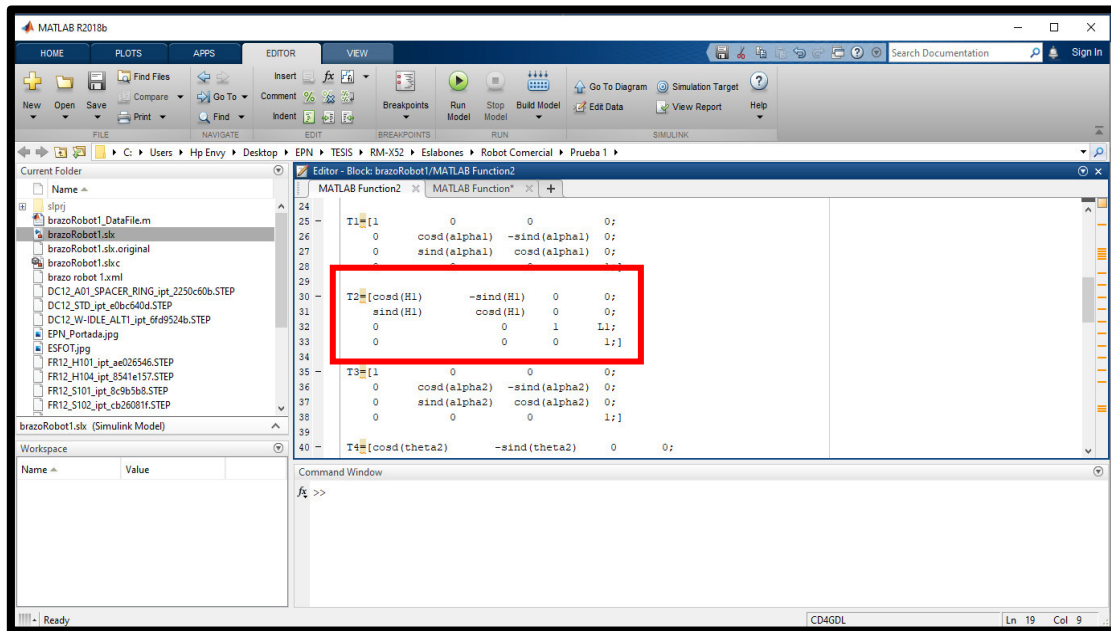


Figura 3.59 Matriz de transformación en Matlab

Luego se realizó una multiplicación de todas las matrices de transformación, para obtener la matriz resultante, de la cual se extrae los valores de x, y, z que indican la posición del efector final del brazo robótico (Figura 3.60).

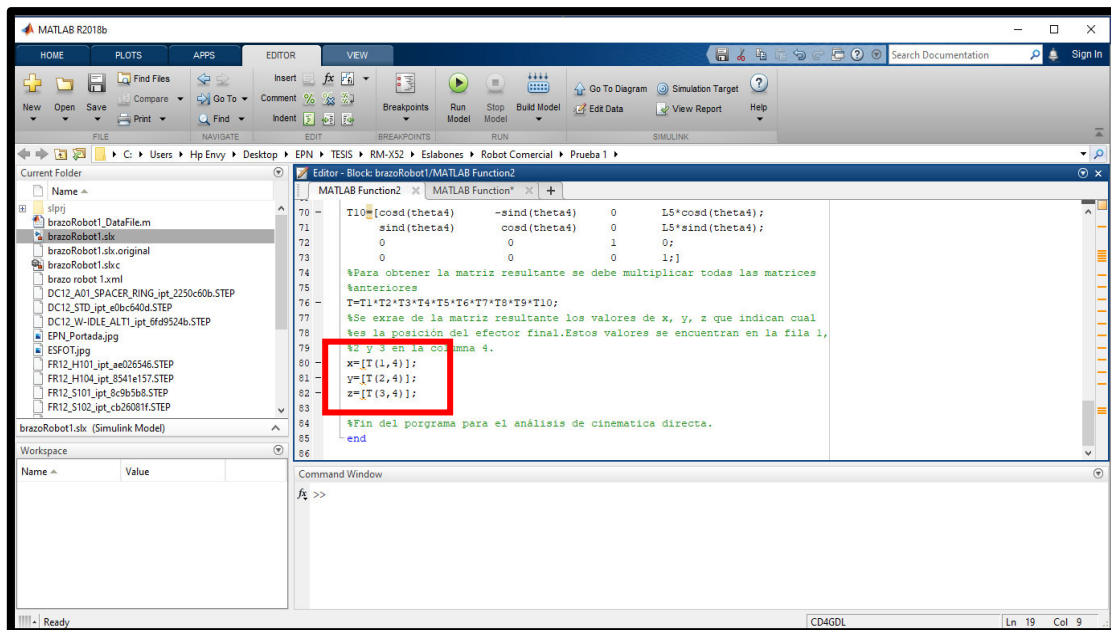


Figura 3.60 Posición del efector final x, y, z

Una vez que se añadieron y configuraron los bloques: Matlab Function, From Workspace, To Workspace, Simulink-Ps Converter, Ps-Simulink Converter, Transform Sensor; se añadieron los bloques que no requieren configuración y son los siguientes: Display, Gain.

Finalmente, en el Anexo 2 se muestran las conexiones entre los bloques que permiten simular el brazo robótico de 4 GDL en el entorno de Simulink.

En el Anexo 3 se muestra el código de programación para resolver la cinemática directa del brazo robótico.

Interfaz del brazo robótico.

Se creó la interfaz con la finalidad de facilitar al usuario el manejo de la simulación del brazo robótico, como es el ingresar los valores de giro para cada articulación, simular el movimiento del brazo robótico y conocer la posición del efector final. Se utilizó la norma ANSI/ISA-S5.5-1985 para crear la interfaz, además se realizó un diagrama de flujo (Anexo 2) que explica su funcionamiento.

Se abrió un nuevo archivo de Matlab, luego en la ventana Command Window se ejecutó el comando “guide” (Figura 3.61).

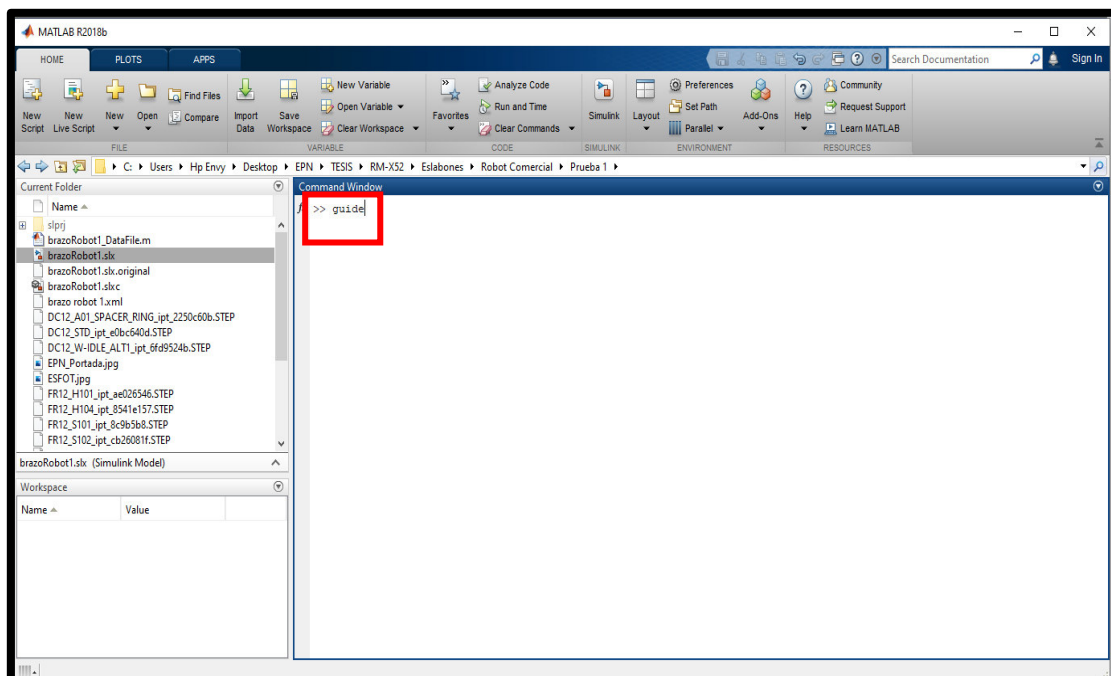


Figura 3.61 Ejecución del comando “guide”

Como resultado de la ejecución del comando “guide” se despliega la ventana GUIDE Quick Start, luego se seleccionó la opción Blank GUI (Default) (Figura 3.62).

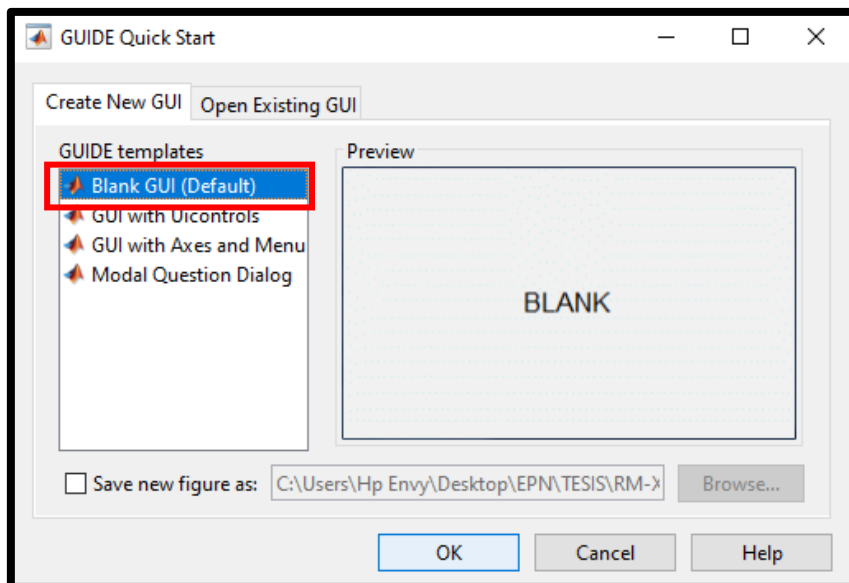


Figura 3.62 Ventana GUIDE Quick Start

Aparece la ventana de trabajo Guide, en el área de diseño se insertaron los componentes: Push Button, Edit Text, Static Text y Axes.

En la Tabla 3.10 se muestra una descripción de los componentes utilizados.

Tabla 3.10 Componentes para crear la interfaz

Control	Valor de estilo	Descripción
Push Button	'pushbutton'	Invoca un evento inmediatamente.
Edit Text	'edit'	Caja para editar texto.
Static Text	'text'	Muestra el texto en una caja.
Axes	'Axes'	Representa de manera gráfica funciones o imágenes.

