

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

Facultad de Ingeniería en Electricidad y Computación

**“DESARROLLO DE UNA SIMULACIÓN DE UN ROBOT DE 6
GRADOS DE LIBERTAD EN MATLAB APLICADA A
SERVOMOTORES EN EL ENTORNO DE STUDIO 5000”**

INFORME DE PROYECTO INTEGRADOR

Previo a la obtención del Título de:

INGENIERO EN ELECTRÓNICA Y AUTOMATIZACIÓN

LEONARDO DAVID ROMO MORA

KEVIN STEVE VILLOTA VITERI

GUAYAQUIL – ECUADOR

AÑO: 2019

DEDICATORIA

El presente proyecto lo dedico a mis padres, Jackeline Viteri Solórzano y Walter Villota Arca, por todo el esfuerzo, paciencia y confianza brindada a lo largo de esta etapa de mi vida; a mis hermanos, Kenny y Kendrick, por su apoyo incondicional; a mi abuelita, Gricelia Solórzano Cedeño, por brindarme su amor y apoyo absoluto.

Kevin Steve Villota Viteri

El presente proyecto es dedicado a mis padres, Leonardo Romo Medina y Glenda Mora Triana por todo el apoyo brindado en este tiempo y demostrarme su amor absoluto, a mi hermana, Leonela Romo Mora, por enseñarme que ante toda circunstancia hay que mantenerse fuerte.

Leonardo Romo Mora

AGRADECIMIENTOS

Agradezco a Dios, por la vida misma.

A mi familia, por su apoyo, cariño y enseñanzas para alcanzar mis metas.

A mi grupo de amigos, de infancia, Lords e Ivanna, por su apoyo y hacer de este trayecto lo más divertido.

Kevin Steve Villota Viteri

Agradezco a Dios, por el amor recibido.

A mi familia, por su apoyo y cariño incondicional que me han demostrado.

A mis amigos, por darme su mano ante cualquier circunstancia y en especial a mi amigo Freddy Bustos Cerda que aunque no se encuentre con nosotros, fue mi compañero principal para terminar esta etapa.

Leonardo Romo Mora

DECLARACIÓN EXPRESA

"Los derechos de titularidad y explotación, nos corresponde conforme al reglamento de propiedad intelectual de la institución; *Leonardo David Romo Mora* y *Kevin Steve Villota Viteri* damos nuestro consentimiento para que la ESPOL realice la comunicación pública de la obra por cualquier medio con el fin de promover la consulta, difusión y uso público de la producción intelectual"

Leonardo David Romo
Mora

Kevin Steve Villota Viteri

EVALUADORES

Ing. Damián Alberto Larco Gómez

PROFESOR DE LA MATERIA

Ing. Alexander Prieto León

PROFESOR TUTOR

RESUMEN

El presente proyecto tiene como objetivo visualizar la simulación 3D de un robot de 6 grados de libertad y la representación de los movimientos de sus articulaciones en servomotores. El laboratorio de Automatización de la Facultad de Ingeniería en Electricidad y Computación (FIEC), no cuenta con un brazo robótico debido a su alto presupuesto y que éste requiere de un tiempo considerable para su montaje, pero dispone de una gama de dispositivos y softwares de la marca Rockwell Automation con los que se modeló el robot.

Un sistema robótico consiste de una consola Teach Pendant, un robot manipulador y un controlador. Para esto, se modeló el robot y se diseñó una interfaz gráfica en MATLAB para la simulación y control de las articulaciones del robot respectivamente. Por otro lado, el movimiento de sus articulaciones se la aplicó en seis servomotores, controlados por los servo-variadores Kinetix 5500, que fueron configurados en Studio 5000. La comunicación de estos dos entornos se la realizó mediante la arquitectura cliente/servidor, a través de OPC KEPServerEX.

Se proyecta tener un control intuitivo, flexible y con un tiempo de respuesta muy bueno. El usuario puede manejar el movimiento de las articulaciones del robot mediante la GUI y su movimiento es simulado en una interfaz 3D y aplicado a los servomotores.

ABSTRACT

The present project aims to visualize the 3D simulation of a 6 degrees of freedom robot (DOF), and the representation of the movements of its joints in servomotors. The Automation Laboratory of the Faculty of Electrical and Computer Engineering (FIEC), doesn't have a robotic arm due to its high budget and that this requires considerable time for assembly, but it has a range of devices and software of the Rockwell Automation brand with which the robot was modeled.

A robotic system consists of a Teach Pendant console, a robot arm and a controller. For this, the robot was modeled and a graphical interface was designed in MATLAB for the simulation and control of the robot joints respectively. On the other hand, the movement of its joints was applied in six servomotors, controlled by the Kinetix 5500 servo-drives, which were configured in Studio 5000. The communication of these two environments was carried out through the client / server architecture, through OPC KEPServerEX.

It is projected to have an intuitive, flexible control and a very good response time. The user can manage the movement of the robot's joints through the GUI and its movement is simulated in a 3D interface and applied to the servomotors.

Keywords: *DOF, 3D simulation, robotic arm, servomotors, servo-drives, GUI, OPC*

ÍNDICE GENERAL

RESUMEN	I
<i>ABSTRACT</i>	II
ÍNDICE GENERAL.....	III
ABREVIATURAS	VI
SIMBOLOGÍA	VII
ÍNDICE DE FIGURAS.....	VIII
ÍNDICE DE TABLAS	X
CAPÍTULO 1	12
1. Introducción	12
1.1 Descripción del problema	13
1.2 Objetivos	13
1.2.1 Objetivo General	13
1.2.2 Objetivos Específicos	14
1.3 Marco Teórico	14
1.3.1 Robot Industrial	14
1.3.2 Robot Angular o Antropomórfico	14
1.3.3 Cinemática directa de un robot	15
1.3.4 Interfaz Gráfica de Usuario	15
1.3.5 Consola de programación	16
1.3.6 PLC	17
1.3.7 OPC	18
1.3.8 Servomotor.....	18
1.3.9 Servo-variador.....	19

1.3.10	Studio 5000 Logix Designer	20
1.3.11	MATLAB	20
1.3.12	Simulink.....	20
1.3.13	KEPServerEX.....	21
1.3.14	Control de movimiento	21
2.	Metodología	22
2.1.	Descripción del robot	23
2.2	Controlador	24
2.3	Topología de red.....	24
2.4.	Modelado del robot	25
2.4.1	Representación de Denavit-Hartenberg (D-H)	25
2.4.2	Matriz de transformación homogénea	28
2.5	Diseño del sistema robótico en MATLAB Simulink	29
2.5.1	GUI – Teach Pendant.....	30
2.5.2	Sistema de control.....	32
2.5.3	Sistema OPC	35
2.6	Configuración OPC KEPServerEX	36
2.7	Control de movimiento de servomotores en Studio 5000	37
2.7.1	Motion Servo On (MSO)	37
2.7.2	Motion Servo Off (MSF)	38
2.7.3	Motion Axis Move (MAM)	39
CAPÍTULO 3.....		41
3.	ANÁLISIS DE Resultados	41
3.1	Hardware	41
3.1.1	Área de trabajo.....	41
3.2	Software.....	42

3.2.1	Funcionamiento de la GUI – Teach Pendant	42
3.2.2	Simulación 3D del robot	43
3.2.3	Cinemática del robot ABB IRB 120	44
3.2.4	Validación del espacio de trabajo.....	46
3.2.5	Comunicación OPC.....	47
3.2.6	Medición de posiciones angulares	48
3.3	Análisis de costos	50
3.4	Ejemplo de aplicación	51
3.5	Manual de usuario	53
CAPÍTULO 4	55
4.	Conclusiones Y Recomendaciones.....	55
4.1	Conclusiones	55
4.2	Recomendaciones	56
BIBLIOGRAFÍA	57
APÉNDICES	59
ANEXOS	74

ABREVIATURAS

ESPOL	Escuela Superior Politécnica del Litoral
FIEC	Facultad de Ingeniería Eléctrica y Computación
GDL	Grados De Libertad
PLC	Programmable Logic Controller
OLE	Object Linking and Embedding
OPC	OLE for Process Control
MATLAB	MATrix LABoratory
GUI	Graphical User Interface
HMI	Human Machine Interface
ABB	Asea Brown Boveri
CIP	Common Industrial Protocol
D-H	Denavit-Hartenberg
T	Matriz de transformación homogénea
3D	Tres Dimensiones
IP	Internet Protocol
MSF	Motion Servo Off
MSO	Motion Servo On
MAH	Motion Axis Home
MAM	Motion Axis Move

SIMBOLOGÍA

- Grados
- mm Milímetro

ÍNDICE DE FIGURAS

Figura 1.1.1 Robot industrial ABB Series IRB 120.....	14
Figura 1.2. Robot antropomórfico	15
Figura 1.3 Cinemática directa	15
Figura 1.4 Interfaz gráfica de usuario (GUI).....	16
Figura 1.5 Teach pendant de la marca ABB	17
Figura 1.6 Partes de un PLC compacto y modular	18
Figura 1.7 Partes de un servomotor.....	19
Figura 1.8 Servo-variador Kinetix 350 de la marca Allen Bradley.....	20
Figura 1.9 Modelo de un vehículo eléctrico en el entorno Simulink	21
Figura 2.1 Diagrama de la aplicación de simulación de un robot de 6 GDL orientado a servomotores	23
Figura 2.2 Elementos del sistema robótico ABB IRB 120	24
Figura 2.3 PLC ControlLogix.....	24
Figura 2.4 Topología de la red implementada	25
Figura 2.5 Modelo 3D del robot en MATLAB	27
Figura 2.6 Ubicación de ejes mediante algoritmo D-H.....	28
Figura 2.7 Modelo del sistema Robótico en Simulink	30
Figura 2.8 Subsistema GUI – Teach Pendant	31
Figura 2.9 Subsistema del control de ángulos y pulsos de movimiento.....	32
Figura 2.10 Subsistema de cálculo de cinemática directa	33
Figura 2.11 Subsistema OPC	36
Figura 2.12 Configuración del Host.....	37
Figura 2.13 Habilitación de servo-variadores mediante variable asociada al OPC....	38
Figura 2.14 Deshabilitación de servo-variadores mediante variable asociada al OPC	38
Figura 2.15 Referenciado de ejes de servomotores	39
Figura 2.16 Instrucción para el movimiento de la primera articulación	40
Figura 3.1 Área de servomotores.....	41
Figura 3.2 Indicador para eje de servomotor	42
Figura 3.3 GUI - Teach Pendant.....	42

Figura 3.4 Simulación 3D del robot.....	44
Figura 3.5 Cinemática Directa de robot ABB IRB 120	45
Figura 3.6 Simulación del robot con valores angulares en cero	45
Figura 3.7 Validación de articulaciones	47
Figura 3.8 Estado de las variables en el servidor OPC.....	48
Figura 3.9. Error medido de cada articulación	49
Figura 3.10 Posición angular para cada articulación	50
Figura 3.11 Movimiento de articulaciones en simulación 3D	52
Figura 3.12 Movimiento de servomotores.....	52
Figura 3.13 Gráfica de la posición del elemento terminal	53
Figura 3.14 Manual de usuario del sistema	54

ÍNDICE DE TABLAS

Tabla 2.1 Valores Denavit-Hartenberg del robot IRB 120 de ABB.....	28
Tabla 3.1 Límites para articulaciones del robot ABB IRB 120	47
Tabla 3.2 Análisis de costos	50
Tabla 4.1 Medición de posiciones angulares para articulaciones	55
Tabla 4.2. Subsistemas del sistema robótico.....	62

CAPÍTULO 1

1. INTRODUCCIÓN

En la actualidad se ha tornado imprescindible el uso de robots en los procesos industriales, debido a su gran eficiencia, precisión y flexibilidad; además, las simulaciones del robot otorgan al usuario una mejor perspectiva de la ejecución de tareas en el área de trabajo.

El presente proyecto indaga en la comunicación OPC entre MATLAB y Studio 5000 para la simulación de un robot de 6 grados de libertad y la visualización de las articulaciones del robot, mediante el movimiento de los ejes de los servomotores.

La característica principal de la comunicación OPC es el intercambio de datos a través de una interfaz que simplifica la extensión de drivers de entradas y/o salidas, facilitando la integración de hardware y software de distintos proveedores, que ofrezca una solución efectiva para la comunicación remota entre la PC y los dispositivos de planta.

Para analizar la problemática es necesario establecer las causas. Una de ellas, es la conexión entre MATLAB y Studio 5000 para realizar la simulación en tres dimensiones del robot mediante una interfaz gráfica y establecer un control de los servomotores respectivamente. Para ello, se establece comunicación mediante KEPServerEX (OPC), lo que permite el intercambio de datos entre las dos plataformas, con el fin de tener una arquitectura cliente/servidor. A partir de la interfaz gráfica (GUI), en MATLAB se adquiere los parámetros de cada articulación para el cálculo de la cinemática directa, dichos parámetros son enviados al servidor, a través de OPC Toolbox de Simulink. Una vez obtenidos los parámetros de cada articulación, Studio 5000 los recepta para la configuración de parámetros de los servo-variadores que controlan los ejes de cada servomotor.

Es por lo expuesto anteriormente que, se pretende desarrollar las simulaciones de las articulaciones del robot en los servomotores del laboratorio de Automatización Industrial de la Facultad de Ingeniería en Electricidad y Computación.

1.1 Descripción del problema

En la industria algunos de los procesos son repetitivos y otros requieren una mayor precisión para la ejecución de tareas y que genere mayor eficiencia en la línea de producción. Procesos tales como: montaje y soldadura, requieren de precisión y transferencia de objetos de un punto a otro, aquí nace la necesidad de una máquina multifuncional programable que pueda cumplir de manera eficiente, precisa y flexible alguno de los procesos industriales.

En el laboratorio de Automatización Industrial de la Facultad de Ingeniería Eléctrica y Computación (FIEC), no se cuenta con un brazo robótico para la realización de nuevas prácticas que permitan al estudiante visualizar el funcionamiento y flexibilidad para la programación de un robot. Dado que el montaje y pruebas físicas requieren de un alto presupuesto, mucho tiempo e implica riesgos en daño de otros dispositivos, las simulaciones suelen ser de gran ayuda, en donde el robot pueda ser reemplazado por un modelo en software e interactúe con un sistema de control real.

Al trabajar con un robot de seis grados de libertad, su modelamiento requiere de un software de cálculo robusto para representar la cinemática del robot. Uno de los inconvenientes es establecer una comunicación entre estos dos distintos ambientes, el modelado en software mediante una interfaz gráfica, y un software que realice las diferentes configuraciones de los servo-variadores para el control de servomotores.

1.2 Objetivos

1.2.1 Objetivo General

Desarrollar un sistema robótico de 6 grados de libertad, simulado en MATLAB, que pueda controlar los servo-variadores del entorno de Studio 5000, mediante comunicación a través de servidor OPC, para impulsar nuevas prácticas del laboratorio de automatización.

1.2.2 Objetivos Específicos

- Establecer una interfaz gráfica para la simulación de un robot de 6 GDL en MATLAB.
- Determinar la cinemática del robot para definir la posición de su articulación final.
- Establecer comunicación OPC entre MATLAB y Studio 5000, para interacción entre el robot simulado y los servo-variadores.

1.3 Marco Teórico

1.3.1 Robot Industrial

Es un manipulador reprogramable y multifuncional con tres o más ejes, capaz de posicionar y orientar objetos para la ejecución de tareas en las distintas etapas de la producción industrial. La Figura 1.1 muestra un robot industrial manipulador de la marca ABB.



Figura 1.1.1 Robot industrial ABB Series IRB 120

1.3.2 Robot Angular o Antropomórfico

Configuración en donde todas las articulaciones del robot son de tipo rotacional. Se lo denomina antropomórfico debido a las similitudes que este posee con un brazo humano, como se muestra en la Figura 1.2. Se caracterizan por tener un gran espacio de trabajo y flexibilidad (varios grados de libertad). Posee dificultad

en su control cinemático y dinámico, esta complejidad aumenta a medida que se incrementa sus grados de libertad.

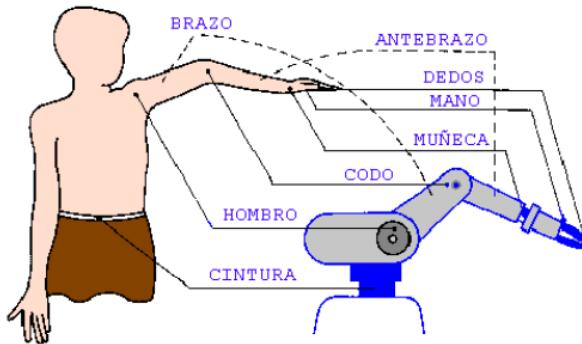


Figura 1.2. Robot antropomórfico

1.3.3 Cinemática directa de un robot

Permite determinar la posición y orientación del elemento terminal del robot para un conjunto dado de valores de articulaciones y parámetros geométricos propios del robot, con respecto a un sistema de coordenadas tomado como referencia, como se muestra en la Figura 1.3.



Figura 1.3 Cinemática directa

1.3.4 Interfaz Gráfica de Usuario

Es el programa o medio que gestiona la interacción con el usuario en base a una relación visual como iconos, menús o un puntero. La interfaz gráfica de usuario,

también llamada GUI (Graphical User Interface, por sus siglas en inglés), consiste en un programa el cual sirve como medio para tener interacción entre el usuario y la máquina. Un software el cual nos muestra de manera visual todas las acciones del sistema, para que el usuario tenga un monitoreo sin la necesidad de disponer de amplios conocimientos informáticos, en donde la principal finalidad es simplificar y que la interacción con un dispositivo sea mucho más amigable, como lo muestra en la Figura 1.4.

Permite ofrecer una solución personalizada, brindando una comunicación sencilla y permitiendo el aprovechamiento de la tecnología al alcance de cualquiera.