En el Anexo 3 se muestra el código de programación para la interfaz. En la Figura 3.63 se muestra la interfaz creada para la simulación del brazo robótico.

Donde:

Valor calculado : Corresponde a la posición del efector final obtenida mediante la cinemática directa (cm).

Valor medido : Corresponde a la posición del efector final obtenida mediante el robot virtual (cm).

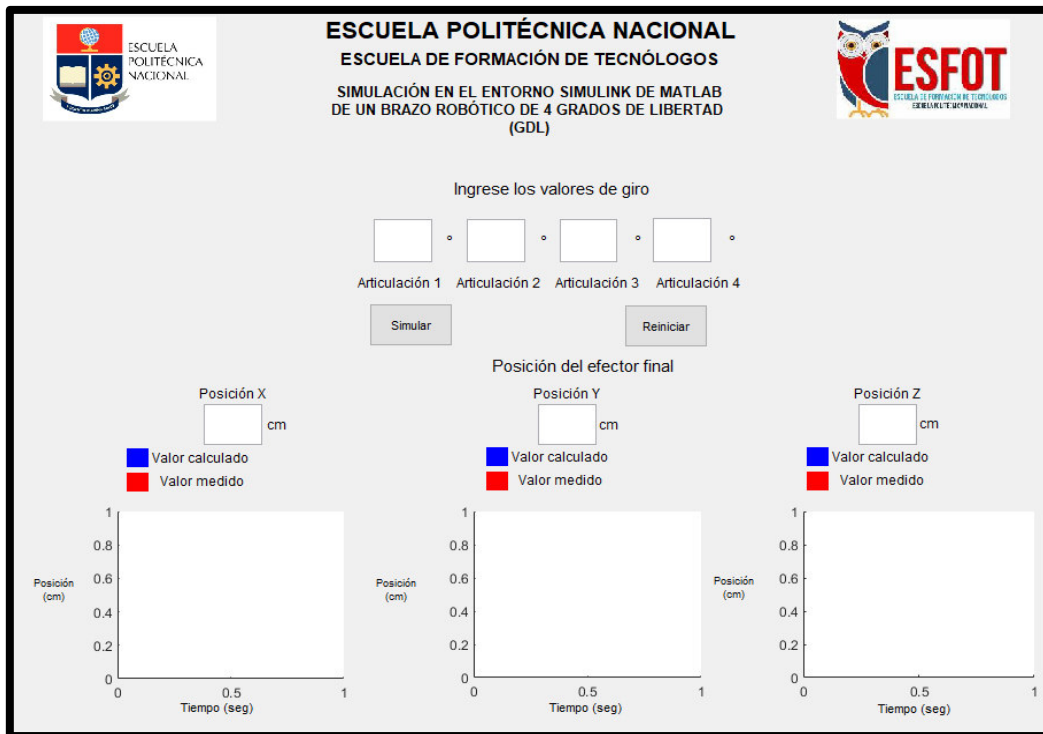


Figura 3.63 Interfaz del brazo robótico.

A continuación, en la Figura 3.64 se muestra un ejemplo de cómo ingresar los ángulos de giro y también cómo se muestran los valores y las gráficas que indican la posición del efector final del brazo robótico.

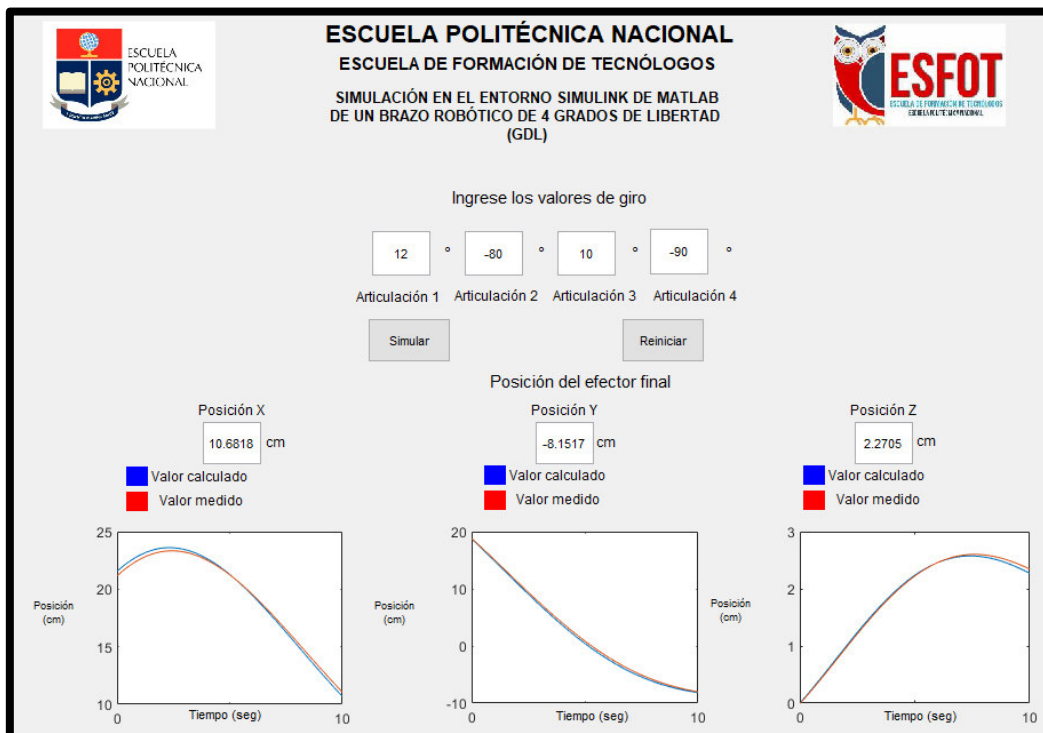


Figura 3.64 Uso de la interfaz del brazo robótico

3.4 Pruebas y análisis de resultados.

Para verificar el funcionamiento de la simulación del brazo robótico, se realizaron tres pruebas, se midió el error relativo porcentual existente de la posición del efector final entre el valor calculado y el valor medido.

Comparación entre Simulink y Autodesk Inventor.

Esta prueba consiste en obtener la posición del efector final del brazo robótico, asignando los mismos valores de giro, tanto en el análisis de cinemática directa en Simulink, como en el robot virtual ensamblado en Autodesk Inventor.

En la Tabla 3.11 se muestran los valores obtenidos en esta prueba.

Tabla 3.11 Prueba entre Autodesk Inventor y Simulink de Matlab

Parámetros		N° de prueba			
		1	2	3	4
Ángulo de giro para las articulaciones (°)	1	33	-97	135	-10
	2	-67	45	230	-35
	3	90	-79	350	-96
	4	-12	66	670	-57
Posición del efector final calculada (cm)	X	25.80	-1.05	5.58	6.13
	Y	14.87	13.34	-10.91	-1.71
	Z	16.76	-8.58	-5.58	-1.08
Posición del efector final medida (cm)	X	25.79	-1.05	5.57	6.13
	Y	14.87	13.34	-10.91	-1.70
	Z	16.75	-8.58	-5.57	-1.08
Error relativo porcentual (%)	X	0.03	0.00	0.17	0.00
	Y	0.00	0.00	0.00	0.58
	Z	0.059	0.00	0.17	0.00

En base Tabla 3.11 se observa que el máximo error obtenido es de 0.585 (%), lo que significa que el efector final del robot virtual alcanza la posición calculada mediante las ecuaciones de cinemática.

En el Anexo 4 se encuentran las imágenes obtenidas al realizar esta prueba.

Comparación entre el análisis de cinemática directa y Simulink

Esta prueba consiste en obtener la posición del efector final del brazo robótico, asignando los mismos valores de giro, tanto en el análisis de cinemática directa en Simulink, como en el robot virtual exportado en Simulink.

En la Tabla 3.12 se muestran los valores obtenidos en esta prueba.

Tabla 3.12 Prueba en Simulink de Matlab

Parámetros		N° de prueba			
		1	2	3	4
Ángulo de giro para las articulaciones (°)	1	13.4	42	-77	-97
	2	-45	53	-63	-26
	3	73	91	55	-11
	4	-82	120	23	-8
Posición del efector final calculada (cm)	X	24.90	-14.50	7.04	-2.70
	Y	13.60	16.10	9.64	4.16
	Z	5.94	-13.00	-30.40	-22.20
Posición del efector final medida (cm)	X	24.70	-14.40	7.00	-2.67
	Y	13.90	16.50	9.56	4.43
	Z	5.89	-13.00	-30.10	-22.00
Error relativo porcentual (%)	X	0.80	0.10	0.50	2.20
	Y	-2.15	-2.30	0.83	-6.40
	Z	0.80	0.30	1.10	1.10

En base Tabla 3.12 se observa que el máximo error obtenido es de 1.1 (%), lo que significa que el efector final del robot virtual alcanza la posición calculada mediante las ecuaciones de cinemática.

En el Anexo 4 se encuentran las imágenes obtenidas al realizar esta prueba.

Comparación del ángulo de giro ingresado

Esta prueba consiste en comparar que el ángulo de giro ingresado en la interfaz del brazo robótico coincida con el medido en el robot virtual exportado a Simulink.

En la Tabla 3.13 se muestran los valores obtenidos en esta prueba.

Tabla 3.13 Prueba del ángulo de giro ingresado a cada articulación

Parámetros		N° de prueba			
		1	2	3	4
Valor de giro ingresado mediante la interfaz (°)	1	56	44	-69	27
	2	22	73	-14	50
	3	-33	12	-8	20
	4	-23	98	-11	30
Valor de giro medido (rad)	1	0.97	0.76	-1.20	0.47
	2	0.38	1.27	-0.20	0.87
	3	-0.57	0.20	-0.13	0.34
	4	-0.40	1.70	-0.10	0.52
Conversión de radianes a grados del valor medido (°)	1	55.90	43.90	-68.90	26.90
	2	21.90	72.90	13.80	49.90
	3	-32.90	11.90	-7.97	19.90
	4	-22.90	97.90	-10.90	29.90
Error relativo porcentual (%)	1	-0.10	0.00	0.10	0.00
	2	0.00	0.00	0.90	0.00
	3	0.10	0.10	0.40	0.10

Parámetros		N° de prueba			
		1	2	3	4
Error relativo porcentual (%)	4	0.10	0.00	0.50	0.00

En base a la Tabla 3.13 se observa que el máximo error obtenido es de 0.9 (%), lo que significa que las articulaciones del robot virtual alcanzan la posición ingresada en la interfaz.

En el Anexo 4 se encuentran las imágenes obtenidas al realizar esta prueba.