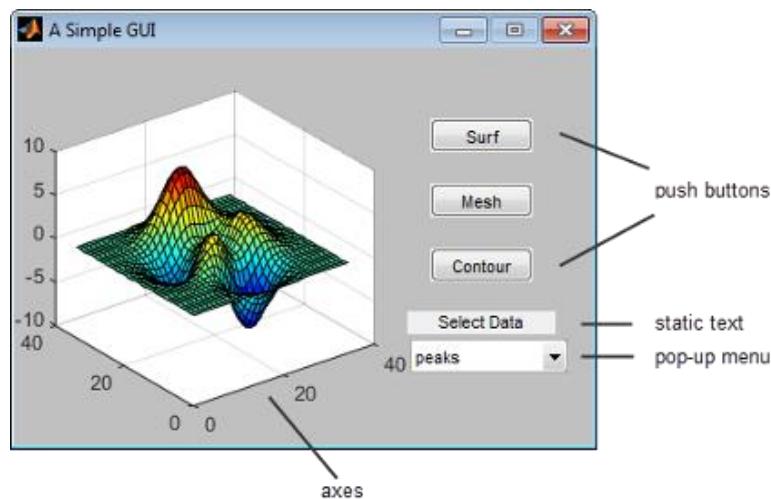


Figura 1.4 Interfaz gráfica de usuario (GUI)

1.3.5 Consola de programación

La consola de programación, también conocida como Teach Pendant en el área de robótica industrial, es una interfaz HMI diseñada para programar y verificar los programas a ejecutarse por parte del robot industrial. Se utiliza para el control de movimiento de robots que realizan tareas exigentes, repetitivas o peligrosas, que permiten a un operador manipular de forma segura las acciones realizadas por el robot. Este dispositivo de mano puede ser cableado o inalámbrico. En la Figura 1.5 se muestra un teach pendant de la marca ABB.



Figura 1.5 Teach pendant de la marca ABB

1.3.6 PLC

El PLC (Controlador Lógico Programable, por sus siglas en inglés), es un dispositivo que se programa para que realice acciones de control automático, principalmente en procesos de la industria, debido a que controlan el tiempo de ejecución y la secuencia de acciones. El PLC es un aparato electrónico que se opera digitalmente, con una memoria programable para almacenar instrucciones para funciones específicas, tales como lógica, secuencias, registro y control de tiempos, conteo y operaciones aritméticas para el control, por medio de módulos analógicos y digitales.

El sistema PLC puede contener una serie de procesos y sistemas, incluso puede contar con conexión a Internet por lo que permite realizar un monitoreo de forma remota, tanto en su operación como en los resultados, accediendo incluso por medio de una computadora común.

Existen dos tipos de PLC, el compacto y el modular. El PLC compacto (Figura 1.6), no requiere de un gran número de I/O, tiene incorporado la fuente de alimentación, el CPU, sus entradas, salidas y puertos de comunicación. Por otro lado, el PLC modular posee una estructura ampliamente configurable, consta de varios elementos como: CPU, fuente de alimentación, módulos de entradas y salidas, y módulos de expansión de hardware.

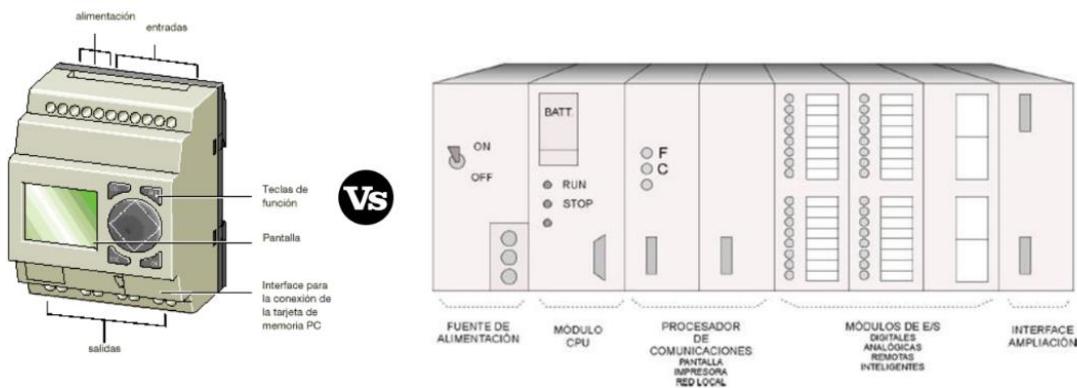


Figura 1.6 Partes de un PLC compacto y modular

1.3.7 OPC

Por sus siglas OPC (OLE for Process Control), es un estándar de comunicación para el intercambio de datos que simplifica la extensión de drivers (dispositivos de planta y banco de datos), mediante una interfaz común. Ofrece una solución flexible al problema de drivers por fabricante, requiriendo únicamente el desarrollo de una interfaz para cada fabricante de Hardware y Software. Esta comunicación se realiza mediante arquitectura cliente-servidor, en donde el servidor OPC es la fuente de datos y la aplicación OPC pueda acceder al servidor para leer y/o escribir variables.

1.3.8 Servomotor

Es un dispositivo electromecánico de alto nivel de precisión, que cuenta con un sistema de retroalimentación, el cual le indica al servo-drive la posición actual del eje para corregir su posición, si esta fuese incorrecta. Tiene la capacidad de ubicarse en una posición determinada dentro de un rango de operación.

Consiste de un motor eléctrico, engranajes y una tarjeta de control, como se muestra en la Figura 1.7. El motor es el encargado de darle movilidad al servo, al invertir sus terminales de potencial es posible cambiar su sentido de giro. Los engranajes reductores son aquellos que reducen la alta velocidad de giro del motor para incrementar su capacidad de torque. Finalmente, la tarjeta electrónica es la encargada de establecer el control de posición por retroalimentación.

Su funcionamiento es manejado por la modulación de ancho de pulso, en donde la duración de cada pulso es interpretada como una señal de posicionamiento del motor. La duración del pulso cambia según el fabricante. El control de velocidad se logra modificando la frecuencia con la que se entregan los pulsos. Si la frecuencia de generación de pulsos es mayor, la carga logrará la posición requerida rápidamente; caso contrario, a frecuencia menor, le tomará más tiempo en alcanzar la posición requerida.

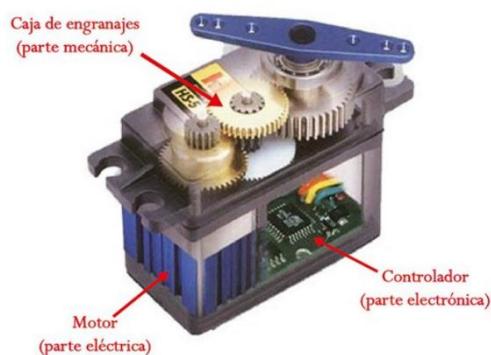


Figura 1.7 Partes de un servomotor

1.3.9 Servo-variador

Es un amplificador electrónico especial diseñado para el control de servomotores. Este dispositivo controla la señal de retroalimentación, que detecta posibles errores para el ajuste continuo a la desviación del comportamiento esperado. El lazo cerrado de control maneja la posición, velocidad y torque del eje del motor. La Figura 1.8 muestra un servo-variador comúnmente utilizado para aplicaciones de ensamblaje, envasado y manejo de materiales.



Figura 1.8 Servo-variador Kinetix 350 de la marca Allen Bradley

1.3.10 Studio 5000 Logix Designer

Es una aplicación reservada para la configuración, programación y mantenimiento de toda la familia de productos de controladores y dispositivos Allen-Bradley. Posee un ambiente de diseño único para los ámbitos de seguridad, movimiento, procesos y variadores. Además, de contar con un desarrollo colaborativo, simplificando la programación con la ayuda de funciones y editores para el incremento de la productividad. También, cuenta con niveles de seguridad basadas en licencias para la protección del código.

1.3.11 MATLAB

Es un software matemático con un entorno de desarrollo integrado (IDE, por sus siglas en inglés), que posee su propio lenguaje de programación, el lenguaje M, que puede ejecutarse en el entorno interactivo, como a través de un script. Destaca por la manipulación de matrices, implementación de algoritmos, creación de GUI y la representación de modelos, datos y funciones.

1.3.12 Simulink

Es un entorno de programación gráfica de alto nivel que funciona en conjunto con el software programación de MATLAB. Es una herramienta de mucha utilidad para la simulación de modelos y sistemas. Reduce costos en prototipos mediante simulación hardware in-the-loop (HIL, por sus siglas en inglés), como lo muestra en la Figura 1.9, evitando pruebas peligrosas, costosas o lentas.

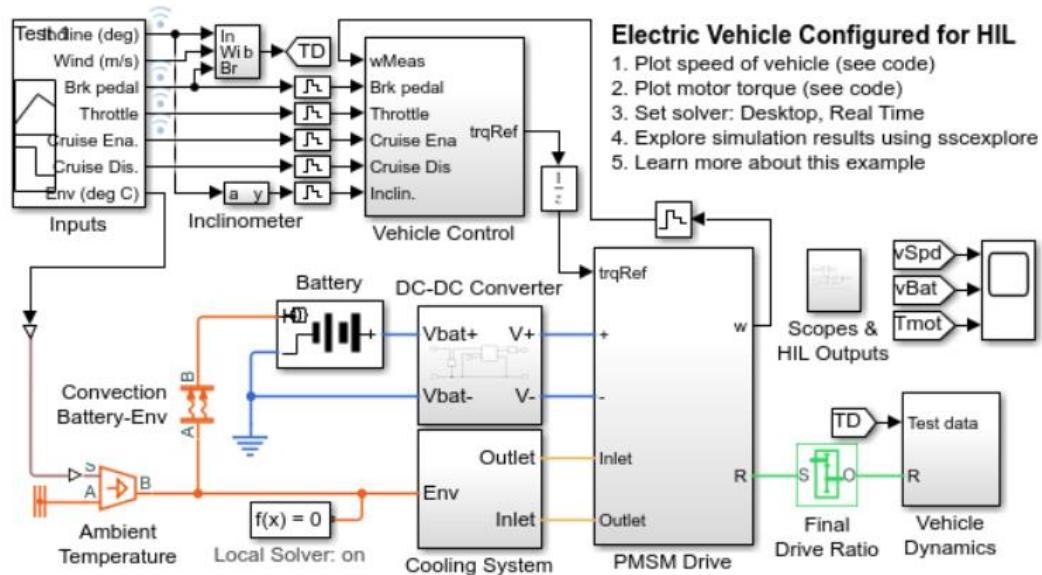


Figura 1.9 Modelo de un vehículo eléctrico en el entorno Simulink

1.3.13 KEPServerEX

Es una plataforma de comunicación que provee una única fuente de datos, que aprovecha OPC y los protocolos de comunicación centrados en la tecnología de la información (TI, por sus siglas en inglés), para las aplicaciones de la automatización industrial. Su diseño permite conectarse, administrar, monitorear y controlar múltiples dispositivos y aplicaciones de software mediante una interfaz de usuario intuitiva. Esta plataforma fue desarrollada para desempeñar el rendimiento, facilidad y confiabilidad de los clientes.

1.3.14 Control de movimiento

Es un sub-campo de la automatización, que comprende los sistemas implicados en el movimiento de partes de máquinas de una manera controlada. Consiste en un controlador de movimiento, que genera puntos de ajuste y cierran un bucle de realimentación de posición o velocidad; un amplificador de energía, que transforma la señal de control en energía para inyectarla al actuador; y motores, para el movimiento de la salida del sistema. El control de movimiento puede realizarse en lazo abierto o cerrado.

CAPÍTUL Para la simulación del sistema robótico se utilizaron los softwares: MATLAB Simulink, donde se modeló el robot IRB 120 de ABB (se analizará en epígrafe posterior), junto con una GUI que hace de Teach Pendant (consola de programación), para el control y simulación 3D del robot; Studio 5000, para el control de los servomotores mediante las instrucciones de Motion Control de los servo-variadores, que simulan las articulaciones del robot; y KEPServerEX OPC, para la comunicación de la GUI y los servo-variadores, que se encuentran en dos ambientes distintos. Esta comunicación trabaja con la arquitectura Cliente/Servidor para el intercambio de datos. El proceso para el desarrollo de esta aplicación se muestra en la Figura 2.1.

O 2

2. METODOLOGÍA

Para la simulación del sistema robótico se utilizaron los softwares: MATLAB Simulink, donde se modeló el robot IRB 120 de ABB (se analizará en epígrafe posterior), junto con una GUI que hace de Teach Pendant (consola de programación), para el control y simulación 3D del robot; Studio 5000, para el control de los servomotores mediante las instrucciones de Motion Control de los servo-variadores, que simulan las articulaciones del robot; y KEPServerEX OPC, para la comunicación de la GUI y los servo-variadores, que se encuentran en dos ambientes distintos. Esta comunicación trabaja con la arquitectura Cliente/Servidor para el intercambio de datos. El proceso para el desarrollo de esta aplicación se muestra en la Figura 2.1.



Figura 2.1 Diagrama de la aplicación de simulación de un robot de 6 GDL orientado a servomotores

2.1. Descripción del robot

Es un robot manipulador antropomórfico de configuración en serie, el más pequeño de la corporación suiza ABB. Posee seis grados de libertad (GDL), con un peso de 25 kg, pudiendo soportar una carga de 3 kg, y con un alcance de 580 mm. Destaca por su fiabilidad y flexibilidad para generar una gran producción a cambio de una baja inversión en comparación con modelos más grandes.

Utiliza el controlador compacto IRC5, haciendo uso de su tecnología de control de movimiento que potencia el rendimiento del robot. Su funcionalidad se ve mejorada en términos de rapidez, precisión, programación y sincronización con otros dispositivos.

Todas las tareas de manipulación y programación del robot son ejecutadas por el FlexPendant (consola de programación), en conjunto con el software RobotStudio, que maneja los parámetros de configuración, gestiona los programas y realiza el acceso remoto de robot.

Se seleccionó este robot debido a su diseño compacto, preferible para ámbitos académicos. La Figura 2.2 muestra el sistema robótico IRB 120 de ABB.



Figura 2.2 Elementos del sistema robótico ABB IRB 120

2.2 Controlador

Se eligió el ControlLogix L73 (ver Figura 2.3), para el control de movimiento, debido a que utiliza procesamiento distribuido y sincronizado, y brinda una solución de movimiento integrada. Además, tiene compatibilidad con 31 instrucciones de control de movimiento que pueden ser programadas mediante el lenguaje de lógica de escalera. Permite ejecutar tareas de control de movimiento complejas, multi-ejes y sincronizadas con la arquitectura integrada.



Figura 2.3 PLC ControlLogix

2.3 Topología de red

Los servo-variadores utilizan el protocolo industrial común (CIP, por sus siglas en inglés), que proporciona la funcionalidad de control de movimiento de bucle

cerrado determinístico, en tiempo real, a través de Ethernet. CIP utiliza las capas de transporte implementadas en aplicaciones Ethernet, esto permite que los dispositivos se conecten fácilmente utilizando switches.

Los dispositivos de la red se conectaron en topología anillo, como se muestra la Figura 2.4. Los servo-variadores Kinetix 5500 incorporan conectividad de puerto doble al igual que el ControlLogix L73, facilitando la conexión sin la inclusión de módulos para la expansión de puertos. Además, posee una arquitectura sólida, ya que al ser desconectado uno de los dispositivos, el resto seguirá comunicando pasando a formar un anillo abierto.

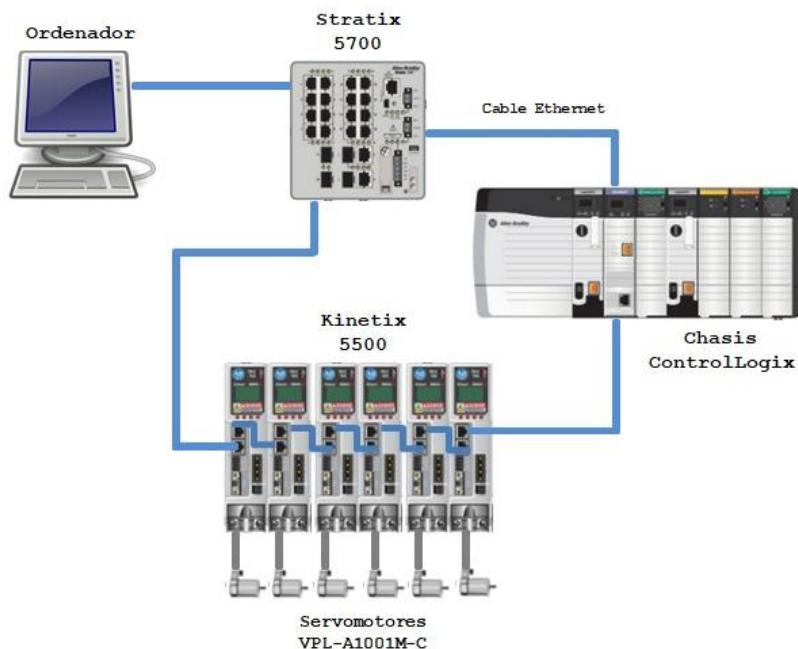


Figura 2.4 Topología de la red implementada

2.4. Modelado del robot

2.4.1 Representación de Denavit-Hartenberg (D-H)

Se optó por este algoritmo debido a que, define la cinemática del robot en función de los ángulos de sus articulaciones, mediante la obtención sistemática de la matriz de transformación homogénea T. Este consiste en un método matricial que establece la posición que debe tomar cada sistema de coordenadas $\{S_i\}$ junto a cada eslabón i de un enlace articulado. Para

sistematizar la representación de elementos contiguos se realiza cuatro transformaciones básicas que dependen únicamente de las características geométricas del eslabón. Estas cuatro transformaciones se resumen en una sucesión de rotaciones y traslaciones que relacionan el sistema de referencia del elemento $i-1$ con el sistema del elemento i . Las transformaciones son:

- α_i : Define la rotación alrededor del eje x_i .
- θ_i : Define la rotación alrededor del eje z_{i-1} .
- d_i : Define la traslación a lo largo z_{i-1} . Vector $(0,0,d_i)$.
- a_i : Define la traslación a lo largo de x_i . Vector $(a_i,0,0)$.

La matriz de transformación parcial A_i^{i-1} se define por:

$$A_i^{i-1} = \text{Rotz}(\theta_i) T(0,0,d_i) T(a_i, 0, 0) \text{Rotx}(\alpha_i)$$

Y realizando el producto entre matrices se obtuvo:

$$A_i^{i-1} = \begin{bmatrix} C\theta_i & -S\theta_i C \alpha_i & S\theta_i S \alpha_i & a_i C\theta_i \\ S\theta_i & C\theta_i C \alpha_i & -C\theta_i S \alpha_i & a_i S\theta_i \\ 0 & S \alpha_i & C \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

La Figura 2.5 muestra los eslabones y articulaciones del robot en un modelo 3D en MATLAB, para el cálculo de los parámetros D-H.

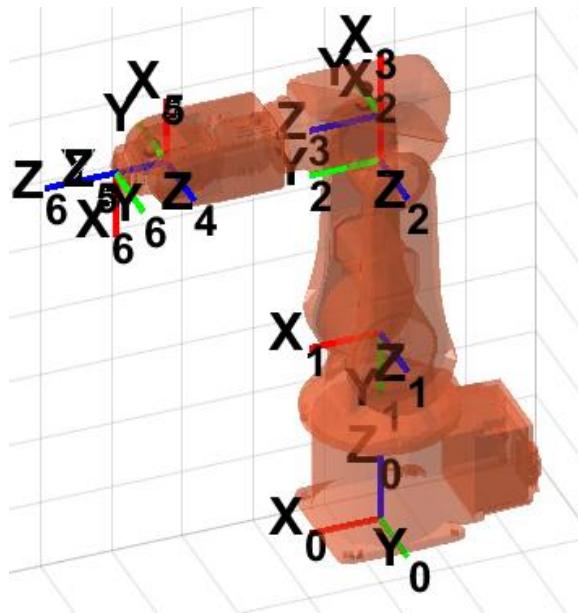


Figura 2.5 Modelo 3D del robot en MATLAB

Se utilizó el siguiente algoritmo para la ubicación de los ejes de cada eslabón:

1. Se numera los eslabones y articulaciones iniciando con $n = 1$ y terminando con n (último eslabón y grado de libertad respectivamente).
2. Se localiza el eje de cada articulación. Si el eje es rotativo, éste será su propio eje de giro. Si es prismático, el eje será a lo largo del cual se produce el desplazamiento.
3. Posicionar el eje z_i sobre el eje de la articulación $i + 1$, donde i toma valores de 0 a $n - 1$.
4. Posicionar los ejes x_0 e y_0 de modo que se muevan en el sentido de las manecillas del reloj respecto a z_0 , formando el sistema de la base $\{S_0\}$.
5. Posicionar el sistema $\{S_i\}$ solidario al eslabón i . Para $i = 1$ a $n - 1$ en la intersección del eje z_i con la línea normal común a z_{i-1} y z_i .
6. Posicionar x_i en la línea normal común a z_{i-1} y z_i .
7. Posicionar y_i haciendo que forme un sistema dextrógiro con x_i y z_i .
8. Posicionar el sistema $\{S_n\}$ en el extremo del robot logrando que z_n concuerde con la dirección de z_{n-1} y x_n sea normal a z_{n-1} y z_n .

Una vez situado los ejes de cada articulación, como se muestra en la Figura 2.6, se procedió a obtener los parámetros D-H que se detallan en la Tabla 2.1.

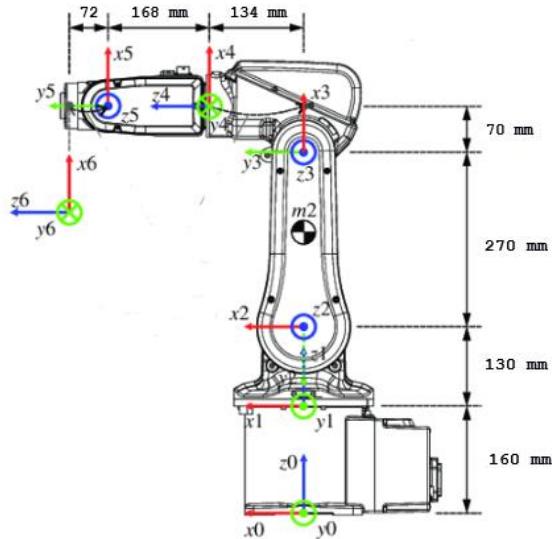


Figura 2.6 Ubicación de ejes mediante algoritmo D-H

Tabla 2.1 Valores Denavit-Hartenberg del robot IRB 120 de ABB

Eslabón	θ_i	d_i	a_i	α_i
1	θ_1	290	0	-90°
2	$\theta_2 - 90^\circ$	0	270	0°
3	θ_3	0	70	-90°
4	θ_4	302	0	90°
5	θ_5	0	0	-90°
6	$\theta_6 + 180^\circ$	72	0	0°

2.4.2 Matriz de transformación homogénea

La matriz de transformación homogénea T, permitió relacionar la posición y orientación del elemento terminal respecto a un sistema fijo de referencia y se la obtuvo mediante el producto de las matrices parciales de cada

eslabón. Por otro lado, las matrices parciales resultantes se obtuvieron reemplazando los parámetros D-H del eslabón i : α_i , θ_i , d_i , a_i , en la Tabla 2.1. El robot IRB120 posee seis grados de libertad, por tanto, se obtuvo seis matrices parciales. La matriz resultante T , ésta definida por:

$$T = A_6^0 = A_1^0 \cdot A_2^1 \cdot A_3^2 \cdot A_4^3 \cdot A_5^4 \cdot A_6^5$$

2.5 Diseño del sistema robótico en MATLAB Simulink

El sistema consistió de tres subsistemas principales (ver Figura 2.7), y son:

- GUI – Teach Pendant: Consiste en una interfaz gráfica para el control y simulación del robot.
- Sistema de control: Envía y recibe las señales de control a la GUI y los parámetros necesarios para el control de los servomotores al sistema del OPC.
- Sistema OPC: Recibe los parámetros del sistema de control y los envía al entorno de Studio 5000 para el manejo de los Servo-variadores.


Integrantes:

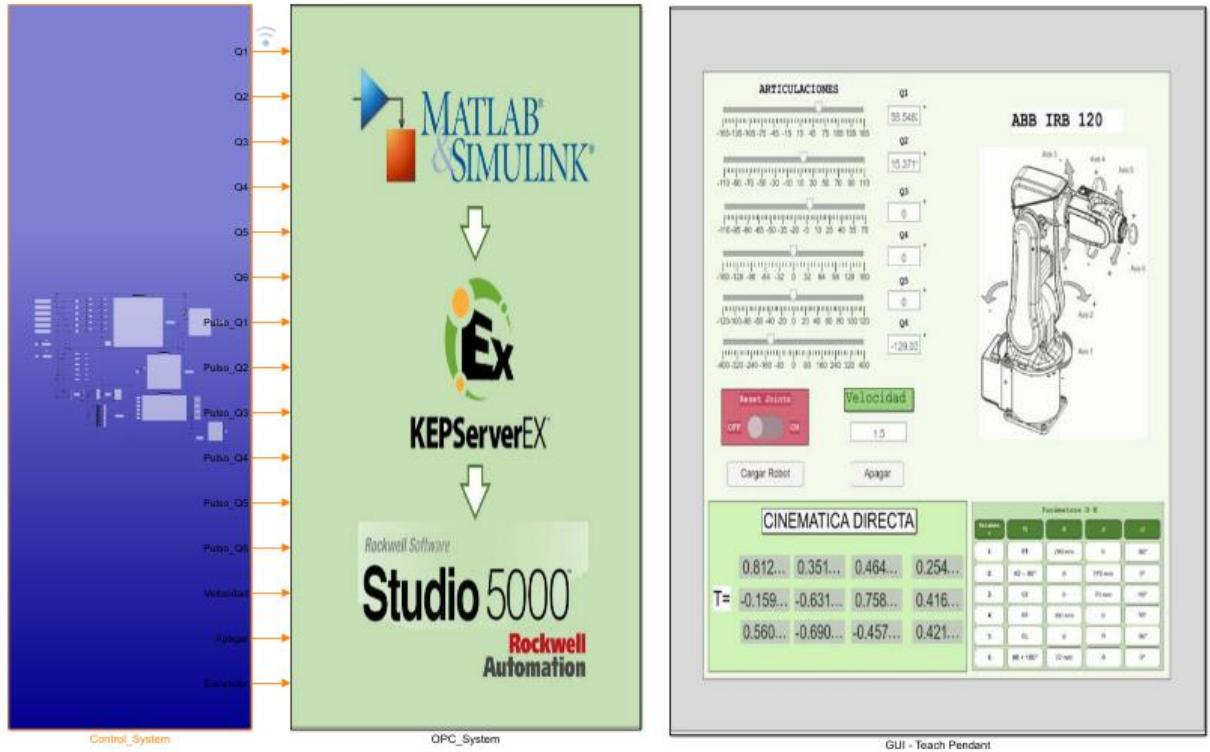
 Leonardo Romo Mora
 Kevin Villota Viteri


Figura 2.7 Modelo del sistema Robótico en Simulink

2.5.1 GUI – Teach Pendant

Este bloque se desarrolló con la finalidad de que el usuario interactúe con cada una de las articulaciones del robot y configure parámetros de posición y velocidad de los servos. La Figura 2.8 muestra el subsistema GUI – Teach Pendant.

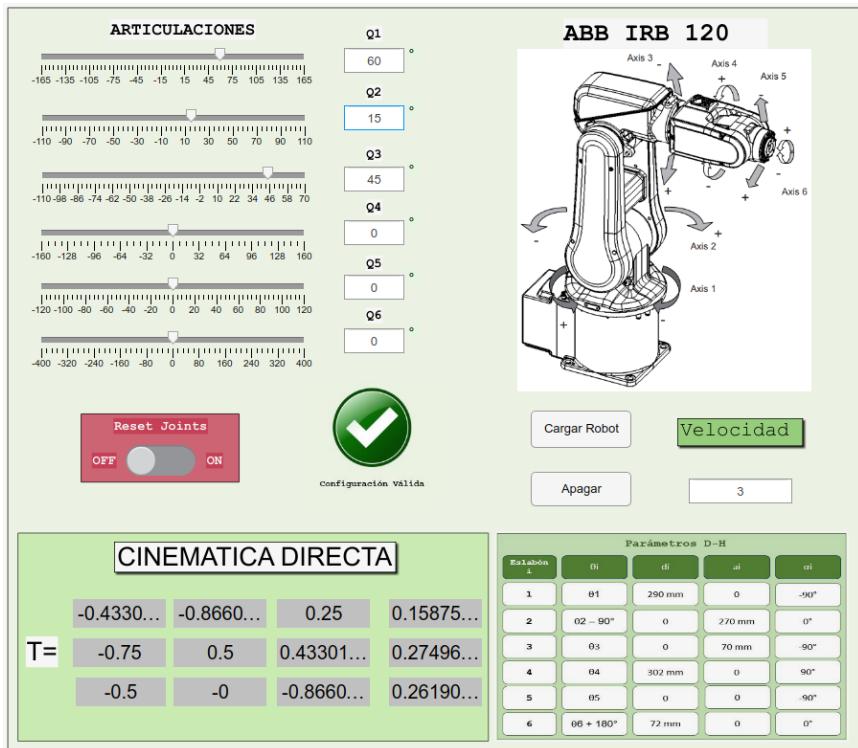


Figura 2.8 Subsistema GUI – Teach Pendant

La GUI se diseñó con bloques de la librería *Dashboard* de Simulink, que contiene controles e indicadores para la interacción con la simulación. Los bloques utilizados fueron:

- *Deslizador (Slider)*: Se utilizó para ajustar los valores de las articulaciones del robot.
- Bloque de texto (Edit): Se la añadió con la finalidad de escribir el valor de la articulación de manera precisa. Además, uno de estos permite ajustar la velocidad de todas sus articulaciones.
- *Interruptor (Slider switch)*: Utilizado para el reseteado de los valores de las articulaciones y ajustar el gráfico 3D a su posición inicial (todas sus articulaciones referenciadas a cero grados).
- *Pulsador (Push Button)*: Se implementó dos botones, ambos envían pulsos de habilitación para la carga y apagado del robot.
- *Imágenes de estados múltiples*: Este bloque muestra imágenes dependiendo del movimiento de una articulación en particular para observar la interacción con la simulación.

2.5.2 Sistema de control

Este bloque contiene el cálculo de la cinemática directa y los parámetros necesarios para el control de la GUI y la configuración de las instrucciones Motion Control tales como: posición, velocidad, encendido/apagado del campo de los servomotores y pulsos habilitadores para los bloques en la *rutina principal* de Studio 5000.

El modelamiento del robot se lo realizó mediante la programación de bloques en el Workspace de MATLAB pertenecientes a Simulink. Las Figuras 2.9 y 2.10 muestran la parte interna del sistema de control.

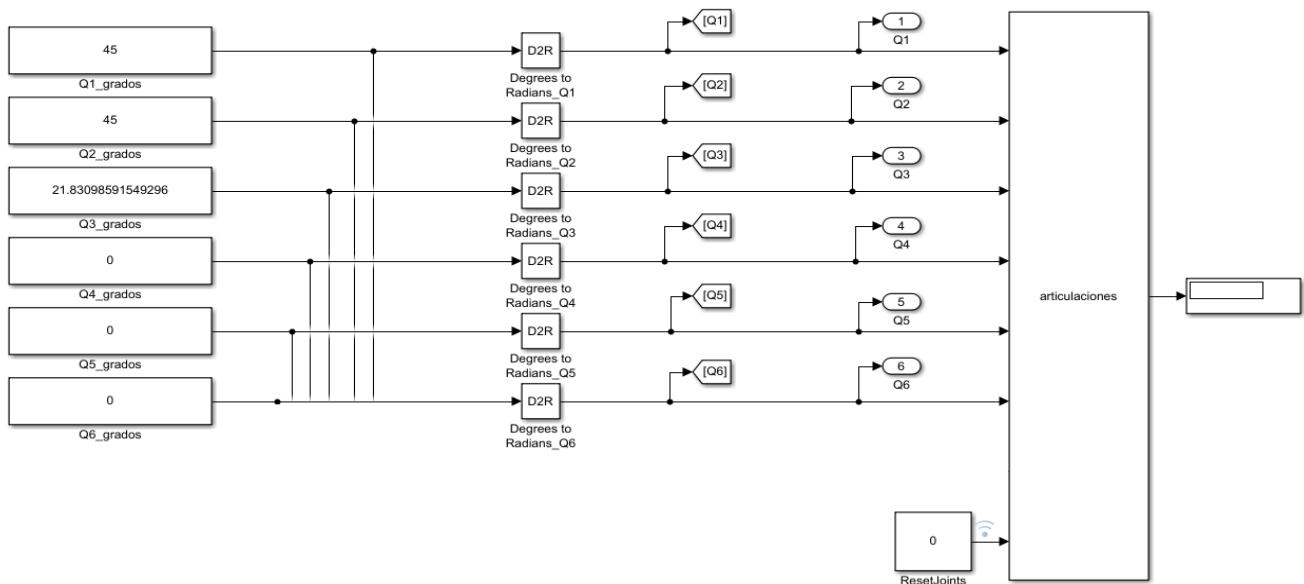


Figura 2.9 Subsistema del control de ángulos y pulsos de movimiento

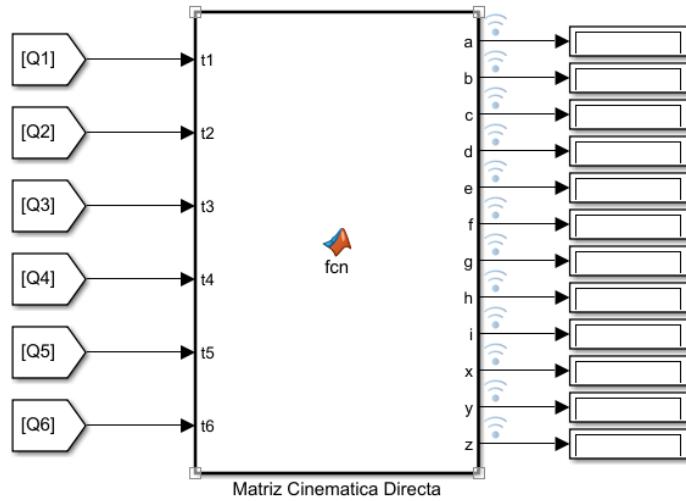


Figura 2.10 Subsistema de cálculo de cinemática directa

2.5.2.1 Bloque articulaciones

Para el control de los movimientos de la simulación 3D de cada una de las articulaciones, se usa del “Library Browser” el bloque “MATLAB function” que permite a través de un script el desarrollo de un bloque propio. El bloque articulaciones ejecuta los movimientos establecidos en la simulación 3D a través del comando “`drawrobot3d(robot)`” que al recibir los cambios en la señal de las orientaciones, la función pondrá la articulación en dicha posición.

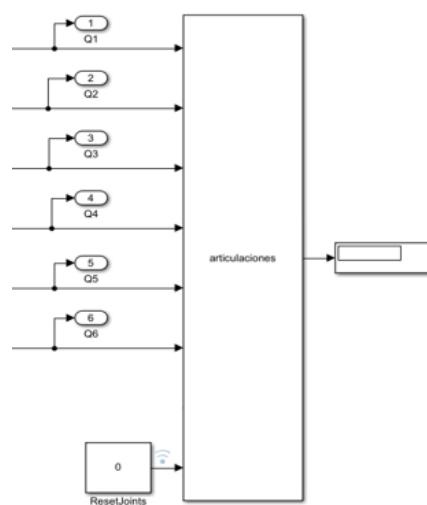


Figura 2.11 Bloque articulaciones

2.5.2.2 Bloque de cinemática directa

Recibe los valores de orientaciones y a través del método “Denavit & Hartenberg” con el sistema referencial ya establecido y el sistema matricial homogéneo en función de los ángulos en los que se encuentra cada articulación, se relacionó la orientación y posición del efecto final para ser obtenido conforme los cambios angulares.

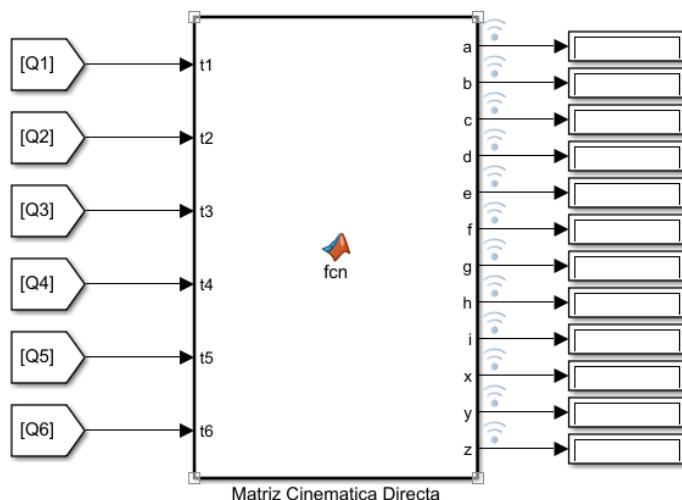


Figura 2.12 Bloque de obtención de cinemática directa

2.5.2.3 Bloque de posición del efecto final

Recibe las coordenadas del bloque de cinemática directa con sus respectivas orientaciones y realiza las gráficas de trayectoria, según los movimientos realizados por el usuario.