3.5 Manual de simulación del brazo robótico.

En la Figura 3.65 se presenta el código QR del video de la simulación del brazo robótico, donde se explica de manera detallada sobre el uso de la simulación.



Figura 3.65 Código QR de la simulación del brazo robótico

4 CONCLUSIONES Y RECOMENDACIONES

4.1 Conclusiones

- El brazo robótico con el que se trabajó en este proyecto es de tipo serial, que tiene: 4 articulaciones rotacionales que son los grados de libertad, 4 eslabones móviles y un eslabón fijo que es considerado la base del brazo robótico. Además, tiene una estructura que facilita el análisis de cinemática directa debido a la ubicación de sus articulaciones.
- En la exportación del brazo robótico se obtuvo un diagrama de bloques en Simulink que consta de: 6 piezas rígidas que son los eslabones, 4 articulaciones incluido la pinza y un sistema de referencia de coordenadas que es la base del modelo mecánico.
- En este proyecto se empleó el método de Denavit Hartenberg para realizar el análisis de cinemática directa, ya que este método permite considerar la estructura y el número de articulaciones que tiene el brazo robótico seleccionado.
- En el análisis de cinemática directa se obtuvieron 10 matrices de transformación, debido al número de pasos empleados para obtener los parámetros Denavit Hartenberg. Se multiplicaron las matrices de transformación para obtener la matriz resultante y conocer la posición del efector final del brazo robótico.
- En la prueba realizada entre Autodesk Inventor y Simulink se determinó un error máximo de 0.58 %, confirmando que el obtener los parámetros de Denavit Hartenberg de un brazo robótico desde Autodesk Inventor y al aplicarlos en la cinemática directa se reduce el error al conocer la posición del efector final.
- Al realizar la prueba en Simulink entre el valor calculado y el valor medido se determinó un error máximo de 1.1 %, por el motivo que el valor medido se obtuvo del centro del efector final del robot virtual y no en la posición determinada en el análisis de cinemática directa.
- En la prueba realizada entre el ángulo giro ingresado mediante la interfaz del brazo robótico y el medido en el robot virtual en Simulink se determinó un error máximo de 0.9 % referente a la posición angular de cada articulación, mismo que es justificado por que el valor de giro ingresado en la interfaz se encuentra en unidades de grados y el valor medido en Simulink tiene unidades en radianes.

- Si las pruebas de funcionamiento se realizarán en un brazo robótico real el error incrementará debido a los parámetros inerciales como son: la inercia, el peso, la fricción, etc.
- El manual de usuario se realizó con la finalidad de que cualquier persona que tenga conocimientos básicos sobre programación, uso de Simulink de Matlab y robótica, sea capaz de utilizar la simulación de un brazo robótico de 4 grados de libertad.

4.2 Recomendaciones

- Antes de realizar la vinculación entre Autodesk Inventor y Simulink es necesario verificar que la configuración regional en el sistema operativo del ordenador esté utilizando el punto '.' como símbolo decimal, debido que al realizar la importación del brazo robótico se presenta un error si la configuración regional está utilizando la coma ',' como símbolo decimal.
- Al utilizar los bloques de la librería de Simulink se debe verificar las unidades que tiene por defecto, para que al momento de utilizarlos con el diagrama del brazo robótico todos tengan las mismas unidades y de esta manera evitar errores al realizar la simulación.
- Como un trabajo complementario del proyecto, se recomienda realizar el análisis de cinemática inversa para este brazo robótico.
- Se recomienda realizar la construcción de este brazo robótico, para aplicar la cinemática directa en este mecanismo.
- La Escuela Politécnica Nacional brinda la oportunidad a sus estudiantes de utilizar el programa Matlab con licencia, por este motivo se recomienda a la Escuela de Formación de Tecnólogos la enseñanza de este programa para utilizarlo en diferentes aplicaciones como es el estudio de la robótica.

5 REFERENCIAS BIBLIOGRÁFICAS

- [1] Onshape, «OpenManipulator-X (RM-X52-TNM)». <https://cad.onshape.com/documents/9442f03bd8ccac084fda9dd3/w/039e8dbd53e0782540ea5b0d/e/6f08aa8ac3d3e5b3054f7782>.
- [2] Mathworks, «Simscape Multibody Link 6.0 – Release 2018b». https://la.mathworks.com/campaigns/offers/download_smlink_confirmation.html?elqsid=1607480957797&potential_use=Student (accedido dic. 12, 2020).
- [3] E. F. Aguilar Correa, Adrián Patricio; Loaiza Vanegas, *Desarrollo de un manual de Guías de prácticas orientado al aprendizaje de la robótica industrial*. Cuenca, 2020.
- [4] Escuela Politécnica Nacional, «Escuela de Formación de Tecnólogos». <https://esfot.epn.edu.ec/index.php/oferta-academica/em> (accedido ene. 15, 2021).
- [5] J. S. Markley Freire, *Simulación de un robot con 6 grados de libertad utilizando Toolbox Robotics del software Matlab*. Quito, 2020.
- [6] A. F. Oña Ñacata, *Diseño e implementación de un brazo robótico controlado por sensores incorporados brazales, para la emulación del movimiento de un brazo humano*. Quito, 2014.
- [7] V. H. Oña Ñacata, A. F. & Vimos Tixi, *Operación remota y control adaptativo de un manipulador virtual de 3 grados de libertad - Oña Ñacata, Alex Fabricio, Vimos Tixi, Victor Hugo - 2019.pdf*, Escuela Po. Quito, 2019.
- [8] W. G. Salazar Patín, *Diseño de una Interfaz de usuario y control cinemático de un brazo robótico de 6 grados de libertad para la planificación de trayectorias en software Matlab y Simulink.*, I. Guayaquil, 2015.
- [9] M. Gualan, M; Pintado, *Elaboración de un módulo de guías de práctica para el aprendizaje y entrenamiento en el desarrollo de sistemas scada*. Cuenca, 2019.
- [10] M. A. Troncos RioFrio, *Diseño y ensamblaje de un brazo robot como módulo de laboratorio para el escaneo de curvas en 3D*. Piura, 2016.
- [11] I. C. Villa Escudero, *Modelamiento y simulación de un algoritmo para el control del brazo robótico*. Riobamba, 2017.

- [12] S. Miller, «What is Simscape Multibody», 3 de marzo, 2016. <https://la.mathworks.com/videos/simscape-multibody-overview-117986.html> (accedido ene. 05, 2021).
- [13] S. Miller, «What is Matlab Function», 4 de marzo, 2016. <https://la.mathworks.com/help/matlab/ref/function.html> (accedido ene. 05, 2021).
- [14] S. Miller, «What is From Workspace», 4 de marzo, 2016. <https://la.mathworks.com/help/simulink/sref/fromworkspace.html> (accedido ene. 05, 2021).
- [15] S. Miller, «What is To Workspace», 15 de mayo, 2017. <https://la.mathworks.com/help/simulink/sref/toworkspace.html> (accedido ene. 05, 2021).
- [16] S. Miller, «What is Display», 22 de abril, 2017. <https://la.mathworks.com/help/simulink/sref/dashboarddisplay.html> (accedido ene. 05, 2021).
- [17] S. Miller, «What is Simulink-Ps Converter», 12 de junio, 2018. <https://la.mathworks.com/help/physmod/simscape/ref/simulinkpsconverter.html> (accedido ene. 05, 2021).
- [18] S. Miller, «What is PS-Simulink Converter», 27 de mayo, 2019. <https://la.mathworks.com/help/physmod/simscape/ref/pssimulinkconverter.html> (accedido ene. 05, 2021).
- [19] S. Miller, «What is Transform Sensor», 27 de mayo, 2019. <https://la.mathworks.com/help/physmod/sm/ref/transformsensor.html> (accedido ene. 05, 2021).
- [20] O. C, *Apuntes de mecánica-dinámica*. Piura, 2015.
- [21] P. F. ardos, Eileen; Fernández, Adel; Marrero Osorio, Sergio A; Guardado, «Scielo, Modelos cinemático y dinámico de un robot de cuatro grados de libertad», *Diciembre*, 2017. http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S1815-59282017000300006 (accedido ene. 05, 2021).
- [22] Asnalema Condo, A., 2021. *Desarrollo de un sistema de visión artificial a través de una tarjeta de desarrollo para controlar el brazo robótico Mitsubishi*. [online] Dspace.ups.edu.ec. Available at:

<<https://dspace.ups.edu.ec/bitstream/123456789/14537/1/UPS%20-%20ST003185.pdf>> [Accessed 28 April 2021].

[23] J. Carvajal, "Modelamiento y diseño de robots industriales", Universidad de la Salle.

ANEXOS

ANEXO 1: CERTIFICADO DE FUNCIONAMIENTO



ESCUELA POLITECNICA NACIONAL

Campus Politécnico "J. Rubén Orellana R

Quito, 13 de mayo de 2021

CERTIFICADO DE FUNCIONAMIENTO DE PROYECTO DE TITULACIÓN

Yo, *Catalina Elizabeth Armas Freire*, docente a tiempo completo de la Escuela Politécnica Nacional y como director de este trabajo de titulación, certifico que he constatado el correcto funcionamiento de la simulación en el entorno Simulink de MATLAB de un brazo robótico de 4 grados de libertad (GDL), la cual fue simulada por el estudiante Alex Humberto Guacho Tituaña

El proyecto cumple con los requerimientos de diseño y parámetros necesarios para que los usuarios de la ESFOT puedan usar la simulación con seguridad para los equipos y las personas.