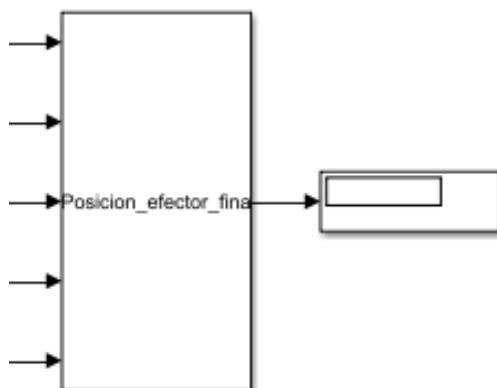


Figura 1.13 Bloque para graficación de trayectoria

2.5.2.4 Bloque de validar configuraciones

Validar configuraciones determina si el robot esta trabajando dentro del area de trabajo del IRB120 de ABB, la orientacion de cada una de las articulaciones fue parametrizada dentro de intervalos , tal que al no estar dentro de los valores angulares establecidos, manda error y no permite

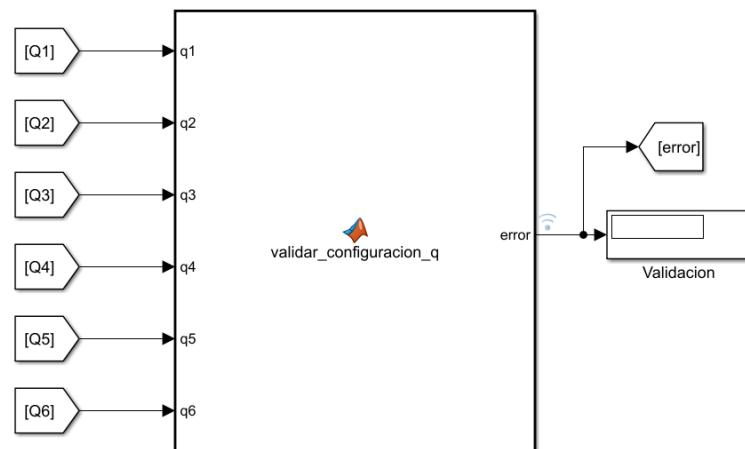


Figura 2.14 Validar configuraciones

2.5.3 Sistema OPC

Para la comunicación entre MATLAB y Studio 5000 se empleó KepServerEX 6, que realiza el intercambio de datos mediante la arquitectura Cliente/Servidor. En Simulink se utilizó el Toolbox de OPC, donde se configuró el servidor a utilizar. Además, contiene el bloque OPC Write, que realiza la escritura de variables del entorno de Simulink hacia el servidor OPC. La Figura 2.11 muestra el subsistema OPC.

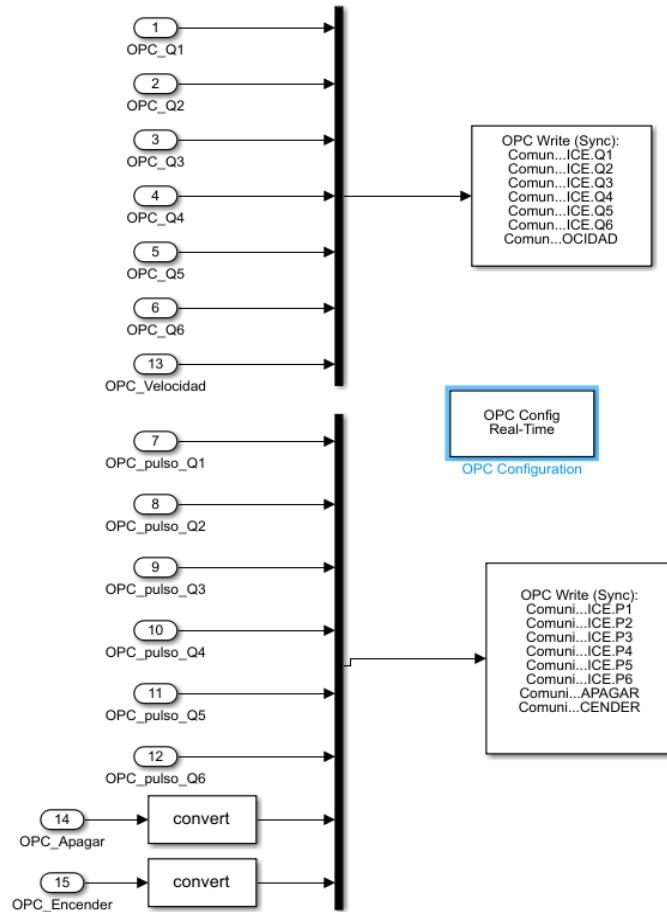


Figura 2.15 Subsistema OPC

2.6 Configuración OPC KEPServerEX

Para la comunicación OPC se utilizó KepServerEX 6, el cual hará la función de servidor, teniendo como clientes a MATLAB y Studio 5000, para la comunicación entre ellos. En el proyecto, se configuró los parámetros del host local (ver Figura 2.12), estableciendo su dirección IP y la familia del PLC, indicando la familia a la que pertenece, la dirección IP y el slot en el que se encuentra ubicado. Además, se crean los diferentes tipos de variables del servidor y estableciendo el tiempo de escaneo de la variable en los clientes.

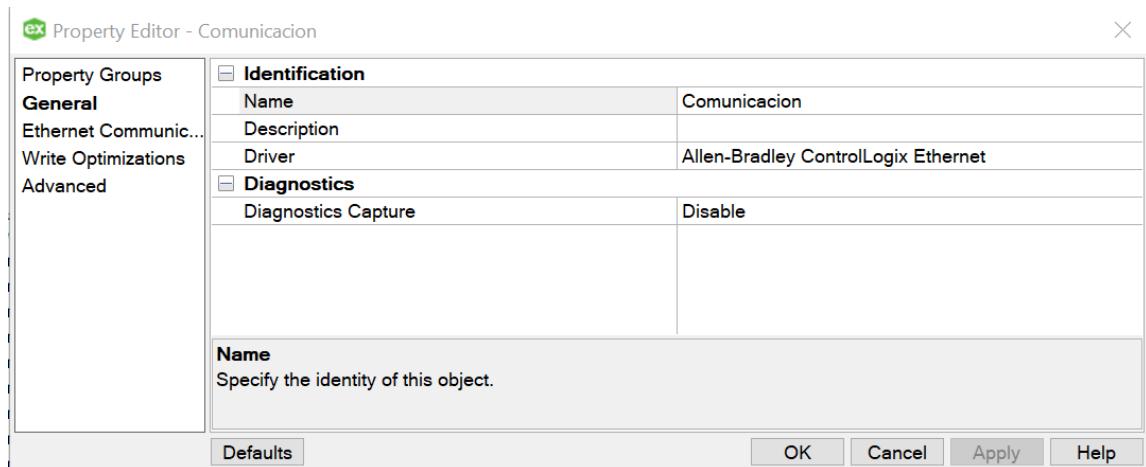


Figura 2.16 Configuración del Host

2.7 Control de movimiento de servomotores en Studio 5000

Para el control preciso de los ejes de los servomotores, se necesitó establecer configuraciones en el controlador y los servo-variadores mediante las instrucciones *Motion Control* en Studio 5000. Los valores de los parámetros de habilitación, posición y velocidad de las instrucciones fueron asignadas mediante las variables recibidas del OPC. Se seleccionó el servo-variador Kinetix 5500, compatible con movimiento integrado en la red Ethernet/IP para el control de los servomotores. Las instrucciones utilizadas fueron:

2.7.1 Motion Servo On (MSO)

Esta es una instrucción de salida que permite habilitar el servo-variador y activar el campo del eje asociado a éste. Se activa al cargar el robot en la GUI – Teach Pendant, a través de la variable *ENCENDER* asociada al OPC. La Figura 2.13 muestra la habilitación de los servo-variadores para cada uno de los ejes.

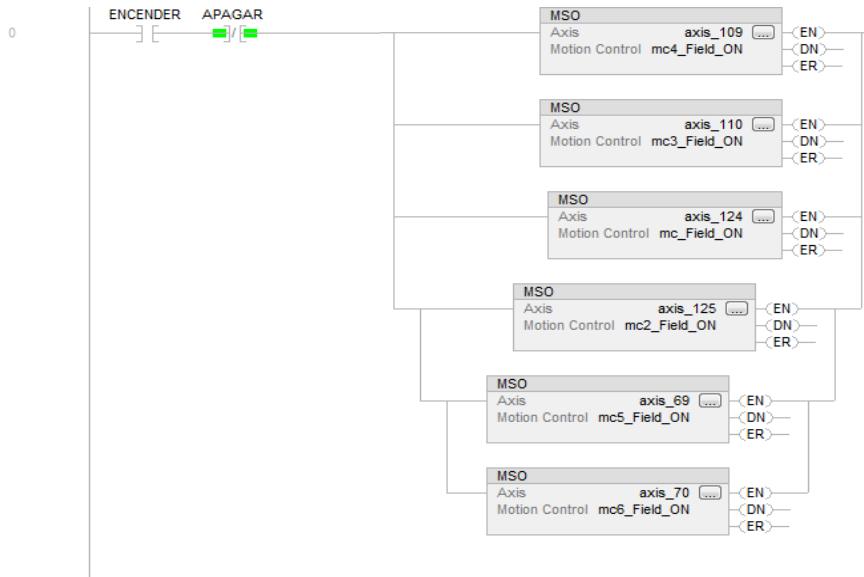


Figura 2.17 Habilitación de servo-variadores mediante variable asociada al OPC

2.7.2 Motion Servo Off (MSF)

Es una instrucción de salida, que deshabilita el servo-variador y el campo del eje asociado a éste. Al activarse la instrucción, el eje frena de forma incontrolada conservando su posición actual. La instrucción se habilita por el pulso de la variable *APAGAR* asociada al OPC (ver Figura 2.14), y controlada por el botón de apagado en la GUI.

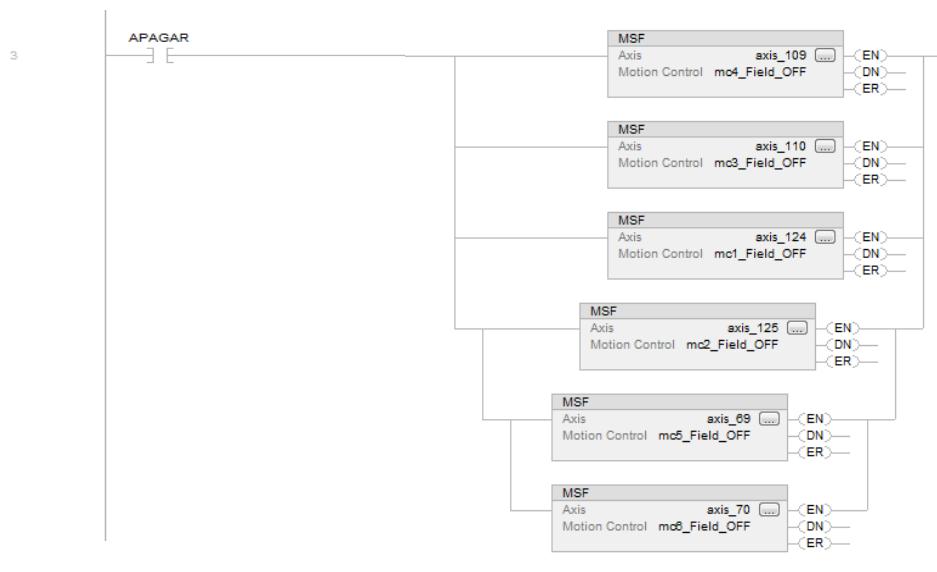


Figura 2.18 Deshabilitación de servo-variadores mediante variable asociada al OPC

2.7.1. Motion Axis Home (MAH)

Es una instrucción de salida que permite posicionar al eje a su referencia inicial de 0°. Esta instrucción se ejecuta al activarse la salida de un temporizador, comandado por la variable de encendido del robot, como se muestra en la Figura 2.15.

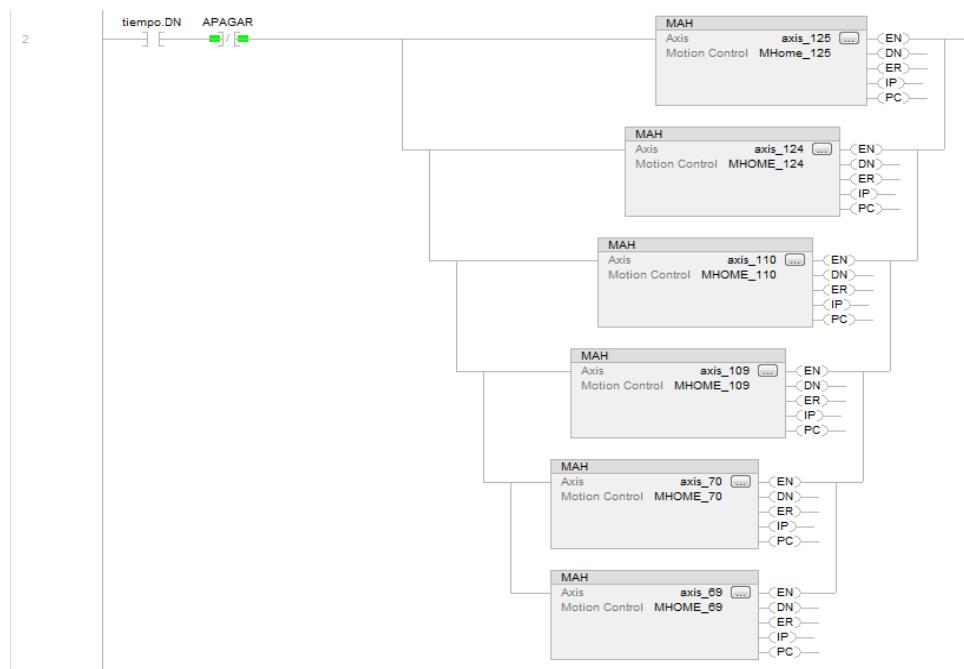


Figura 2.19 Referenciado de ejes de servomotores

2.7.3 Motion Axis Move (MAM)

Es una instrucción de salida utilizada para iniciar el movimiento de un eje. En este bloque se configuró la posición y velocidad del eje. La ejecución de esta instrucción se realizó mediante la activación de un pulso recibido por la variable asociada al OPC, proveniente del circuito de control de la GUI. La Figura 2.16 muestra la instrucción MAM para el movimiento de la primera articulación. La configuración de las instrucciones para cada eje se detalla en el anexo.

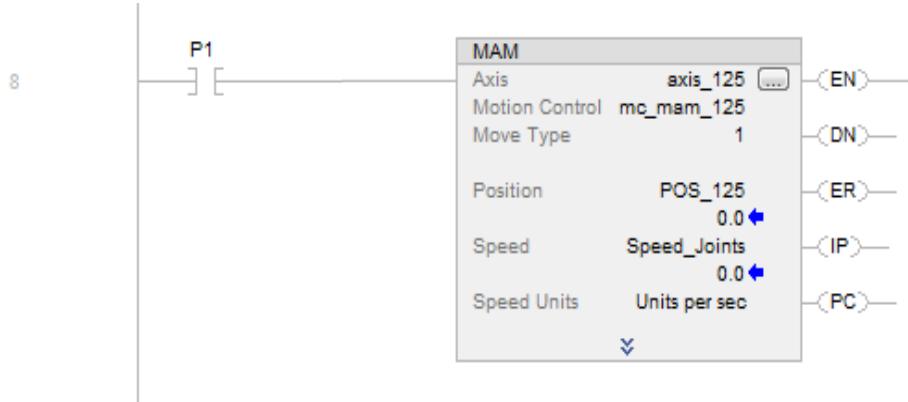


Figura 2.20 Instrucción para el movimiento de la primera articulación

Para concluir, el servidor OPC fue de gran importancia para establecer comunicación y lograr el control de posición y velocidad de los servomotores mediante la interacción entre la GUI y las instrucciones Motion Control de los servo-variadores. En el capítulo posterior, se analizará el funcionamiento de esta interacción y la visualización de los movimientos de las articulaciones en la simulación 3D.

CAPÍTULO 3

3. ANÁLISIS DE RESULTADOS

3.1 Hardware

3.1.1 Área de trabajo

La Figura 3.1 muestra el área de los servomotores dispuesta en tres tableros, que representan las seis articulaciones del brazo robótico. Los servomotores fueron controlados por los servo-variadores Kinetix 5500; estos últimos, se conectaron en anillo a un switch Stratix 5700 mediante el protocolo de comunicación EtherNet/IP.



Figura 3.1 Área de servomotores

El movimiento de los servomotores es controlado por la GUI (detallada en el epígrafe posterior), realizada en MATLAB Simulink. En la Figura 3.2 se aprecia la perilla del eje acoplado a cada servomotor que indica de manera precisa su movimiento en grados.



Figura 3.2 Indicador para eje de servomotor

3.2 Software

3.2.1 Funcionamiento de la GUI – Teach Pendant

La Figura 3.3 muestra la GUI desarrollada con la librería Dashboard de Simulink, para el control de movimiento de los servomotores y la simulación 3D del robot en MATLAB.

Eslabón /	θ_i	d_i	a_i	α_i
1	θ_1	290 mm	0	-90°
2	$\theta_2 - 90^\circ$	0	270 mm	0°
3	θ_3	0	70 mm	-90°
4	θ_4	302 mm	0	90°
5	θ_5	0	0	-90°
6	$\theta_6 + 180^\circ$	72 mm	0	0°

Figura 3.3 GUI - Teach Pendant

42

Su funcionamiento se detalla a continuación:

- Se logró controlar la posición angular de cada articulación mediante el uso de los sliders y cuadros de texto.
- Se consiguió realizar una simulación 3D para observar las distintas configuraciones que el robot puede ejercer sobre su espacio de trabajo.
- Se logra modelar la cinemática directa del robot, calculando la matriz de rotación y traslación del efecto terminal.
- Se logró mostrar la validación de las configuraciones de las articulaciones del robot mediante un indicador.
- Se consiguió establecer la velocidad de los servomotores mediante la GUI.
- Se logra visualizar el movimiento punto a punto del efecto final en la simulación.

3.2.2 Simulación 3D del robot

La Figura 3.4 muestra la simulación 3D del robot ABB IRB 120 que interactúa con la GUI en Simulink. El dibujo permitió observar el movimiento del robot a medida que sus articulaciones son manipuladas en la consola de programación, esto fue de gran ayuda para apreciar las distintas configuraciones posibles en el espacio de trabajo del robot.

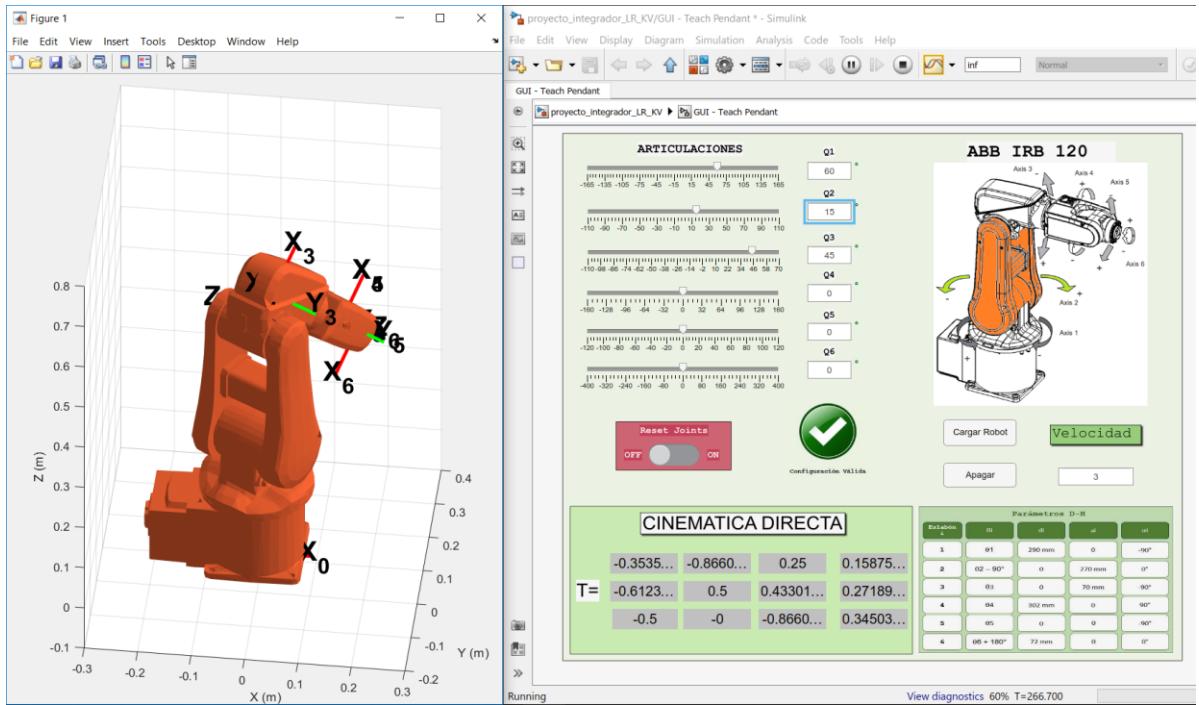


Figura 3.4 Simulación 3D del robot

3.2.3 Cinemática del robot ABB IRB 120

3.2.3.1 Cinemática directa

Se obtuvo la cinemática directa, en el cual se determinó la posición y orientación del elemento terminal del robot ABB IRB 120, con respecto al sistema de coordenadas que se tomó como referencia, conociendo los valores angulares de las articulaciones y los parámetros geométricos de los elementos del robot. Se validó el modelo matemático con los valores angulares en cero de todas las articulaciones, siendo así, las posiciones de la cinemática directa, la suma de las longitudes de los eslabones, como se muestra en la Figura 3.5 y 3.6.

CINEMATICA DIRECTA			
	-0	-0	1
T=	-0	1	0
	-1	-0	-0
			0.374
			0
			0.63

Figura 3.5 Cinemática Directa de robot ABB IRB 120

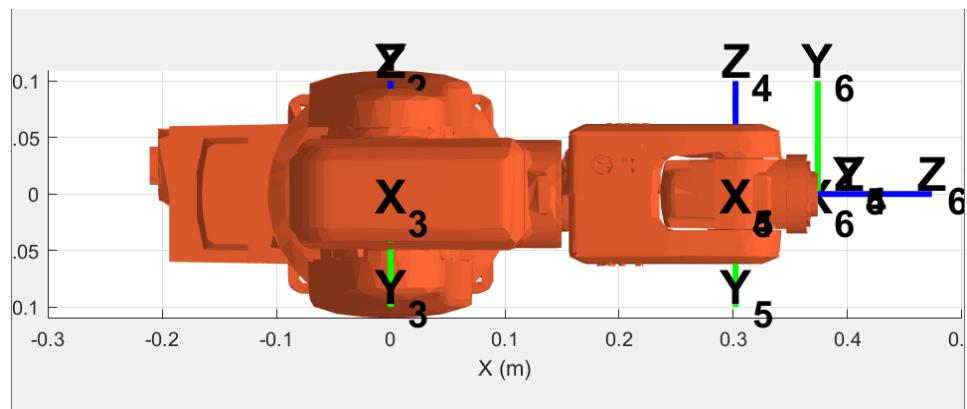


Figura 3.6 Simulación del robot con valores angulares en cero

3.2.3.2 Cinemática inversa

Se obtuvo la cinemática inversa, a partir de la posición del elemento terminal del robot en el espacio, en el cual se obtienen de forma automática al variar las posiciones, la orientación de cada una de las articulaciones y los valores de la cinemática directa. Se validó el modelo matemático con las coordenadas en cero de los tres ejes, siendo así, las orientaciones de las 6 articulaciones igual a cero Figura 3.7 y 3.8.

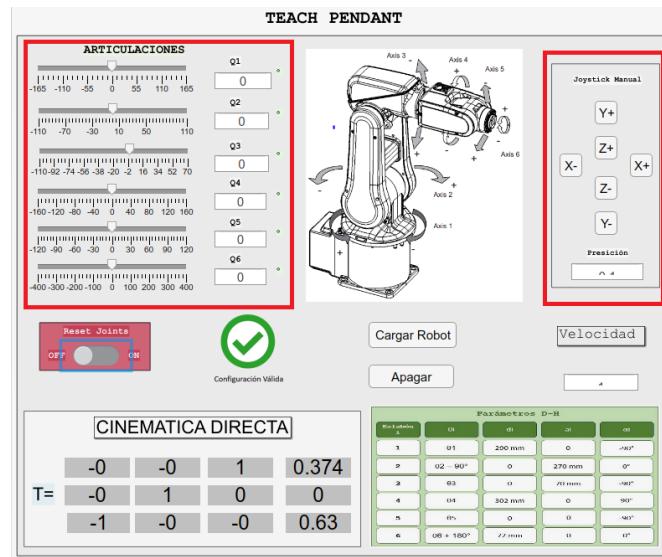


Figura 3.7 Validación de la cinemática inversa

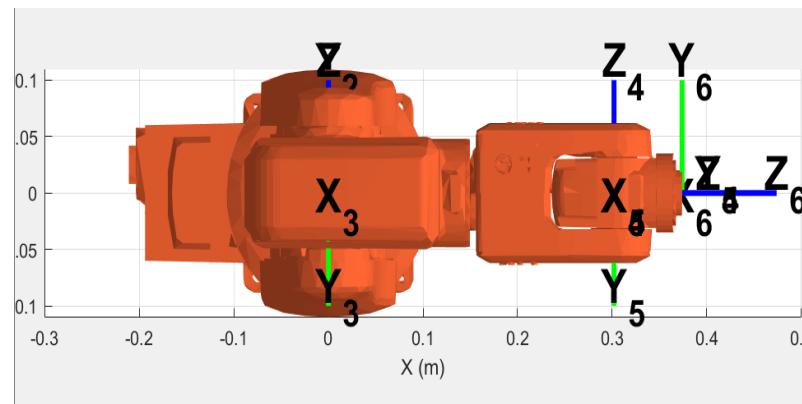


Figura 3.8 Posición del robot sin movimientos en los ejes de coordenadas

3.2.4 Validación del espacio de trabajo

Se validó las configuraciones de las articulaciones del robot, la Figura 3.7 muestra el indicador para una configuración válida en su parte izquierda y otro para una configuración inválida en su parte derecha. Los límites permisibles para cada articulación se detallan en la Tabla 3.1.

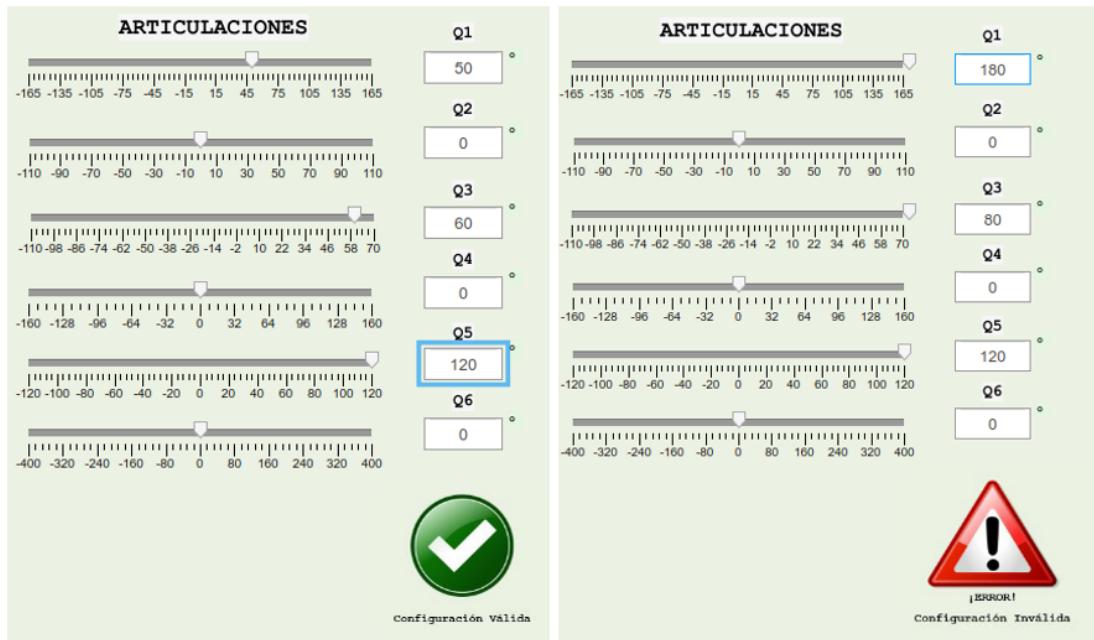


Figura 3.7 Validación de articulaciones

Tabla 3.1 Límites para articulaciones del robot ABB IRB 120

Articulación	Límite Máximo	Límite Mínimo
1	165	-165
2	110	-110
3	70	-110
4	160	-160
5	120	-120
6	400	-400

3.2.5 Comunicación OPC

En la Figura 3.8 muestra la comunicación entre MATLAB Simulink y Studio 5000 por medio del servidor KEPServerEX 6. Se realizó un escaneo de todas las variables desde el servidor con un tiempo de 100 ms, realizando escritura desde el entorno de MATLAB y lectura en Studio 5000.

Item ID	/	Data Type	Value	Timestamp
Comunicacion.DEVICE._CipConnecti...	Word	Unknown	02:41:25.098	
Comunicacion.DEVICE._CipConnecti...	Word	500	02:41:25.098	
Comunicacion.DEVICE.APAGAR	Boolean	Unknown	02:41:25.099	
Comunicacion.DEVICE.ENCENDER	Boolean	Unknown	02:41:25.099	
Comunicacion.DEVICE.P1	Boolean	Unknown	02:41:25.099	
Comunicacion.DEVICE.P2	Boolean	Unknown	02:41:25.099	
Comunicacion.DEVICE.P3	Boolean	Unknown	02:41:25.099	
Comunicacion.DEVICE.P4	Boolean	Unknown	02:41:25.099	
Comunicacion.DEVICE.P5	Boolean	Unknown	02:41:25.099	
Comunicacion.DEVICE.P6	Boolean	Unknown	02:41:25.099	
Comunicacion.DEVICE.Q1	Double	Unknown	02:41:25.099	
Comunicacion.DEVICE.Q2	Double	Unknown	02:41:25.099	
Comunicacion.DEVICE.Q3	Double	Unknown	02:41:25.099	
Comunicacion.DEVICE.Q4	Double	Unknown	02:41:25.099	
Comunicacion.DEVICE.Q5	Double	Unknown	02:41:25.099	
Comunicacion.DEVICE.Q6	Double	Unknown	02:41:25.099	
Comunicacion.DEVICE.VELOCIDAD	Double	Unknown	02:41:25.099	

Figura 3.8 Estado de las variables en el servidor OPC

3.2.6 Medición de posiciones angulares

La Figura 3.9 muestra el error de la posición angular medida para cada articulación respecto a la ingresada en la GUI. Para la siguiente configuración: $Q = [20^\circ, -60^\circ, 0^\circ, 0^\circ, 40^\circ, 0^\circ]$. La medición se la realizó con los encoders de los servo-variadores.

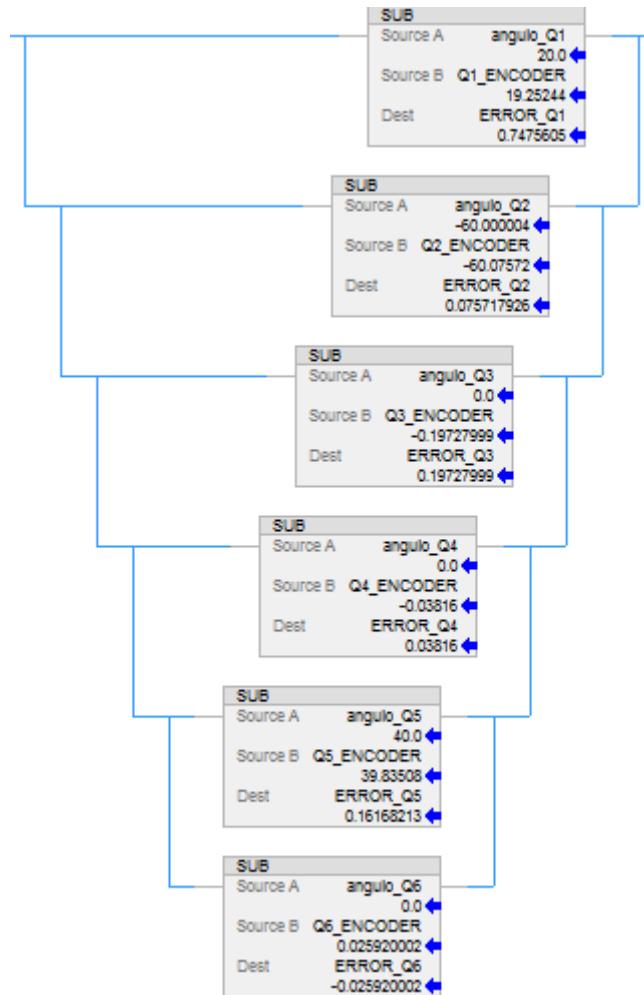


Figura 3.9. Error medido de cada articulación

La Figura 3.10 muestra una gráfica de las posiciones angulares para las articulaciones del robot. Inicialmente todas las articulaciones están ubicadas en cero grados, se aprecia el cambio para dos de sus articulaciones.

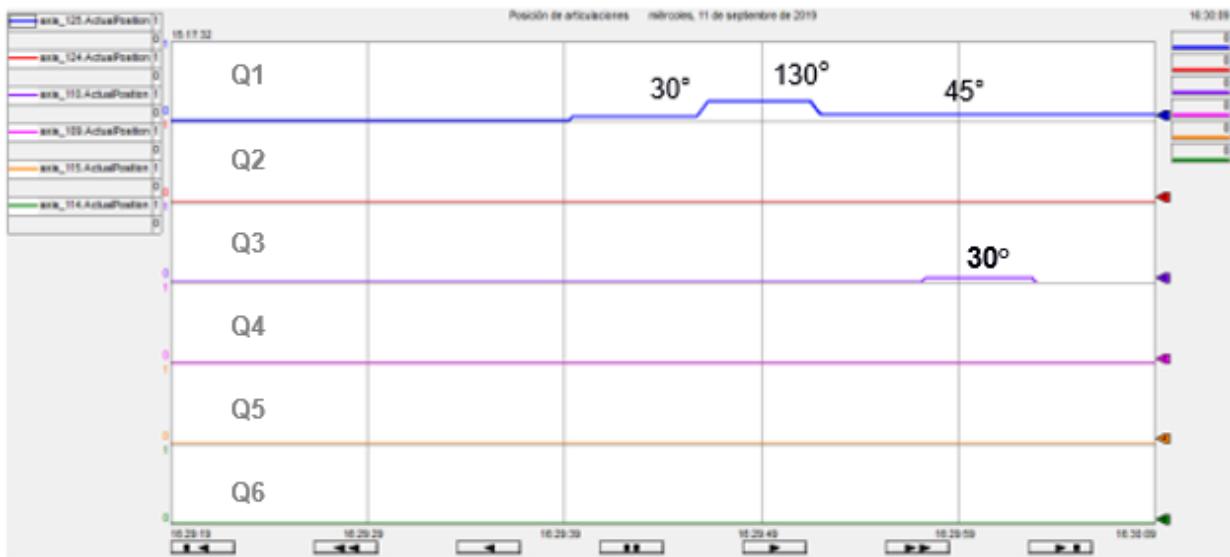


Figura 3.10 Posición angular para cada articulación

3.3 Análisis de costos

Se planteó reducir el costo que generaría adquirir un robot industrial para su uso en prácticas y/o proyectos para los estudiantes de pre-grado y post-grado en el laboratorio de automatización. Con esta finalidad, se planteó el desarrollo de una simulación de un sistema robótico con las herramientas y dispositivos disponibles en el laboratorio. El costo de los equipos utilizados se muestra en la Tabla 3.2.

Tabla 3.2 Análisis de costos

Número de catálogo	Descripción	Cantidad	Precio por unidad	Precio total
1756-A4	Chasis del ControlLogix con 4 slots	1	\$ 495.00	\$ 495.00
1756-EN2TR	Módulo EtherNet/IP 2 puertos	1	\$ 2,860.00	\$ 2,860.00
1756-L73	Controlador ControlLogix	1	\$ 10,600.00	\$ 10,600.00
1756-PA75	Fuente de alimentación	1	\$ 1,070.00	\$ 1,070.00
1783-BMS06TA	Switch Stratix 5700 de 6 puertos	1	\$ 3,070.00	\$ 3,070.00

VPL-A1001M-CJ12AA	Servo motor	6	\$ 1,170.00	\$ 7,020.00
2198-H003-ERS	Servo-variador Kinetix 5500	6	\$ 1,260.00	\$ 7,560.00
RJ45	Cable EtherNet 2.5 m	3	\$ 2.00	\$ 6.00
-	Filamento PLA 0.5 kg	1	\$ 14.00	\$ 14.00
-	Perillas para ejes de los servomotores	6	\$ 2.00	\$ 18.00
Total				\$ 32,713.00

3.4 Ejemplo de aplicación

Para la práctica se dispone realizar una prueba para la validación de configuraciones de las articulaciones del robot, donde se muestra el cálculo de la cinemática directa, la simulación 3D del robot, la trayectoria del elemento terminal y el reflejo del movimiento de las articulaciones en los servomotores. Esta prueba es de utilidad para conocer los puntos en el espacio que pueden ser alcanzados por el elemento terminal del robot.