DIRECTOR

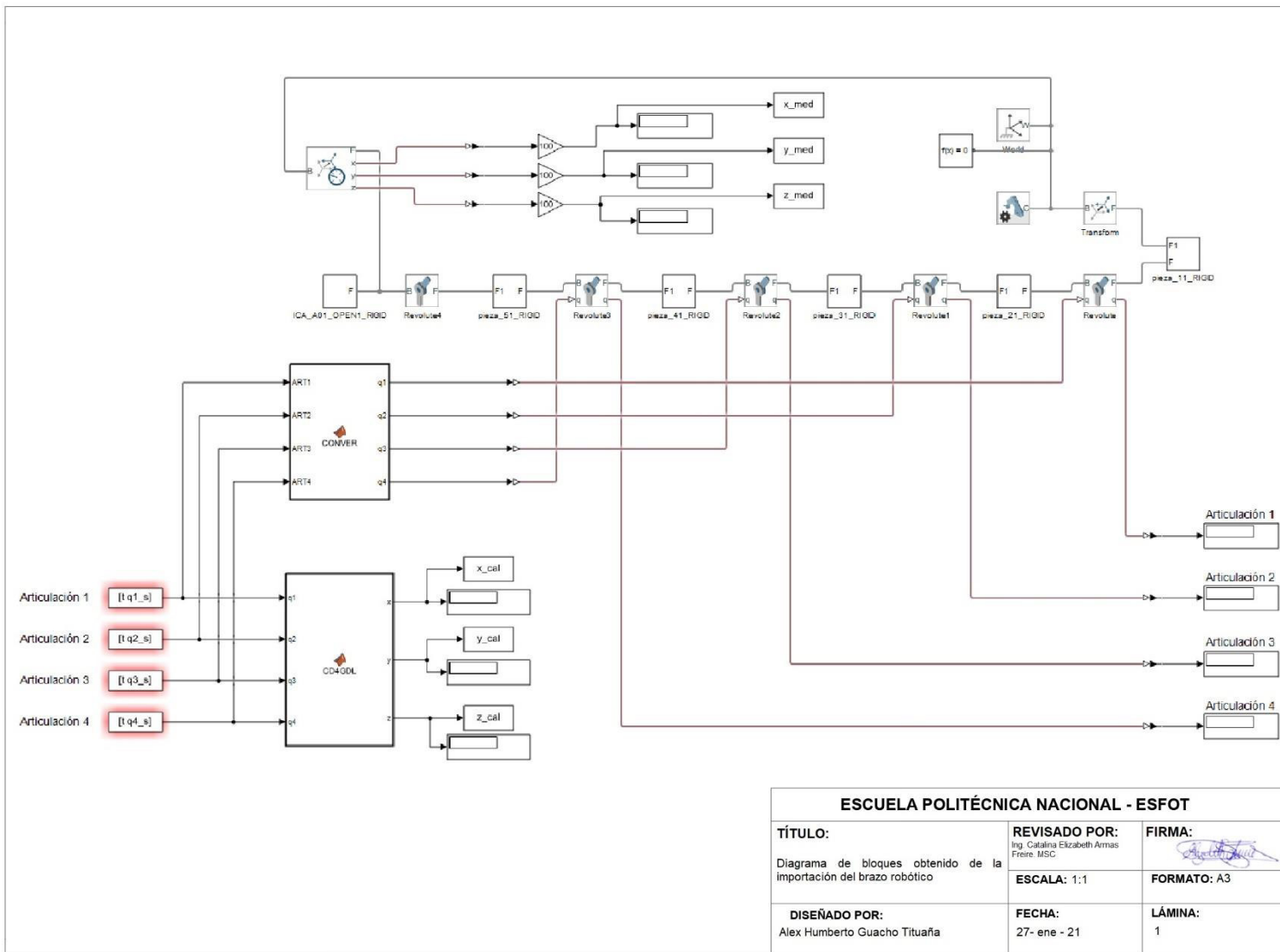
Ing. Catalina Elizabeth Armas Freire., Msc.


Ladrón de Guevara E11-253, Escuela de Formación de Tecnólogos, EXT: 3115

email: elizabeth.armas@epn.edu.ec

Quito-Ecuador

ANEXO 2: PLANOS Y ESQUEMAS



ESCUELA POLITÉCNICA NACIONAL - ESFOT		
TÍTULO: Diagrama de bloques obtenido de la importación del brazo robótico	REVISADO POR: Ing. Catalina Elizabeth Armas Freré MSC	FIRMA: 
DISEÑADO POR: Alex Humberto Guacho Tituaña	ESCALA: 1:1	FORMATO: A3
	FECHA: 27- ene - 21	LÁMINA: 1

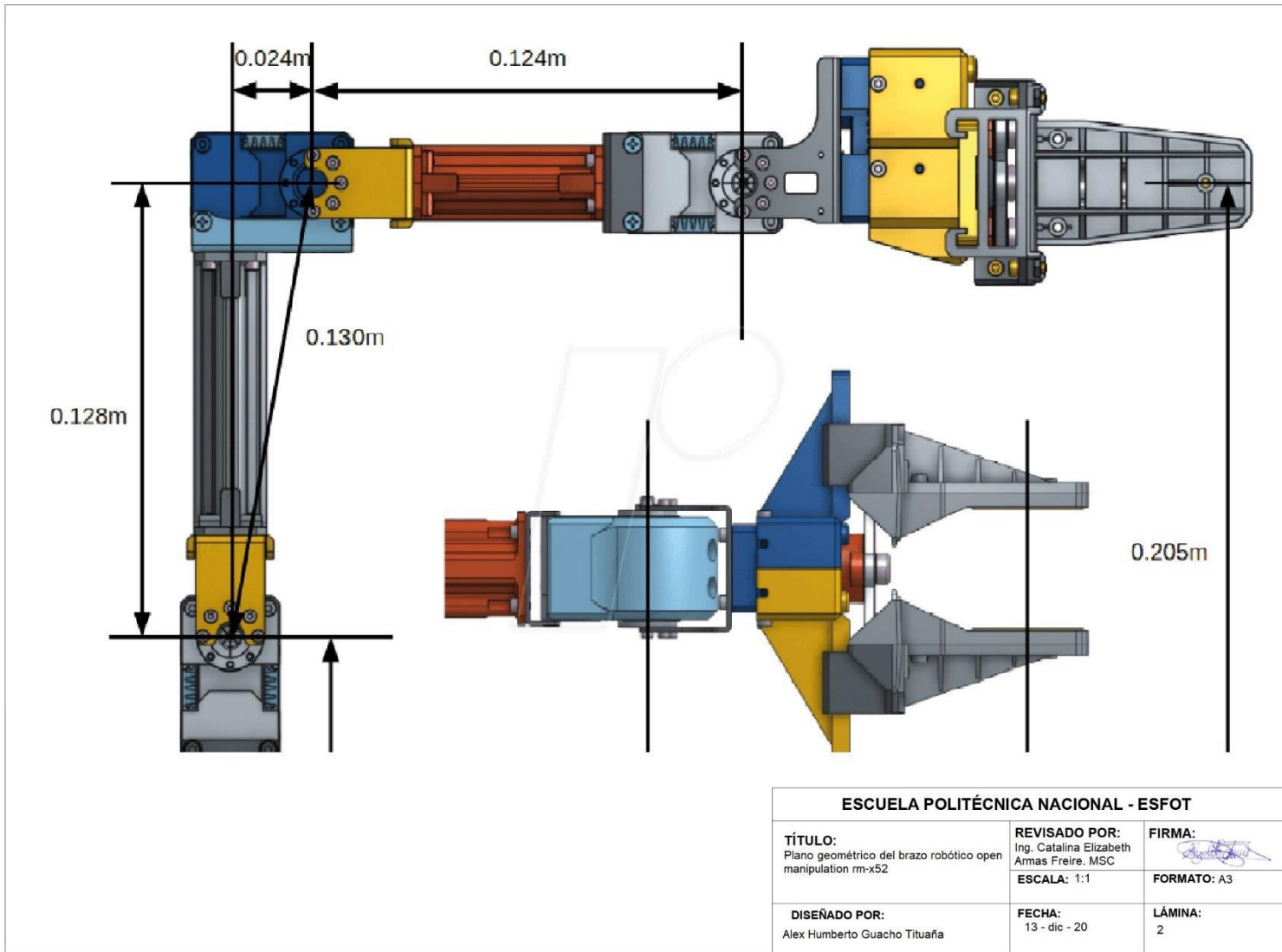
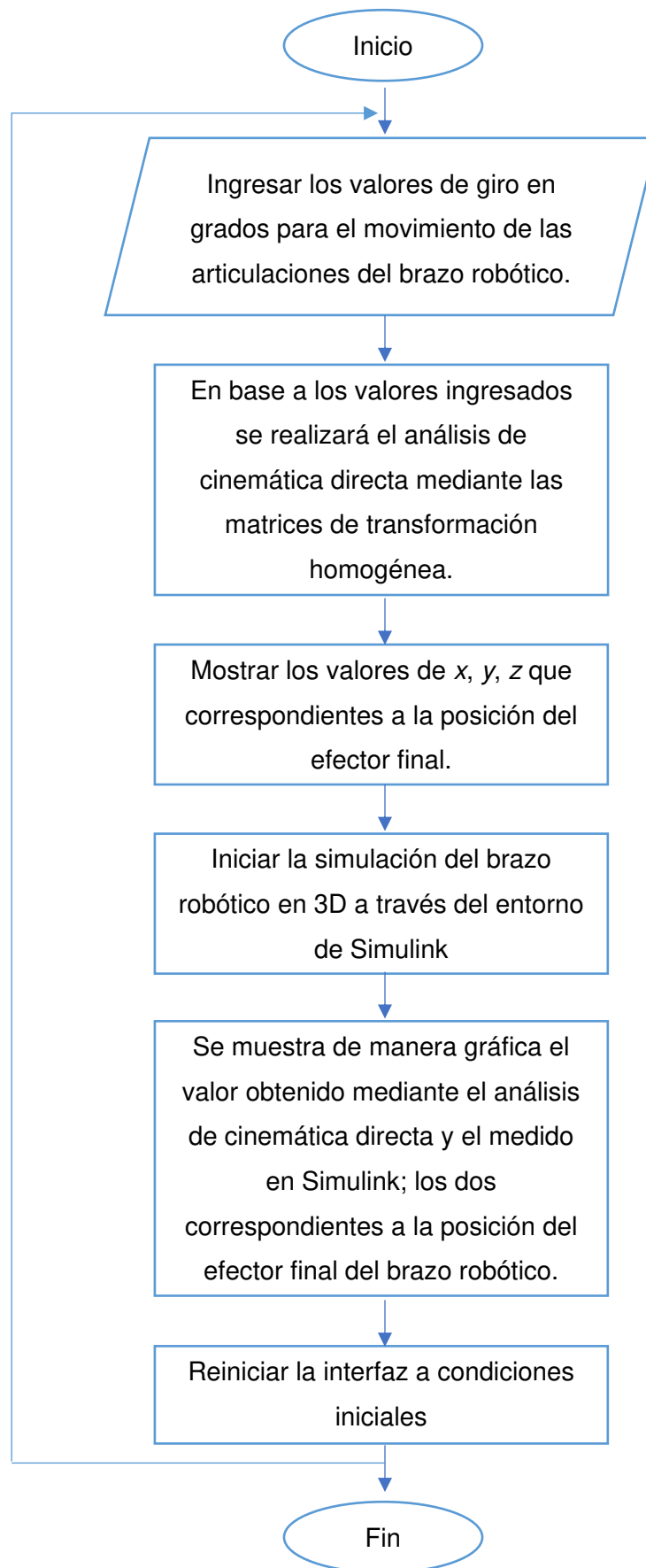


Diagrama de flujo de funcionamiento de la interfaz



ANEXO 3: CÓDIGOS DE PROGRAMACIÓN.