Para esto, se estableció en la GUI la configuración:

$$Q = [60^\circ 0^\circ 0^\circ 0^\circ 0^\circ 0^\circ]$$

Las Figuras 3.11, 3.12 y 3.13, muestran que para esta configuración los ángulos son válidos, y por tanto es capaz de generar una trayectoria sobre su espacio de trabajo, así como el cálculo de su cinemática.

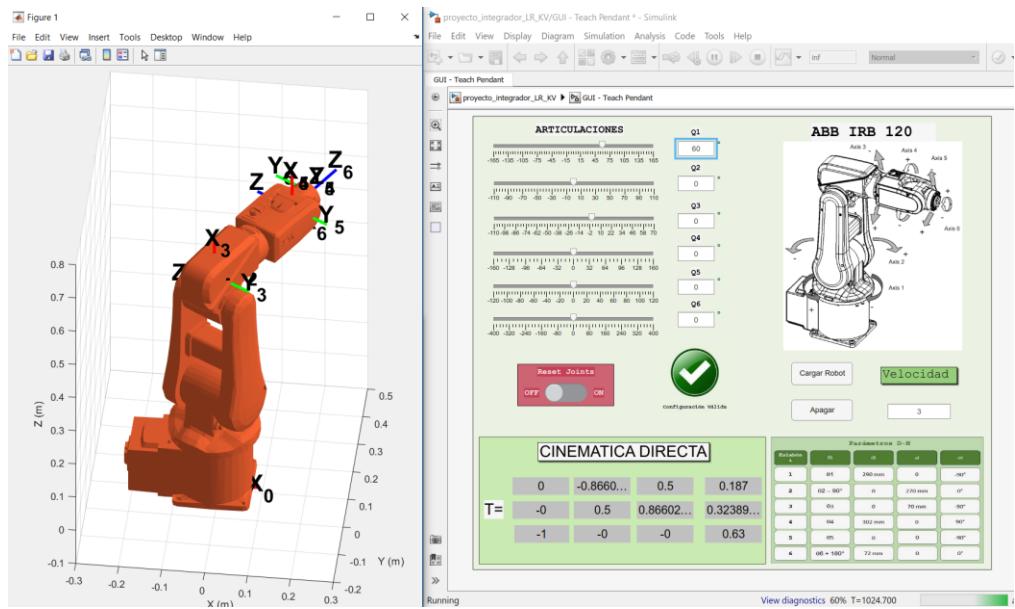


Figura 3.11 Movimiento de articulaciones en simulación 3D

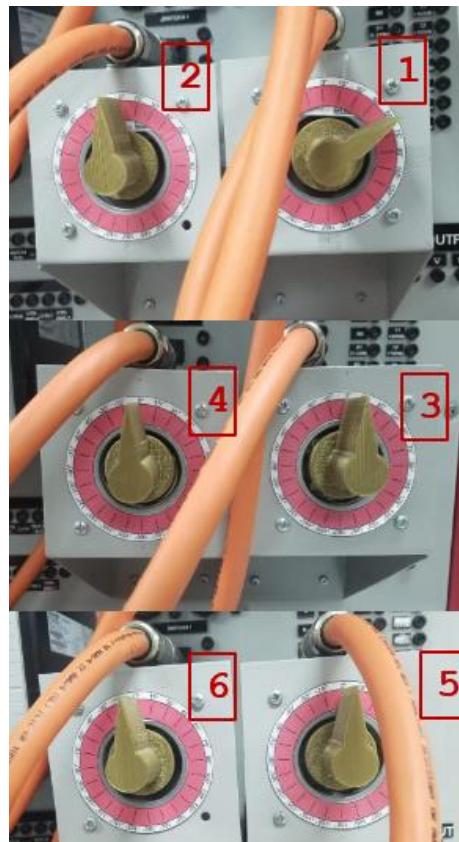


Figura 3.12 Movimiento de servomotores

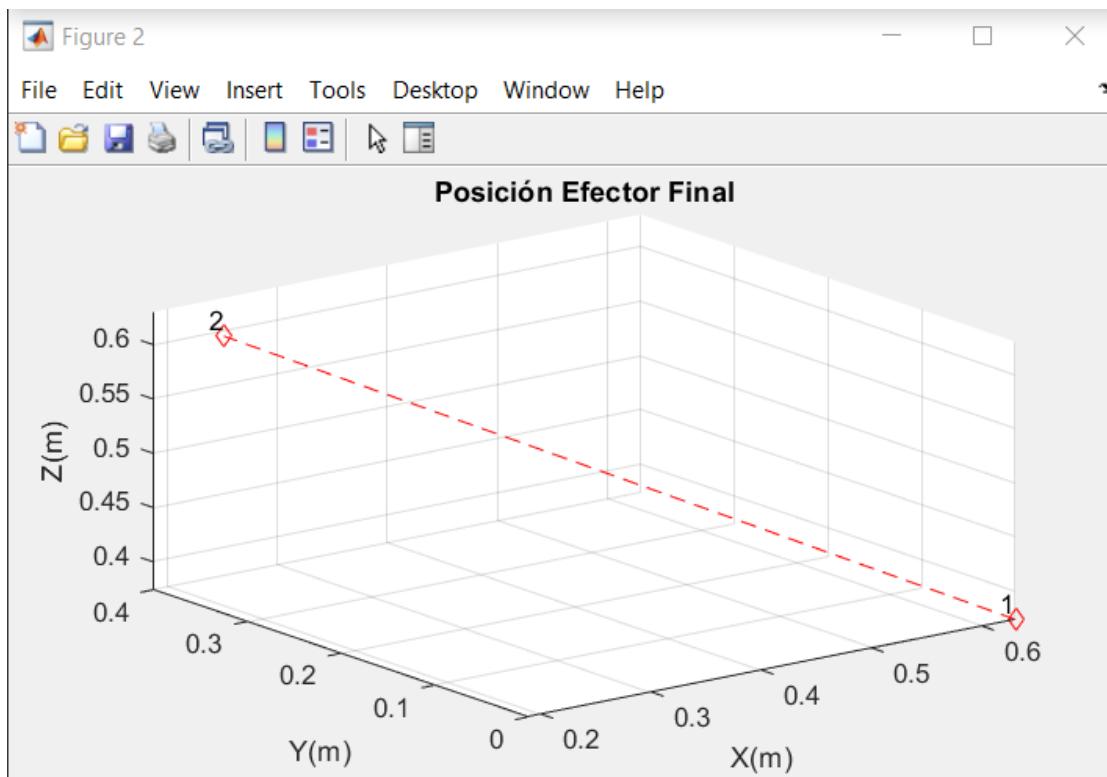
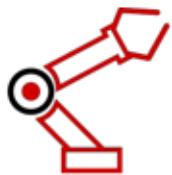


Figura 3.13 Gráfica de la posición del elemento terminal

3.5 Manual de usuario

Se adjuntó un manual de usuario (ver anexos), en donde se especifica las configuraciones realizadas en MATLAB Simulink, Studio 5000 y KEPServerEX para poner en marcha el proyecto. Aquí se detalla a profundidad el funcionamiento de cada componente de la GUI y como ésta interactúa con los demás entornos.



Sistema robótico simulado
ABB IRB 120

MANUAL DE USUARIO

Español

Versión 1.0

Septiembre 2019

Figura 3.14 Manual de usuario del sistema

CAPÍTULO 4

4. CONCLUSIONES Y RECOMENDACIONES

4.1 Conclusiones

- Se diseñó una interfaz gráfica mediante la librería Dashboard de Simulink, que permite monitorear y visualizar la simulación 3D y los servomotores correspondientes a cada una de las articulaciones, trabajando de manera simultánea.
- Se validó las configuraciones para las articulaciones del robot dentro de su espacio de trabajo, siendo de gran importancia para la planificación de trayectorias del elemento terminal.
- Se logró generar trayectorias de mayor precisión mediante el joystick para el elemento terminal del robot. Esto es de gran ayuda para la generación de figuras y números.
- Se consiguió simular y modelar las articulaciones del brazo robótico con los equipos y software de Rockwell Automation.
- Se midió la posición angular para cada articulación con un encoder, los valores de incertidumbres y errores se detallan a continuación:

Tabla 3.1 Medición de posiciones angulares para articulaciones

Articulación Q	Posición angular (°)	Valor medido (°)	Incertidumbre	Error %
1	0.00	0.00684	± 0.00684	0.684
2	0.00	0.01656	± 0.01656	1.656
3	0.00	4.3200e-03	± 4.3200e-03	0.432
4	0.00	0.02232	± 0.02232	2.2232
5	0.00	0.00756	± 0.00756	0.756
6	0.00	0.0036	± 0.0036	0.36

- Se obtuvo un tiempo de respuesta aproximadamente de 1.26 segundos. El uso de una mayor cantidad de aplicaciones en el

computador, ocupa mayores recursos del sistema operativo, haciendo más lenta la respuesta.

4.2 Recomendaciones

- Para la comunicación de los dispositivos es importante examinar su revisión y su respectiva versión del software.
- Para una rápida respuesta en el movimiento de los servomotores, se recomienda realizar los cálculos de posición sobre el entorno de Studio 5000.
- Adquirir licencia del software KEPServerEX, caso contrario se obtendrá un problema de comunicación del OPC cada dos horas, además de tener un límite de prueba de 30 días.
- Se recomienda configurar la puesta en marcha del servo-drive en modo absoluto, para obtener la posición actual exacta en referencia al cero del sistema.
- Verificar la precisión de los equipos de motion control que vayan a ser utilizados, debido a que de aquello depende su exactitud en los movimientos de cada una de las articulaciones.
- Trabajar con un tiempo de escaneo mínimo, para que el tiempo de respuesta de los dos entornos sea más rápido ante las variaciones en la interfaz de usuario.
- Trabajar con el menor número de aplicaciones para ocupar menos recursos del sistema operativo y obtener un tiempo de respuesta del sistema más bajo.

BIBLIOGRAFÍA

- [1] Zhang Lieping, Zeng Aiqun, Zhang Yunsheng (2007, julio). One Remote Real-Time Communication between MATLAB and PLC Based on OPC Technology [En línea]. Disponible en: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4346760>
- [2] M. Aloomian, "HIL EXPERIMENT USING PLC AND MATLAB OPC TOOLBOX", Tesis de Grado, Dep. Ing. Eléctrica, Universidad del estado de California de Northridge, California, USA, diciembre.
- [3] The Math Works Inc. (2004, October). OPC Toolbox for Use with MATLAB [En línea]. Disponible en: https://www.mathworks.com/help/pdf_doc/opc/opc.pdf.
- [4] A. Barrientos, L. Peñín, C. Balaguer, R. Aracil, Fundamentos de robótica, 2nd Ed. McGraw-Hill, 2007.
- [5] PTC Inc. (2019). kepware kepserverex [En línea]. Disponible en: <https://www.kepware.com/getattachment/5759d980-7641-42e8-b4fb-7293c835a2f9/kepserverex-manual.pdf>
- [6] A. Gutiérrez, "Desarrollo de una interfaz para el control del robot IRB120 desde MATLAB", Tesis de Grado, Dep. Ing. Electrónica y Automática Industrial, Universidad de Alcalá, Madrid, España, 2014.
- [7] Allen-Bardley Inc. Kinetix Servo Drives Specifications [En línea]. Disponible en: https://literature.rockwellautomation.com/idc/groups/literature/documents/td/knx-td003_-en-p.pdf
- [8] Allen-Bradley Inc. Kinetix VP Low Inertia Servo Motors with 063...165 mm Frame Size [En línea]. Disponible en: https://literature.rockwellautomation.com/idc/groups/literature/documents/in/vpl-in001_-en-p.pdf

- [9] Allen-Bradley Inc. Kinetix 5500 Servo Drives User Manual [En línea]. Disponible en:
https://literature.rockwellautomation.com/idc/groups/literature/documents/um/2198-um001_-en-p.pdf
- [10] ABB Inc. Especificaciones del producto IRB 120 [En línea]. Disponible en:
<https://search-ext.abb.com/library/Download.aspx?DocumentID=3HAC035960-005&LanguageCode=es&DocumentPartId=&Action=Launch>

APÉNDICES

APÉNDICE A



Sistema robótico simulado
ABB IRB 120

MANUAL DE USUARIO

Español

Versión 1.0

Septiembre 2019

Manual de usuario del sistema robótico

Descripción general

La Figura 1 muestra el sistema robótico diseñado en MATLAB Simulink, dividido en tres subsistemas: el sistema de control, el sistema de comunicación OPC y el sistema de la interfaz gráfica de usuario.

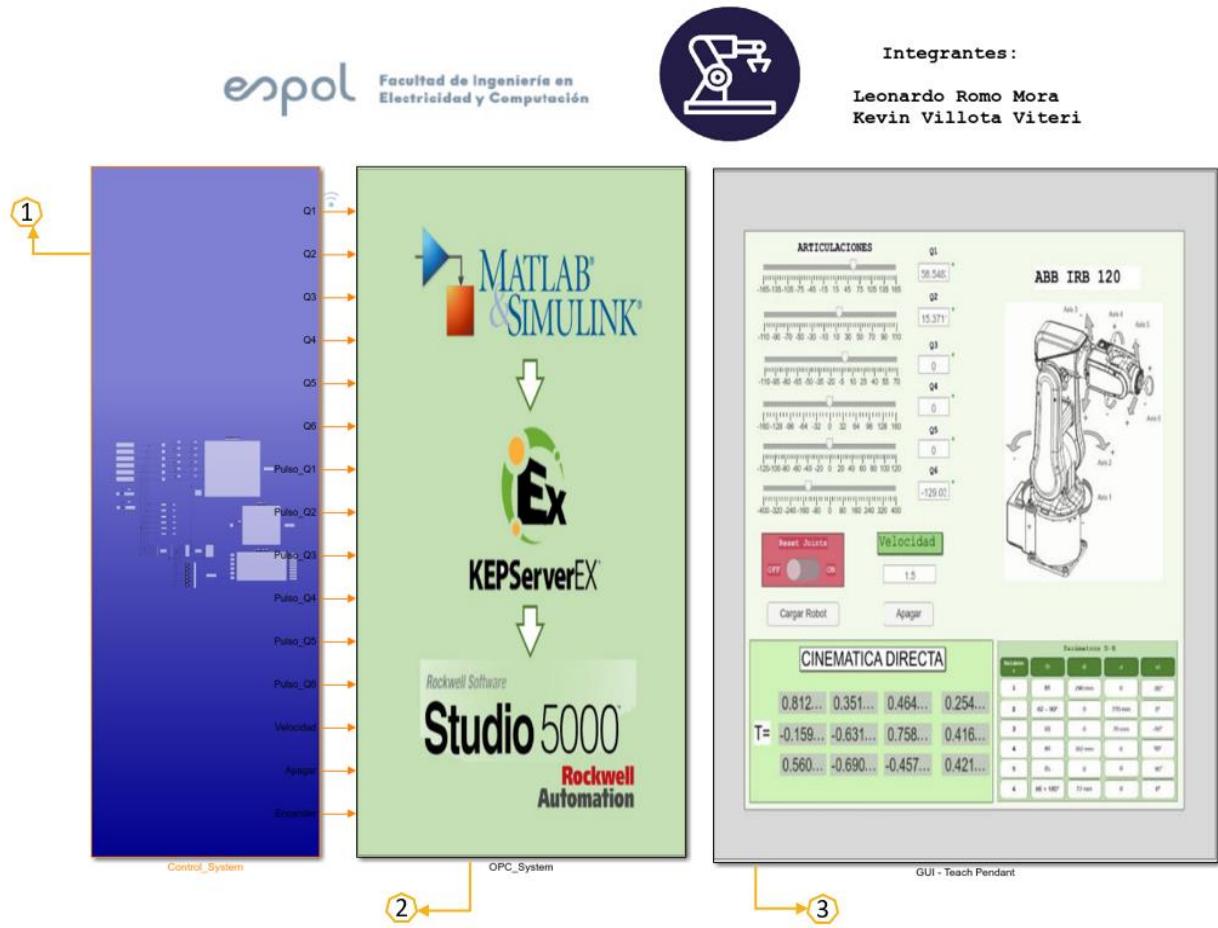


Figura 1. Sistema robótico en MATLAB Simulink

La descripción de cada subsistema se encuentra detallada en la Tabla 1.

Tabla 4.4. Subsistemas del sistema robótico

Número	Subsistema	Descripción
1	Control	Contiene los bloques que permiten establecer un control con la GUI y enviar los parámetros al subsistema del OPC.
2	OPC	Contiene los bloques que permiten establecer comunicación con el software Studio 5000. Se encarga de enviar los parámetros a los servovariadores para el control de los servomotores.
3	GUI – Teach Pendant	Permite al usuario interactuar con la simulación 3D y el movimiento de los servomotores. También es conocida como la consola de programación del sistema robótico.

Cargar el robot en MATLAB

1. Descargar la librería ARTE que contiene modelos CAD de robots de varios fabricantes para su simulación 3D. Link: https://arvc.umh.es/arte/index_en.html
2. Para añadir la ruta de la carpeta ARTE al proyecto, ir a “Browse for folder” y seleccionar la carpeta. Una vez seleccionada la ruta, dar clic derecho sobre la carpeta “arte” en la sección de “Current Folder”, seleccionar “Remove from Path” → “Selected Folders and Subfolders”.

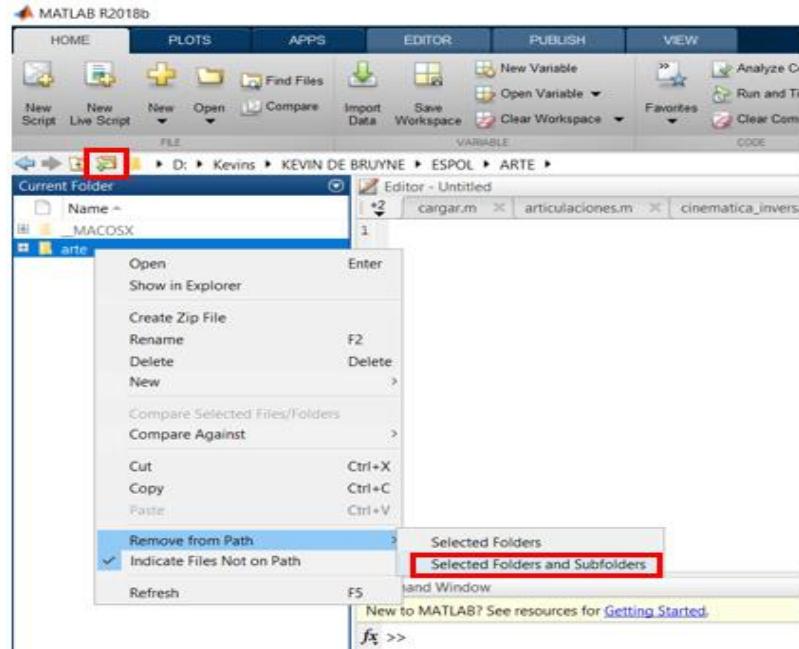


Figura 2 Añadir ruta de librería al proyecto

3. En el “Command Window”, evaluar: `init_lib` para inicializar la librería.

```

>> init_lib
% ARTE (A Robotics Toolbox for Education)
% Copyright (C) 2012 Arturo Gil Aparicio, arturo.gil@umh.es
% http://arvc.umh.es/arte

%
% This program is free software: you can redistribute it and/or modify
% it under the terms of the GNU Lesser General Public License as published by
% the Free Software Foundation, either version 3 of the License, or
% any later version.

%
% This program is distributed in the hope that it will be useful,
% but WITHOUT ANY WARRANTY; without even the implied warranty of
% MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
% GNU Lesser General Public License for more details.

%
% You should have received a copy of the GNU Lesser General Public License
% along with this program. If not, see <http://www.gnu.org/licenses/>.
%
```

Figura 3 Inicializar la librería ARTE

4. Los bloques para modelar el sistema robótico presentes en el sistema de control necesitan de un tiempo de muestreo para actualizar ciertas variables de control e indicadores a medida que corre la simulación, evaluar en la ventana de comandos:

`T = 0.05;`

- Una vez añadida la librería y evaluado el tiempo de muestreo, correr los archivos "articulaciones.m" y "cargar.m" necesarios para la simulación en Simulink. Clic en "Run" → "Add to Path".

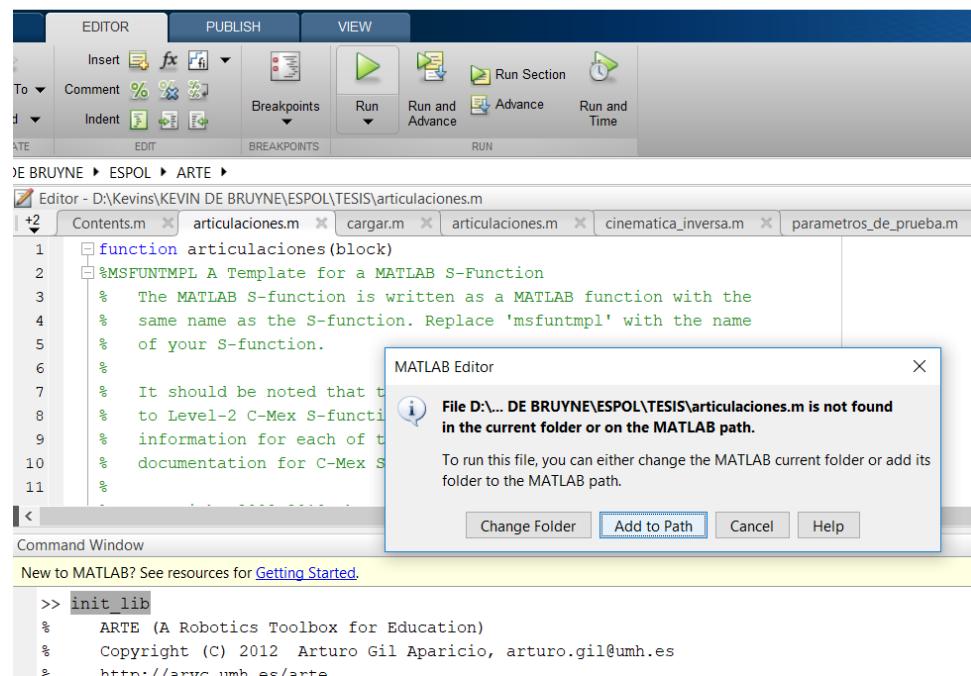


Figura 4 Correr archivos para bloques de simulink

Configuración KEPServer EX

- Abrir el software KepServerEX 6, en la barra herramienta clic en "File" → "Open" y seleccionar el archivo "Proyecto_integrador". Ver Figura 4.

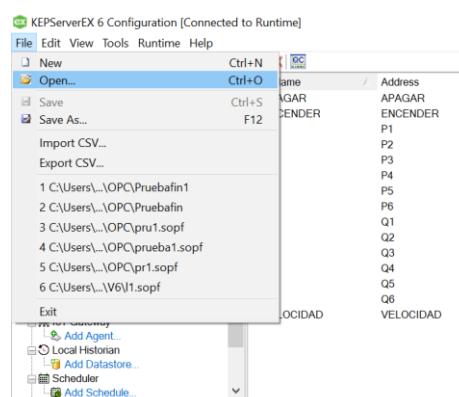


Figura 5 Abrir el proyecto del servidor

2. En el proyecto, dirigirse a “Comunicación” → “Device”, dar doble clic y dentro del Property Editor ir a la sección general, para configurar la dirección IP del PLC con su respectivo slot. Ver Figura 5.

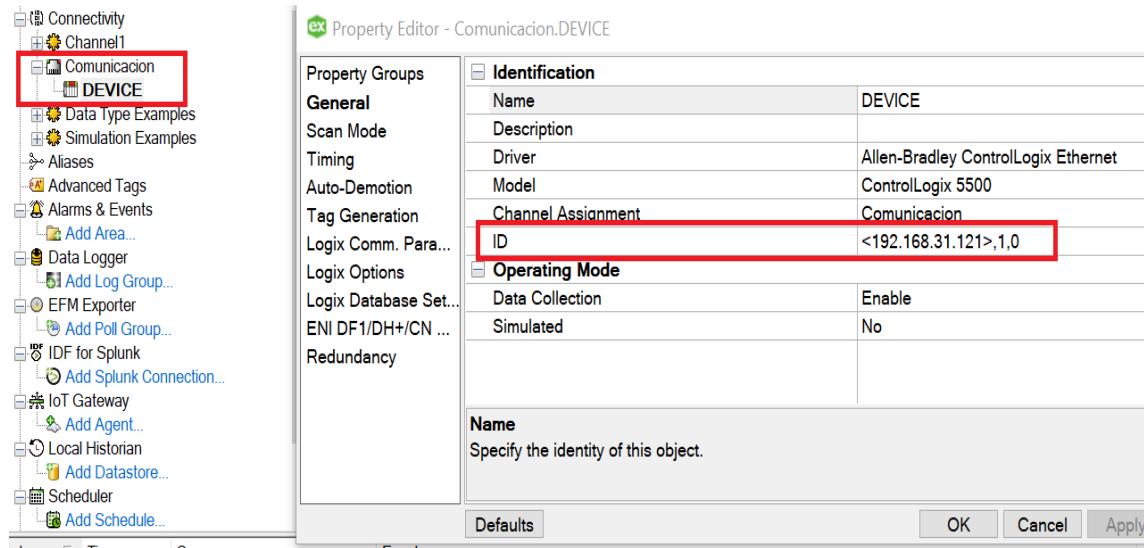


Figura 6 Configuración de la dirección OPC en el servidor

3. Se conectará el servidor en la barra de herramientas dirigirse a “Runtime” → “Connect”. Ver figura 6.

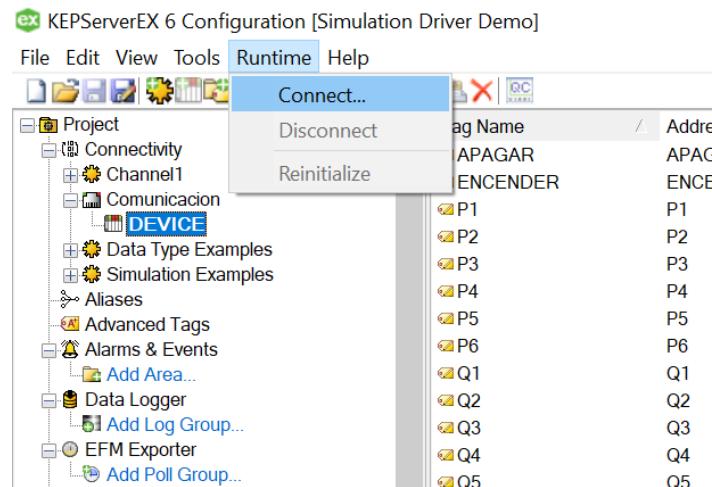


Figura 7 Conectar el servidor

4. Se verifica la correcta comunicación en el servidor dirigiéndose a “Tools” → “Launch OPC Quick Client”. Ver figura 7.

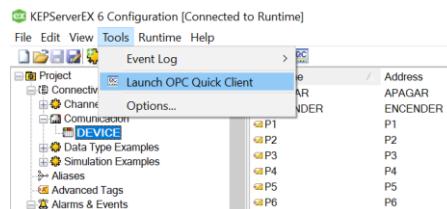


Figura 8 Launch OPC Quick Client

5. Dentro del Kepware diríjase al ítem “Comunicación.Device”. En esta sección podrá verificar, el estado de comunicación de los dos clientes con cada una de sus variables compartidas.

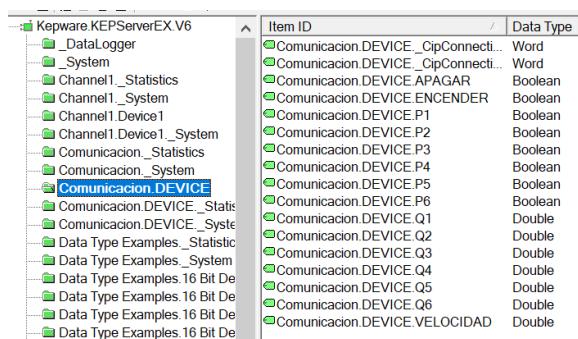


Figura 9 Estado de comunicación de las variables

Configuración del sistema de comunicación

1. Dentro del entorno de Simulink, se agregan los bloques “OPC Configuration” y “OPC Write”, dirigiéndose al “Library Browser” y escribiendo el nombre del bloque.

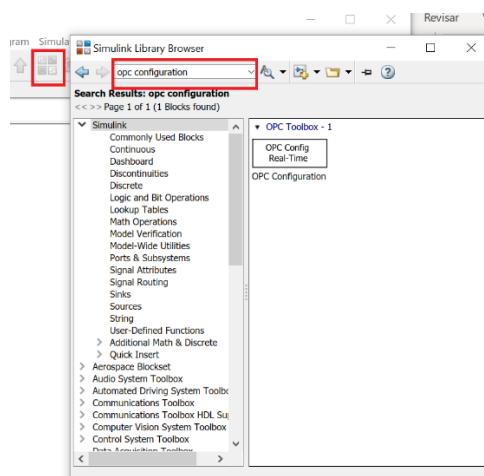


Figura 10 Añadir los bloques OPC

2. Para añadir el OPC a utilizar, dar doble click en el bloque “OPC Configuration” y dirigirse a “Configure OPC Clients” → “Add” → “Select” → “Kepware KepserverEX.V6”. Ver figura 9.

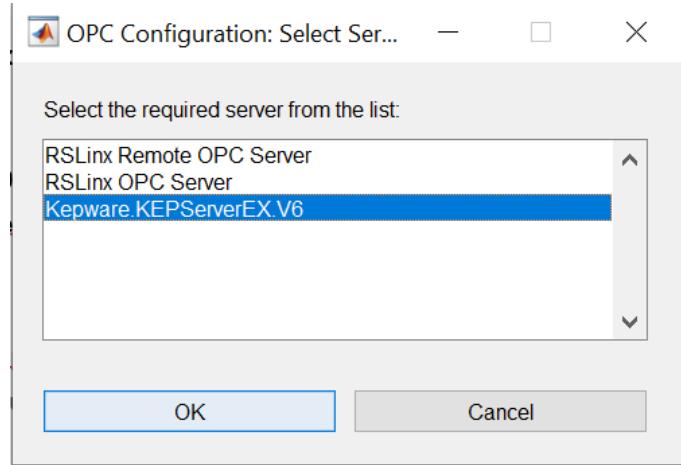


Figura 11 Añadir el OPC

3. Para escribir sobre las variables del servidor, dar doble click en el bloque OPC Write, seleccionar “Add Items” y buscar el nombre del proyecto en el “Available Server Items”.

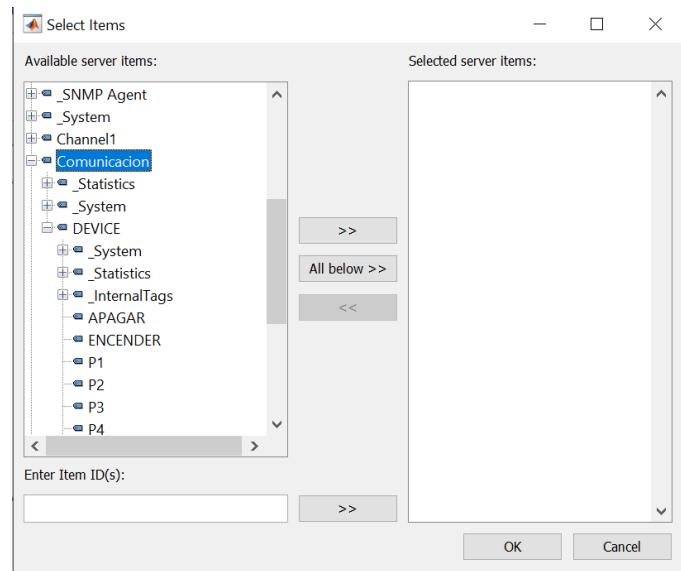


Figura 12 Available Server Items

Cargar programación al controlador

1. Abrir el software Studio 5000, en la barra herramienta clic en “File” → “Open” y seleccionar el archivo “Proyecto_Integrador_Final”. Ver Figura 2.

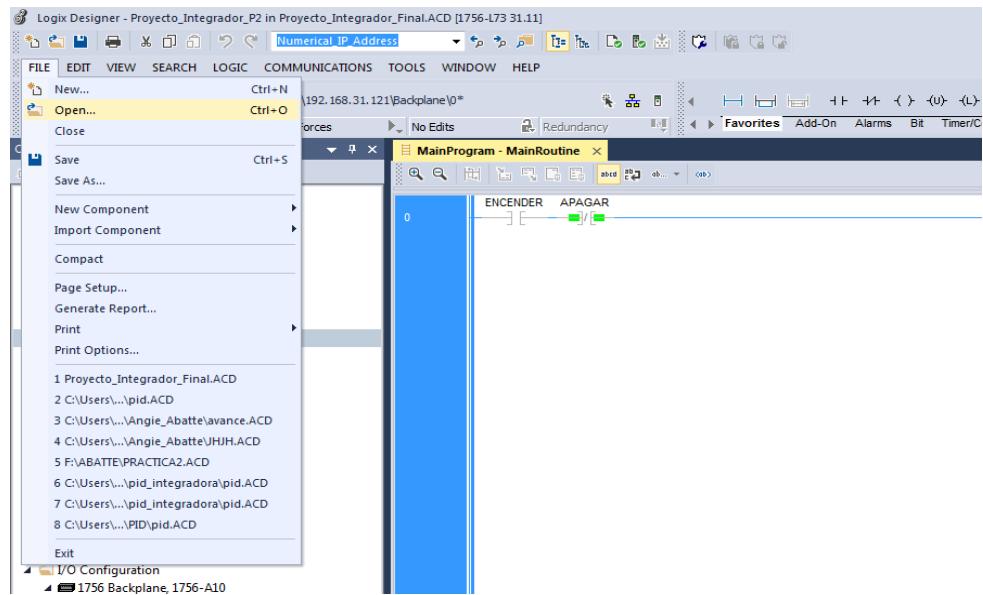


Figura 13 Abrir programa en Studio 5000

2. Verificar que los dispositivos se encuentren conectados a la red Ethernet IP. Dirigirse a la sección “I/O Configuration” en la pestaña “Controller Organizer”, en el módulo Ethernet se muestran los equipos conectados. Ver Figura 3.

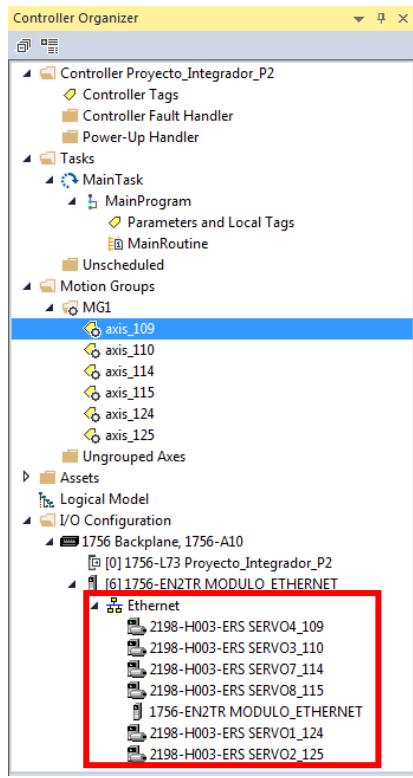


Figura 14 Verificar conexión de dispositivos

3. En la barra de herramientas dar clic en “Communications” → “Who Active”, para crear la ruta de acceso donde se cargará la programación. Ver Figura 4.

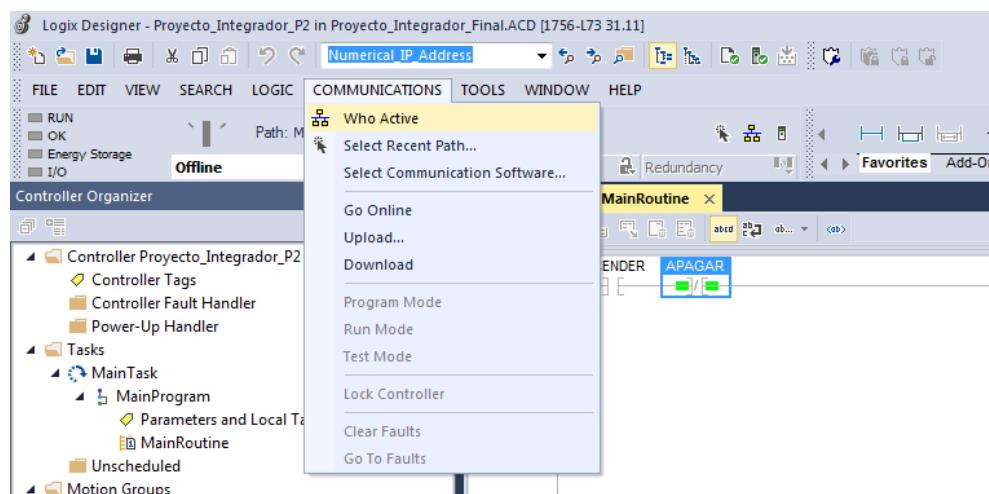


Figura 15 Ruta de acceso para cargar al controlador

4. Finalmente, buscar la dirección IP asignada al controlador en la red. Dar clic en “Download” para cargar la programación al controlador. Ver Figura 5.