Código para el análisis de cinemática directa.

```
%En la función se asigna cuáles serán las entradas y las
%salidas, también cual es el nombre que llevará el bloque del Matlab
%Function que en esta ocasión es CD4GDL que significa cinemática
directa de
%4 grados de libertad.
function [x,y,z] = CD4GDL(q1,q2,q3,q4)
%Parámetros constantes
%Ángulos de rotación(°) alrededor del eje x
alpha1=270;
alpha2=90;
alpha3=270;
alpha4=0;
alpha5=90;
alpha6=0;
% Valor de los desplazamientos(cm)
d1=5.95;
d2=12.80;
a1=2.40;
a2=12.40;
a3=6.75;
%En esta parte del código se encuentran las ecuaciones cinemáticas
%que se obtuvieron al realizar el análisis de cinemática directa
H1=q1*-1;

T1=[1      0      0      0;
    0  cosd(alpha1) -sind(alpha1) 0;
    0  sind(alpha1)  cosd(alpha1) 0;
    0      0      0      1;]

T2=[cosd(H1)      -sind(H1)      0      0;
    sind(H1)      cosd(H1)      0      0;
    0              0      1      d1;
    0              0      0      1;]

T3=[1      0      0      0;
    0  cosd(alpha2) -sind(alpha2) 0;
    0  sind(alpha2)  cosd(alpha2) 0;
    0      0      0      1;]

T4=[cosd(q2)      -sind(q2)      0      0;
    sind(q2)      cosd(q2)      0      0;
    0              0      1      0;
    0              0      0      1;]

T5=[1      0      0      0;
    0  cosd(alpha3) -sind(alpha3) 0;
    0  sind(alpha3)  cosd(alpha3) 0;
    0      0      0      1;]

T6=[cosd(alpha4)      -sind(alpha4)      0      0;
    sind(alpha4)      cosd(alpha4)      0      0;
    0              0      1      d2;
    0              0      0      1;]
```

```

T7=[1          0          0          0;
    0      cosd(alpha5) -sind(alpha5) 0;
    0      sind(alpha5)  cosd(alpha5) 0;
    0          0          0          1;]

T8=[cosd(alpha6)  -sind(alpha6)  0  a1*cosd(alpha6);
    sind(alpha6)  cosd(alpha6)  0  a1*sind(alpha6);
    0              0              1  0;
    0              0              0  1;]

T9=[cosd(q3)  -sind(q3)  0  a2*cosd(q3);
    sind(q3)  cosd(q3)  0  a2*sind(q3);
    0          0          1  0;
    0          0          0  1;]

T10=[cosd(q4)  -sind(q4)  0  a3*cosd(q4);
    sind(q4)  cosd(q4)  0  a3*sind(q4);
    0          0          1  0;
    0          0          0  1;]

%Para obtener la matriz resultante se debe multiplicar todas las
matrices
%anteriores
T=T1*T2*T3*T4*T5*T6*T7*T8*T9*T10;
%Se extrae de la matriz resultante los valores de x, y, z que indican
cual
%es la posición del efector final. Estos valores se encuentran en la
fila 1,
%2 y 3 en la columna 4.
x=[T(1,4)];
y=[T(2,4)];
z=[T(3,4)];

%Fin del programa para el análisis de cinemática directa.
end

```

Código de programa utilizado para crear la interfaz.

Botón Simular

```

%En esta parte del Código de programa se podrá ingresar los ángulos de
giro en grados
%para las articulaciones del brazo robótico
%Ángulo de giro para la Articulación 1
a=str2double(get(handles.art1,'String'));
t=(0:0.01:10)';
q1_s=(a/10).*t;
assignin('base','t',t);
assignin('base','q1_s',q1_s);

%ingresar el valor de giro para la articulación 2
b=str2double(get(handles.art2,'String'));
t=(0:0.01:10)';
q2_s=(b/10).*t;
assignin('base','q2_s',q2_s);

%ingresar el valor de giro para la articulación 3

```

```

c=str2double(get(handles.art3,'String'));
t=(0:0.01:10)';
q3_s=(c/10).*t;
assignin('base','q3_s',q3_s);

%ingresar el valor de giro para la articulación 4
d=str2double(get(handles.art4,'String'));
t=(0:0.01:10)';
q4_s=(d/10).*t;
assignin('base','q4_s',q4_s);

%Se realizará el análisis de Denavit Hartenberg para conocer las
posiciones
%del efector final en los ejes x, y, z

q1=str2double(get(handles.art1,'String'));
q2=str2double(get(handles.art2,'String'));
q3=str2double(get(handles.art3,'String'));
q4=str2double(get(handles.art4,'String'));

%Parámetros de Denavit Hartenberg
%ángulos de giro alrededor del eje x
alpha1=270;
alpha2=90;
alpha3=270;
alpha4=0;
alpha5=90;
alpha6=0;
%Valor de las distancias medidas
d1=5.95;
d2=12.80;
a1=2.40;
a2=12.40;
a3=6.75;
%Matrices de transformación obtenidas para el análisis de cinemática
directa
H1=q1*-1;

T1=[1      0      0      0;
    0  cosd(alpha1) -sind(alpha1)  0;
    0  sind(alpha1)  cosd(alpha1)  0;
    0      0      0      1;]

T2=[cosd(H1)      -sind(H1)      0      0;
    sind(H1)      cosd(H1)      0      0;
    0      0      1      d1;
    0      0      0      1;]

T3=[1      0      0      0;
    0  cosd(alpha2) -sind(alpha2)  0;
    0  sind(alpha2)  cosd(alpha2)  0;
    0      0      0      1;]

T4=[cosd(q2)      -sind(q2)      0      0;
    sind(q2)      cosd(q2)      0      0;
    0      0      1      0;
    0      0      0      1;]

```

```

T5=[1      0      0      0;
    0      cosd(alpha3)  -sind(alpha3)  0;
    0      sind(alpha3)   cosd(alpha3)  0;
    0      0      0      1;]

T6=[cosd(alpha4)  -sind(alpha4)  0      0;
    sind(alpha4)   cosd(alpha4)  0      0;
    0              0              1      d2;
    0              0              0      1;]

T7=[1      0      0      0;
    0      cosd(alpha5)  -sind(alpha5)  0;
    0      sind(alpha5)   cosd(alpha5)  0;
    0      0      0      1;]

T8=[cosd(alpha6)  -sind(alpha6)  0      a1*cosd(alpha6);
    sind(alpha6)   cosd(alpha6)  0      a1*sind(alpha6);
    0              0              1      0;
    0              0              0      1;]

T9=[cosd(q3)  -sind(q3)  0      a2*cosd(q3);
    sind(q3)   cosd(q3)  0      a2*sind(q3);
    0          0          1      0;
    0          0          0      1;]

T10=[cosd(q4)  -sind(q4)  0      a3*cosd(q4);
    sind(q4)   cosd(q4)  0      a3*sind(q4);
    0          0          1      0;
    0          0          0      1;]

%Multiplicación de toda matriz para obtener una matriz resultante
T=T1*T2*T3*T4*T5*T6*T7*T8*T9*T10;
%Extraer los valores de la columna 1, 2, 3 correspondientes a la fila
4;
%estos valores pertenecen a los ejes x, y, z
x=[T(1,4)];
y=[T(2,4)];
z=[T(3,4)];

%se muestra los valores de x, y, z en la interfaz
set(handles.pos_x, 'String', num2str(x));
set(handles.pos_y, 'String', num2str(y));
set(handles.pos_z, 'String', num2str(z));

%Con este botón se cierra la ventana de la interfaz.
%Ingresados los ángulos de giro para las articulaciones se iniciará la
%simulación del brazo robótico en Simulink.
sim('brazoRob_tico.slx')
%Graficar la posición X del brazo robot
axes(handles.axes11);
t=(0:0.01:10)';
plot(t,x_cal);
hold on
plot(t,x_med);
%Graficar la posición Y del brazo robot
axes(handles.axes12);
t=(0:0.01:10)';

```

```

plot(t,y_cal);
hold on
plot(t,y_med);
%Graficar la posición Z del brazo robot
axes(handles.axes13);
t=(0:0.01:10)';
plot(t,z_cal);
hold on
plot(t,z_med);

function pushbutton8_Callback(hObject, eventdata, handles)
%Con este botón se limpiará por completo la interfaz
%se limpiará las gráficas de posición x, y, z en cada uno de los
axes
cla(handles.axes1, 'reset');
cla(handles.axes2, 'reset');
cla(handles.axes3, 'reset');

```

Botón reiniciar

```

%Con este botón se limpiará por completo la interfaz
%se limpiará las gráficas de posición x, y, z en cada uno de los axes
cla(handles.axes11, 'reset');
cla(handles.axes12, 'reset');
cla(handles.axes13, 'reset');
%Se limpiará los valores ingresados para que giren las articulaciones
%también los valores obtenidos de la posición del efector final
s='';
sx=num2str(s);
set(handles.art1, 'string', sx);
set(handles.art2, 'string', sx);
set(handles.art3, 'string', sx);
set(handles.art4, 'string', sx);
set(handles.pos_x, 'string', sx);
set(handles.pos_y, 'string', sx);
set(handles.pos_z, 'string', sx);

```


ANEXO 4: PRUEBAS DE FUNCIONAMIENTO

Imágenes obtenidas de la prueba realizada entre Simulink y Autodesk Inventor.

Prueba N°1

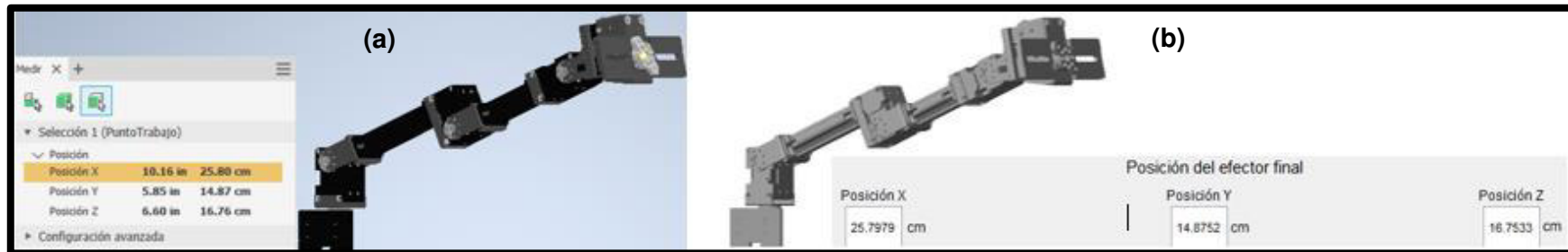


Figura A4. 1 Prueba N°1 (a) Valor obtenido en Autodesk Inventor (b) Valor obtenido en Simulink

Prueba N°2



Figura A4. 2 Prueba N°2 (a) Valor obtenido en Autodesk Inventor (b) Valor obtenido en Simulink

Prueba N°3

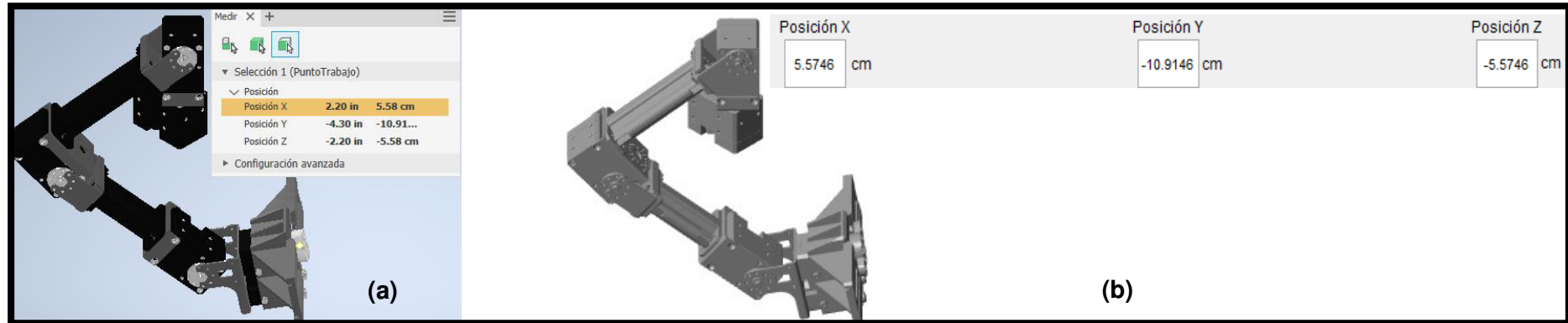


Figura A4. 3 Prueba N°3 (a) Valor obtenido en Autodesk Inventor (b) Valor obtenido en Simulink

Prueba N°4

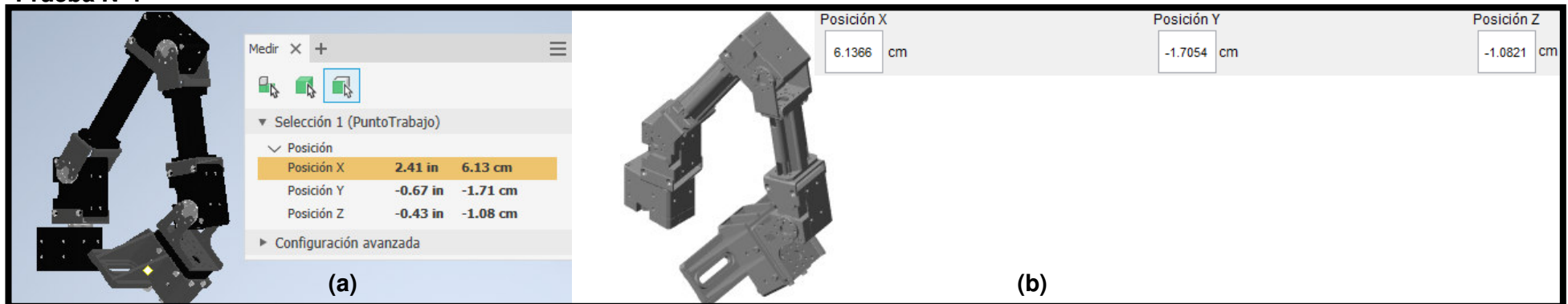


Figura A4. 4 Prueba N°4 (a) Valor obtenido en Autodesk Inventor (b) Valor obtenido en Simulink

Imágenes obtenidas del valor calculado y el valor medido para la posición del efector final del brazo robótico.

Prueba N°1

■ Valor calculado

■ Valor medido

■ Valor calculado

■ Valor medido

■ Valor calculado

■ Valor medido

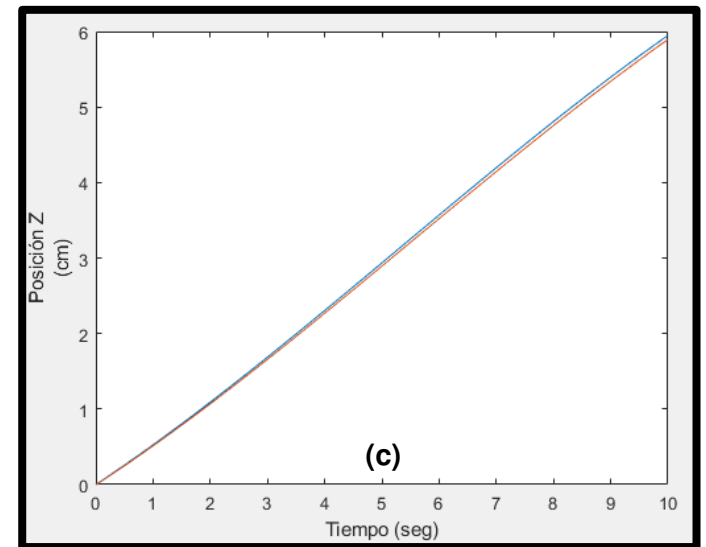
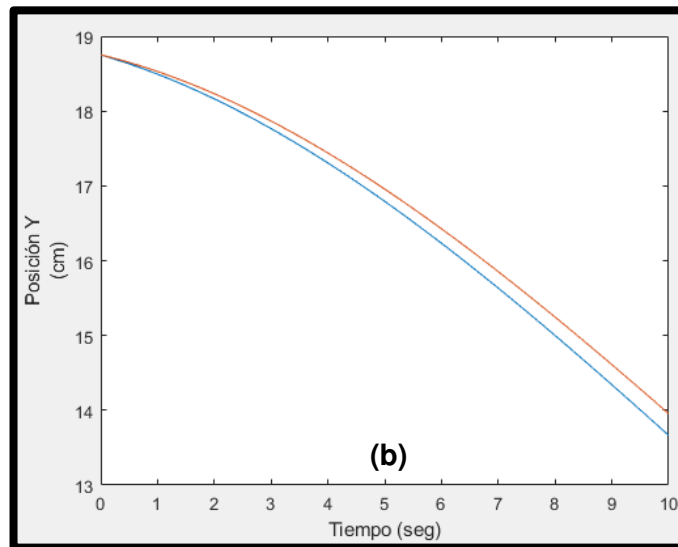
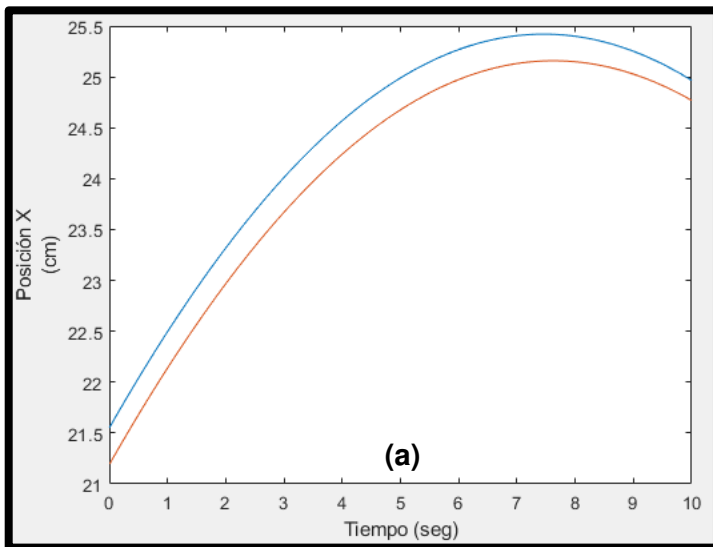


Figura A4. 5 Prueba N°1 (a) Posición x (b) Posición y (c) Posición z

Prueba N°2

Valor calculado

Valor medido

Valor calculado

Valor medido

Valor calculado

Valor medido

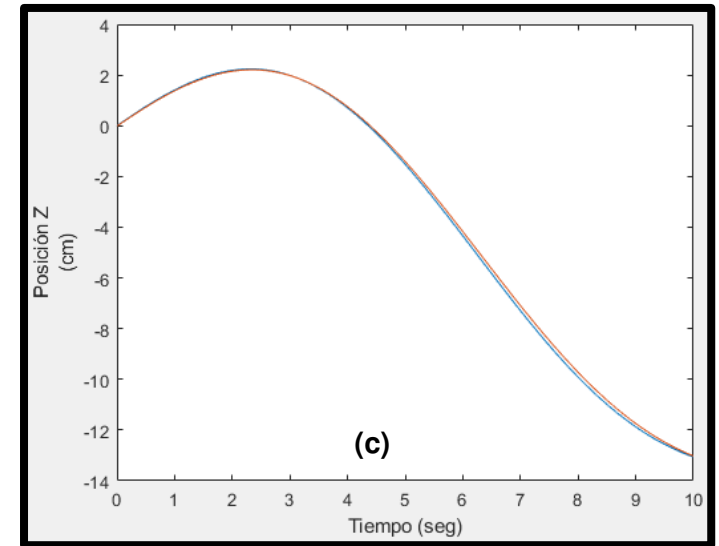
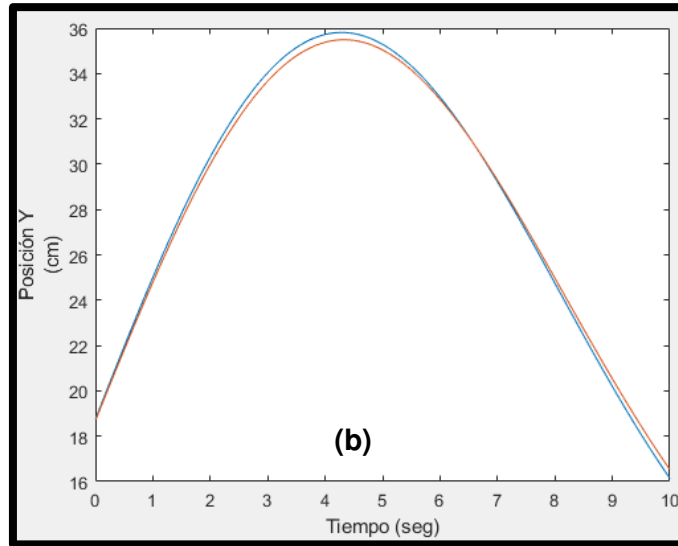
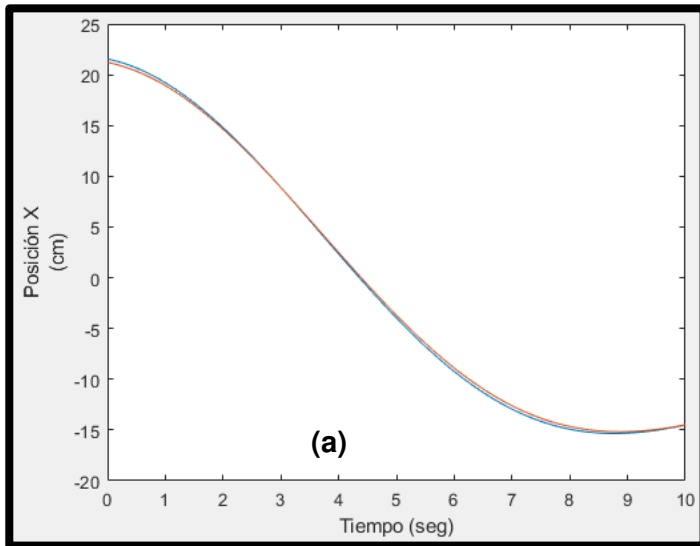


Figura A4. 6 Prueba N°2 (a) Posición x (b) Posición y (c) Posición z

Prueba N°3

Valor calculado

Valor medido

Valor calculado

Valor medido

Valor calculado

Valor medido

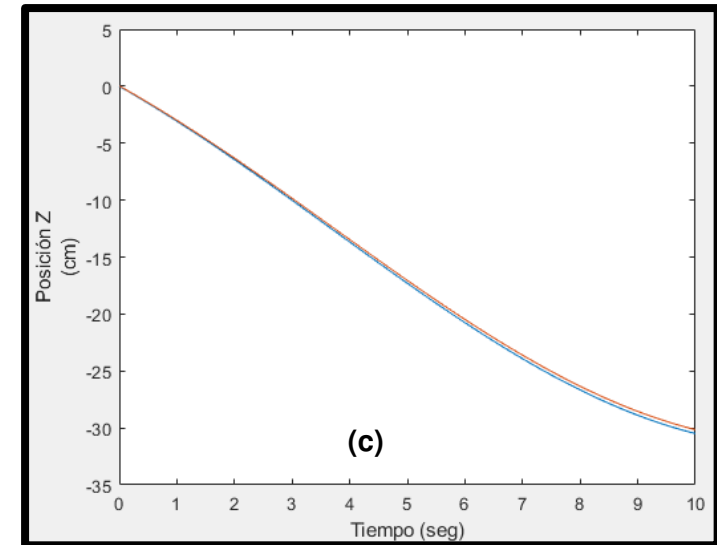
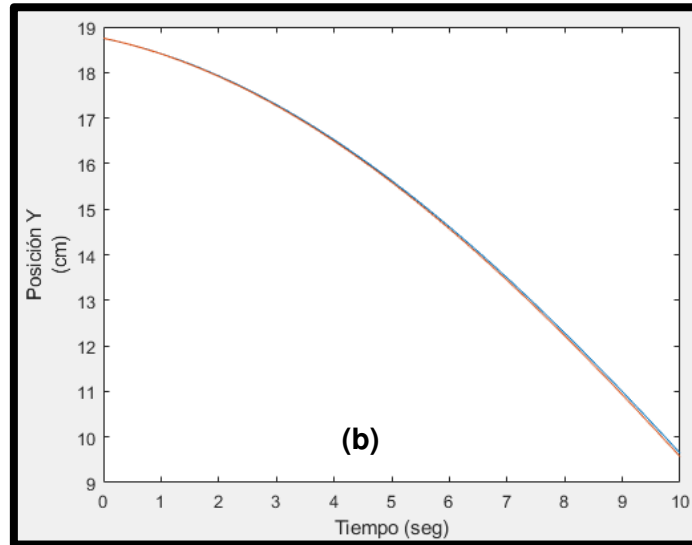
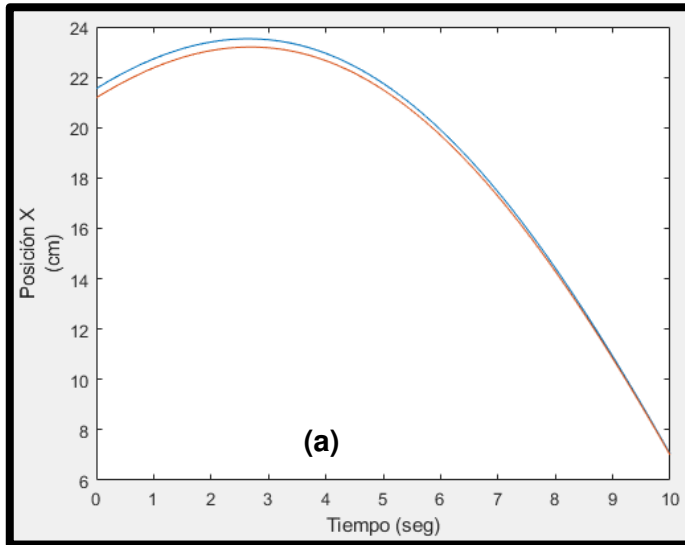


Figura A4. 7 Prueba N°3 (a) Posición x (b) Posición y (c) Posición z

Prueba N°4

Valor calculado

Valor medido

Valor calculado

Valor medido

Valor calculado

Valor medido

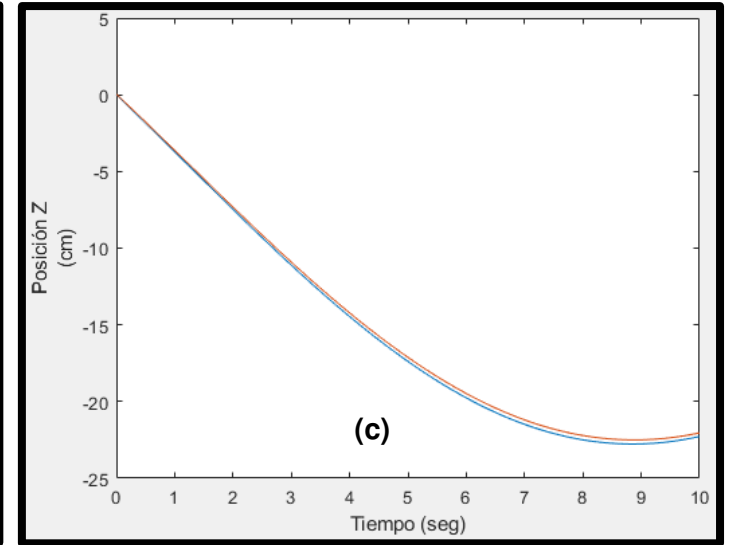
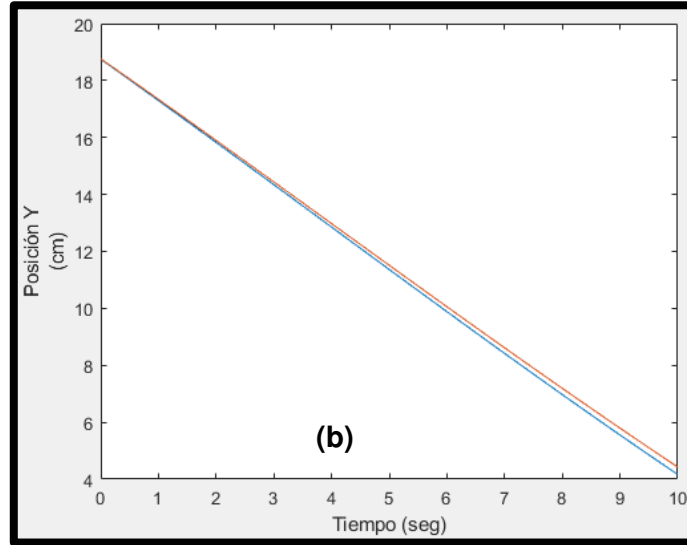
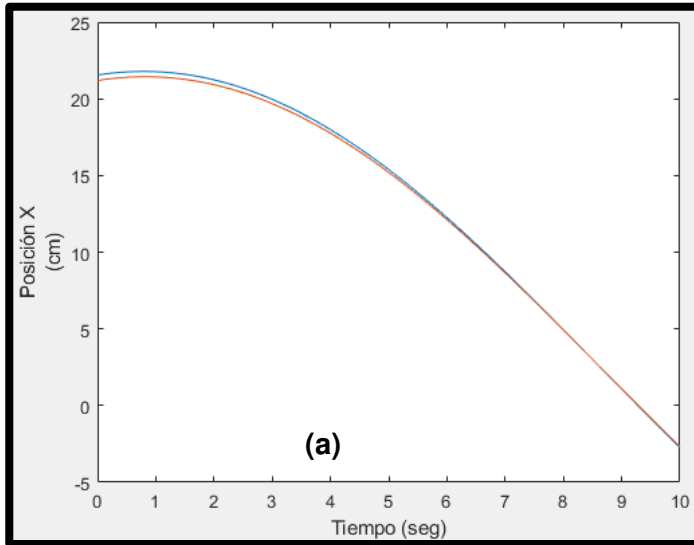


Figura A4. 8 Prueba N°4 (a) Posición x (b) Posición y (c) Posición z

Imágenes obtenidas de los ángulos de giro ingresados y los obtenidos para cada articulación

Prueba N°1

Valores ingresados en grados

Ingrese los valores de giro

<input type="text" value="56"/>	°	<input type="text" value="22"/>	°	<input type="text" value="-33"/>	°	<input type="text" value="-23"/>	°
Articulación 1		Articulación 2		Articulación 3		Articulación 4	

Figura A4. 9 Valores giro ingresados en la Interfaz para la prueba N°1

Valores obtenidos en radianes

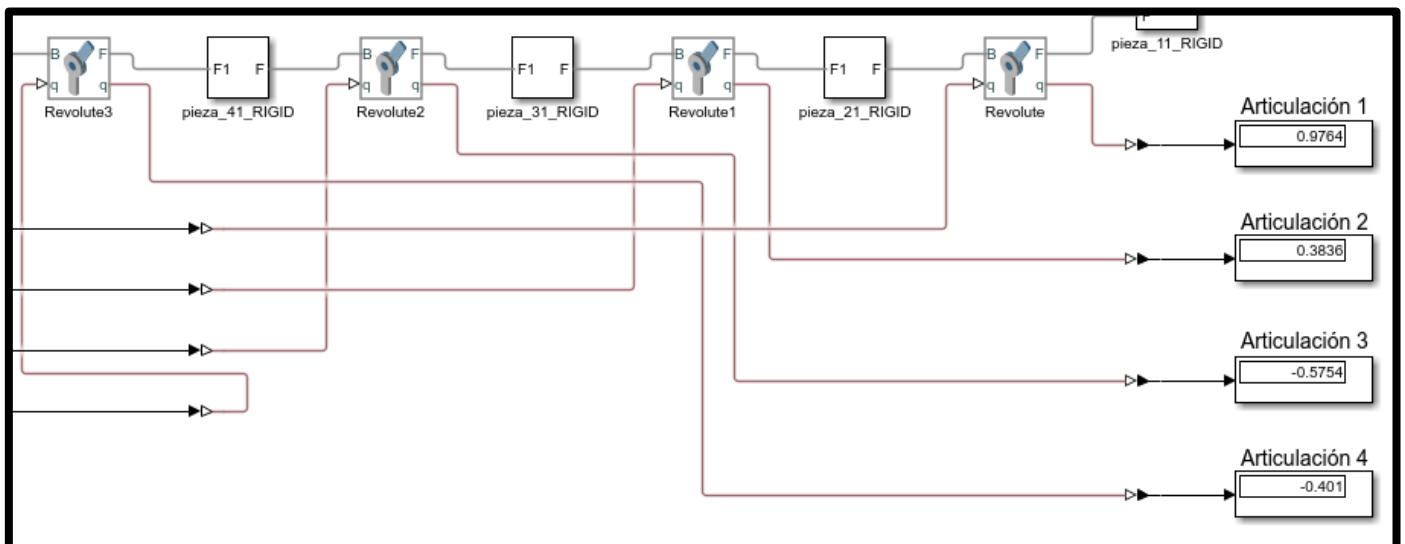


Figura A4. 10 Valores de giro medidos en Simulink referente a la prueba N°1

Prueba N°2

Valores ingresados en grados

Ingrese los valores de giro

<input type="text" value="44"/>	°	<input type="text" value="73"/>	°	<input type="text" value="12"/>	°	<input type="text" value="98"/>	°
Articulación 1		Articulación 2		Articulación 3		Articulación 4	

Figura A4. 11 Valores giro ingresados en la Interfaz para la prueba N°2

Valores obtenidos en radianes

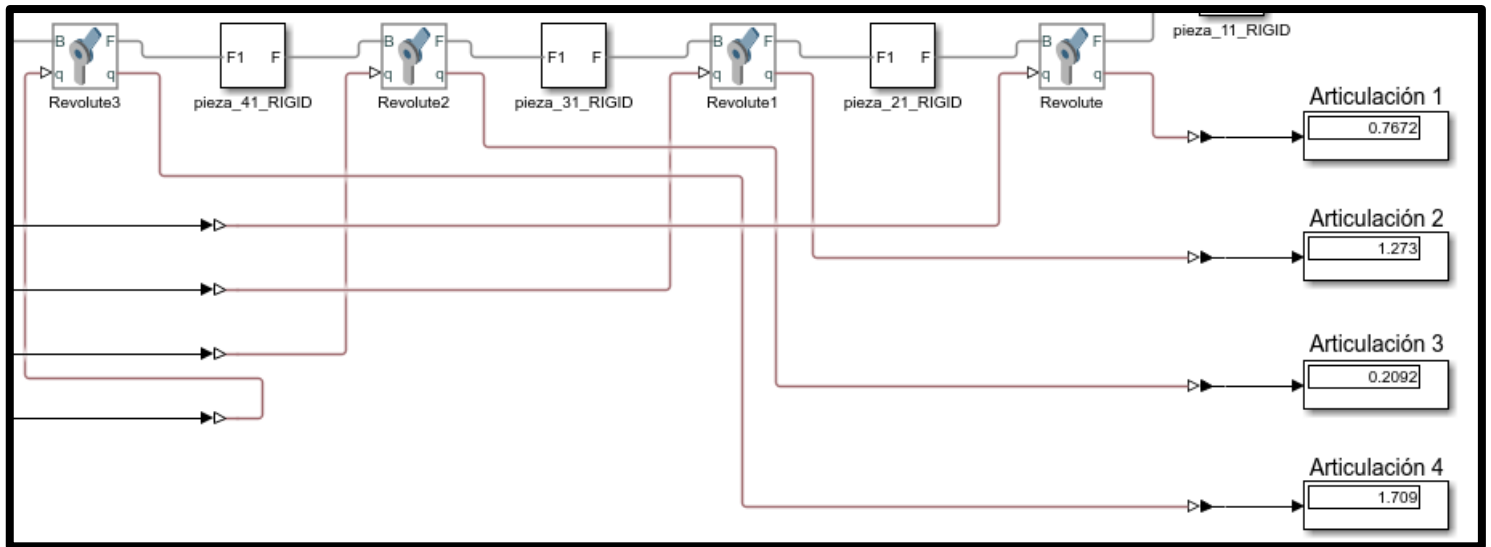


Figura A4. 12 Valores de giro medidos en Simulink referente a la prueba N°2

Prueba N°3

Valores ingresados en grados

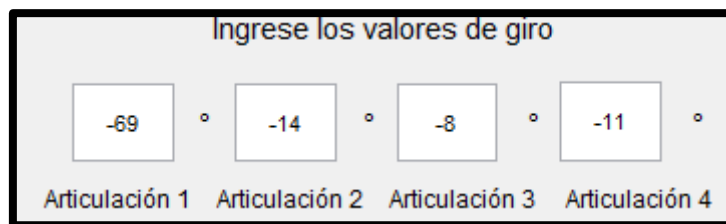


Figura A4. 13 Valores giro ingresados en la Interfaz para la prueba N°3

Valores obtenidos en radianes

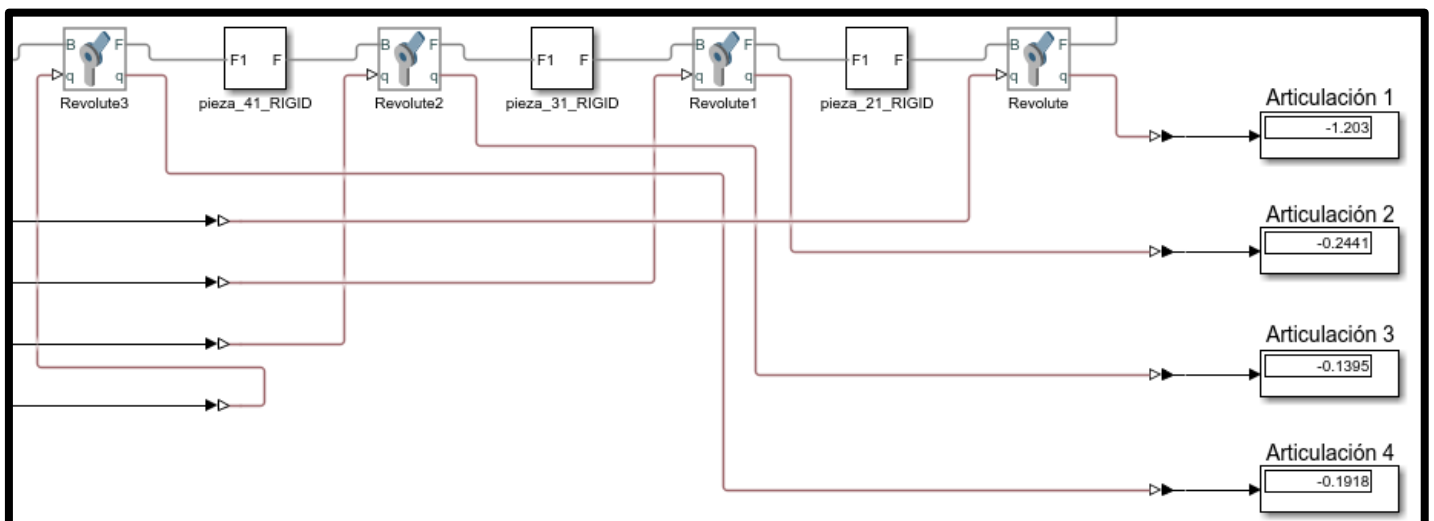


Figura A4. 14 Valores de giro medidos en Simulink referente a la prueba N°3

Prueba N°4

Valores ingresados en grados

Ingrese los valores de giro

27	°	50	°	20	°	30	°
Articulación 1		Articulación 2		Articulación 3		Articulación 4	

Figura A4. 15 Valores giro ingresados en la Interfaz para la prueba N°4

Valores obtenidos en radianes

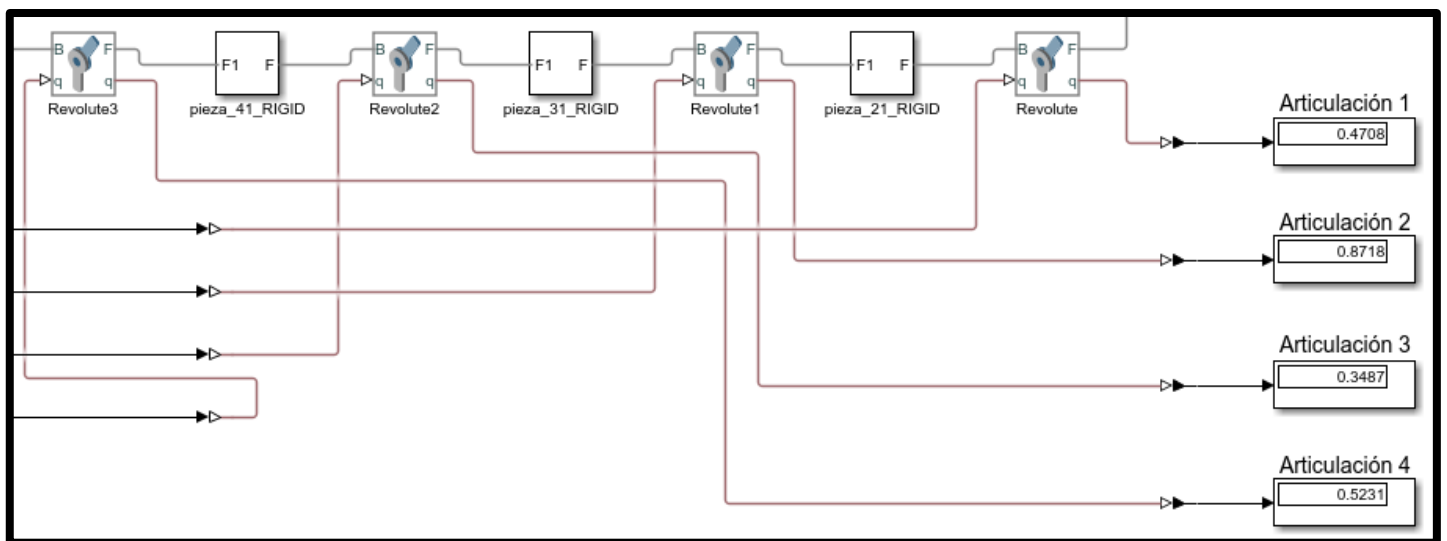


Figura A4. 16 Valores de giro medidos en Simulink referente a la prueba N°4