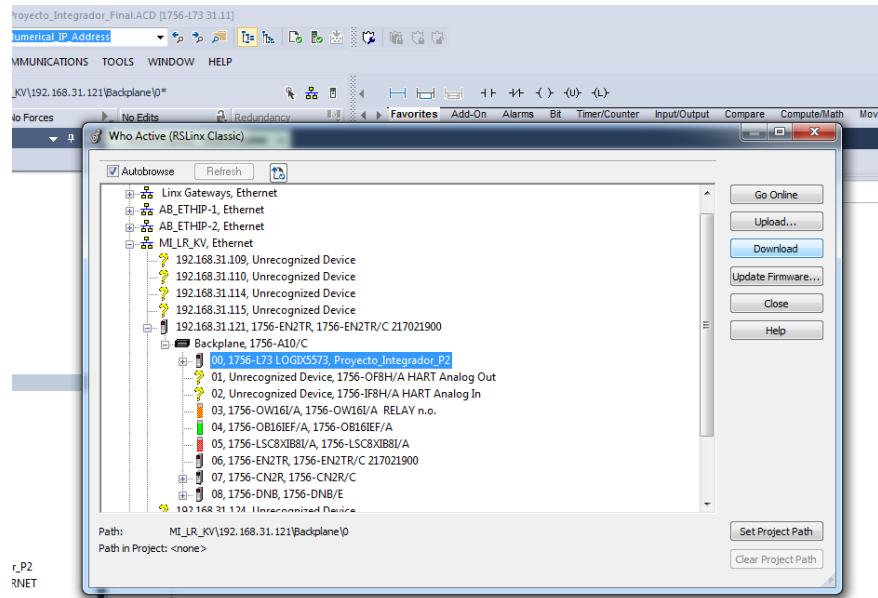


Figura 16 Subir programa al controlador

GUI – Teach Pendant

1. Presionar el botón “Cargar Robot” para que se abra la simulación y se activen los campos de los servomotores. Ver figura 17.

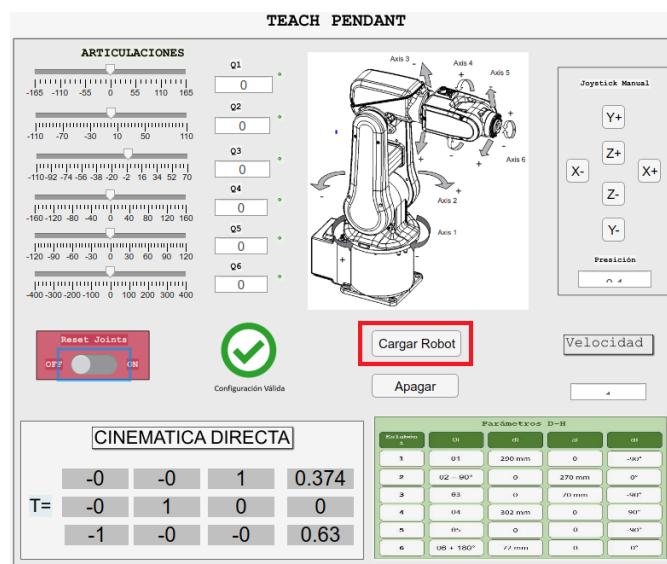


Figura 17 Activar el sistema robótico

2. Mover los sliders que representan la orientación en la que se desea poner a cada articulación. Ver figura 18.

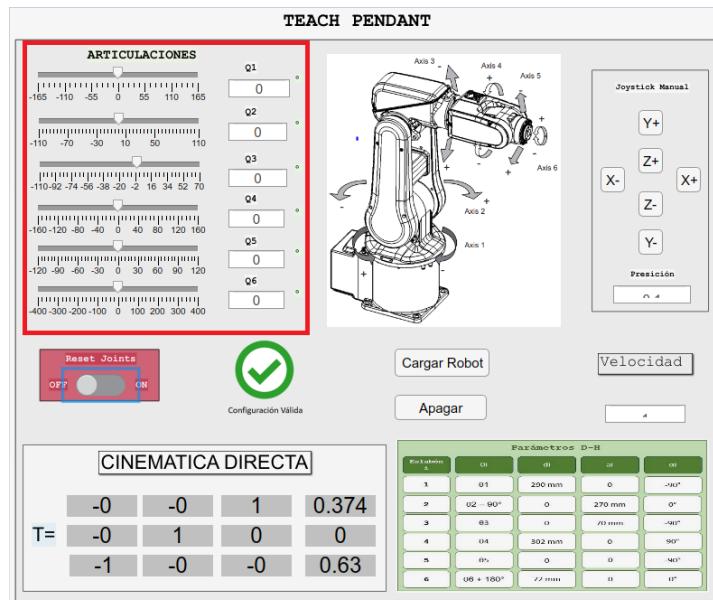


Figura 18 Movimientos de slider que representan las articulaciones

3. Realizar el movimiento a partir del Joystick manual, estableciendo donde se desea ubicar el elemento terminal. Ver figura 19.

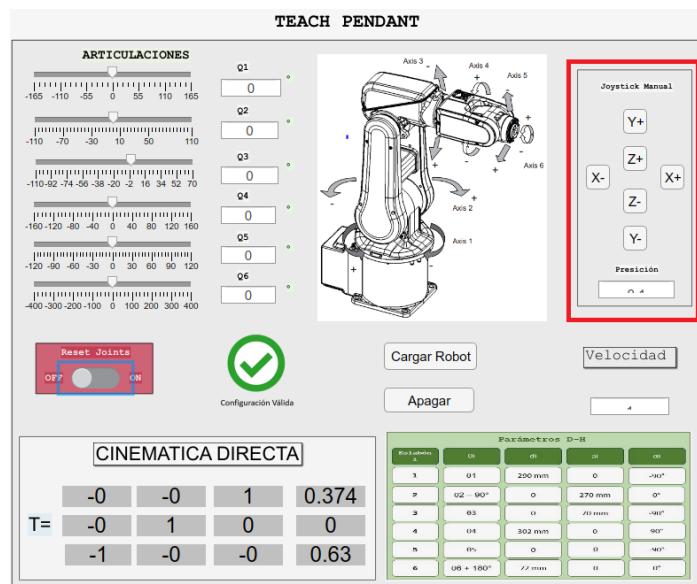


Figura 19 Joystick manual

4. Establecer la velocidad con la cual se desea mover las articulaciones. Ver figura 20.

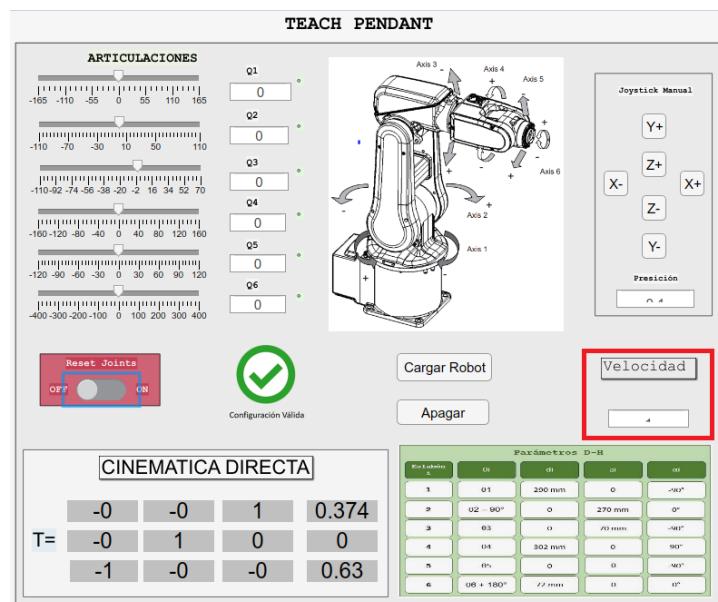


Figura 20 Establecer velocidad

5. Reiniciar las articulaciones para ir a la posición de HOME. Ver figura 21.

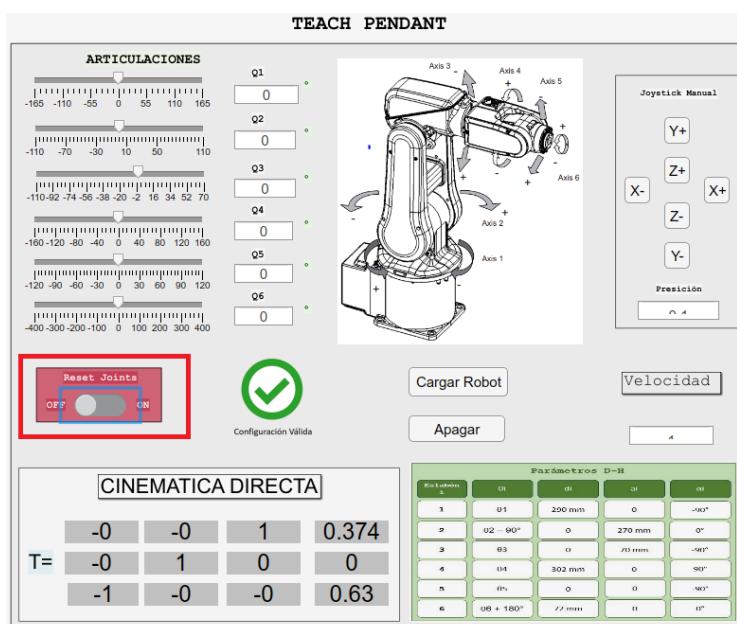


Figura 21 Reinicio de articulaciones

6. Presionar el botón apagar una vez ya realizada las pruebas para desactivar los campos magnéticos de los servomotores. Ver figura 22.

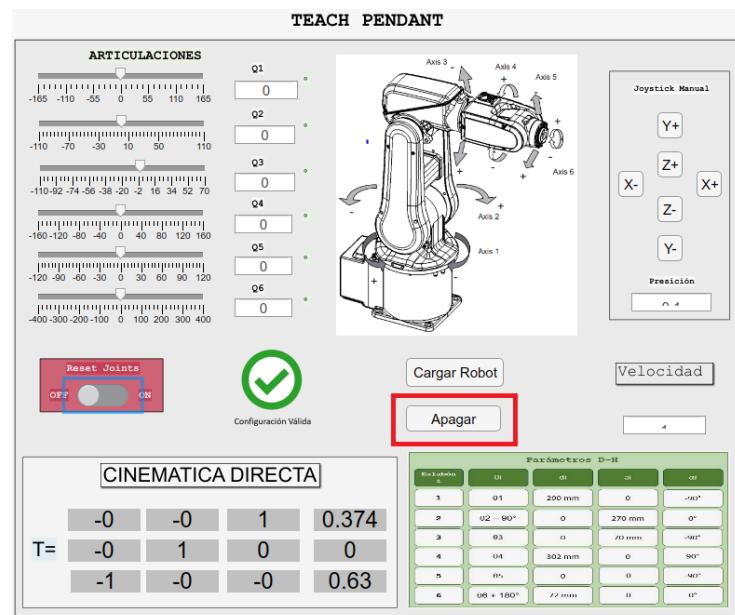


Figura 22 Apagar el sistema robótico

ANEXOS

ANEXO A: FICHA TÉCNICA DEL ROBOT

1.1.2 El robot

Generalidades

El IRB 120-3/0.6 está disponible en dos versiones y ambas admiten el montaje en el suelo, en posición invertida o en pared con cualquier ángulo (inclinado alrededor del eje X o Y).

Tipo de robot	Capacidad de manejo (kg)	Alcance (m)
IRB 120	3 kg	0,58 m

Peso del manipulador

Datos	Peso
IRB 120-3/0.6	25 kg

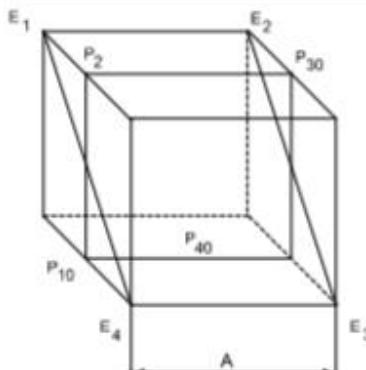
Otros datos técnicos

Datos	Descripción	Nota
Nivel de ruido propagado por el aire	Nivel de presión sonora exterior	< 70 dB (A) Leq (de acuerdo con la Directiva de máquinas 2006/42/CE para áreas de trabajo)

Consumo de potencia

Trayectoria E1-E2-E3-E4 en el cubo ISO, con carga máxima.

Tipo de movimiento	Consumo de potencia (kW)
Velocidad máxima de cubo ISO	0,24 kW
Robot en posición de 0 grados	IRB 120
Frenos aplicados	0,095 kW
Frenos desactivados	0,173 kW



xx0900000265

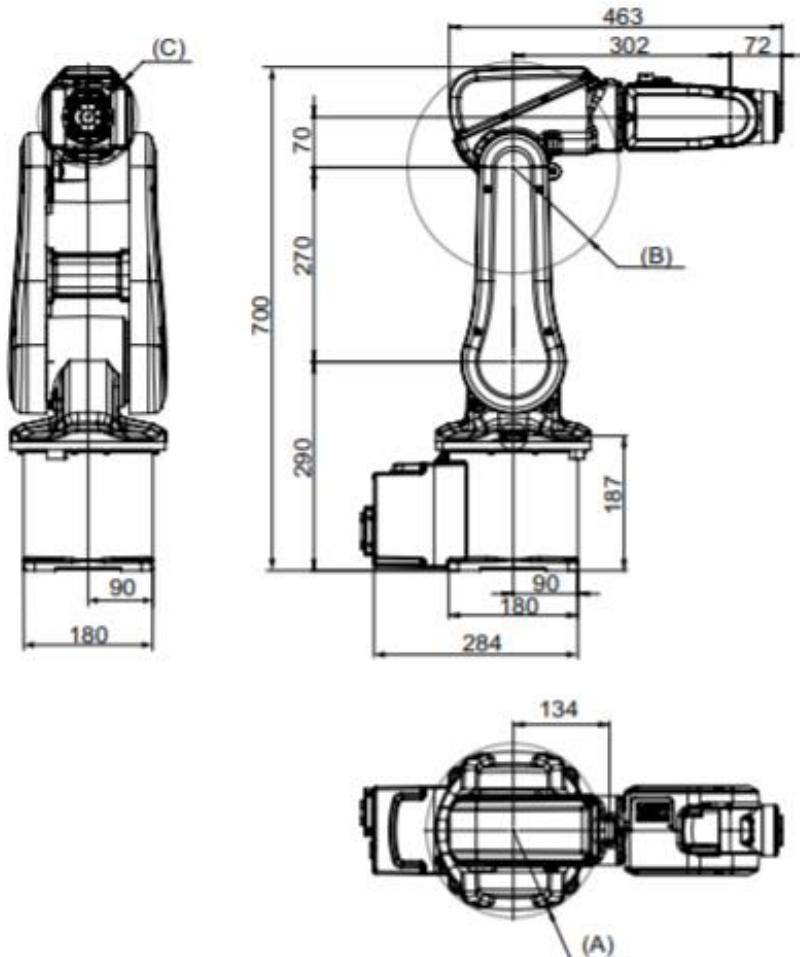
Posición	Descripción
A	250 mm

Continúa en la página siguiente

1 Descripción

1.1.2 El robot Continuación

Dimensiones IRB 120-3/0.6



xx000000256

Posición	Descripción
A	Radio mínimo de giro del eje 1 R=121 mm
B	Radio mínimo de giro del eje 3 R=147 mm
C	Radio mínimo de giro del eje 4 R=70 mm

1 Descripción

1.8 Área de trabajo y tipo de movimiento

1.8 Movimiento del robot

1.8.1 Área de trabajo y tipo de movimiento

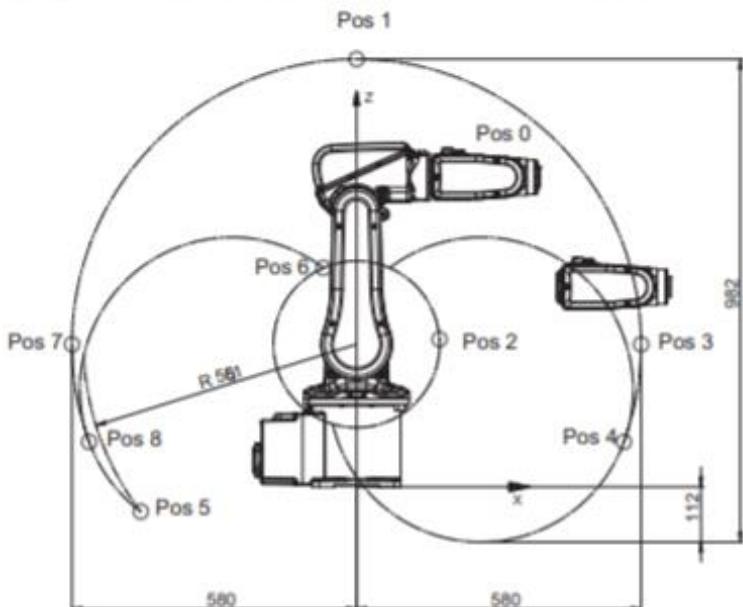
Área de trabajo

En las figuras se detallan las áreas de trabajo del robot.

Las posiciones extremas del brazo de robot se especifican respecto del centro de la muñeca (dimensiones en mm).

Área de trabajo

La figura muestra el área de trabajo sin restricciones del robot.



xx0900000263

Posi- ción	Posición en el centro de la muñeca (mm)		Ángulo (grados)	
	X	Z	Eje 2	Eje 3
A	302 mm	630 mm	0°	0°
B	0 mm	870 mm	0°	-77°
C	169 mm	300 mm	0°	+70°
D	580 mm	270 mm	+90°	-77°
E	545 mm	91 mm	+110°	-77°
F	-440 mm	-50 mm	-110°	-110°
G	-67 mm	445 mm	-110°	+70°
H	-580 mm	270 mm	-90°	-77°
J	-545 mm	91 mm	-110°	-77°

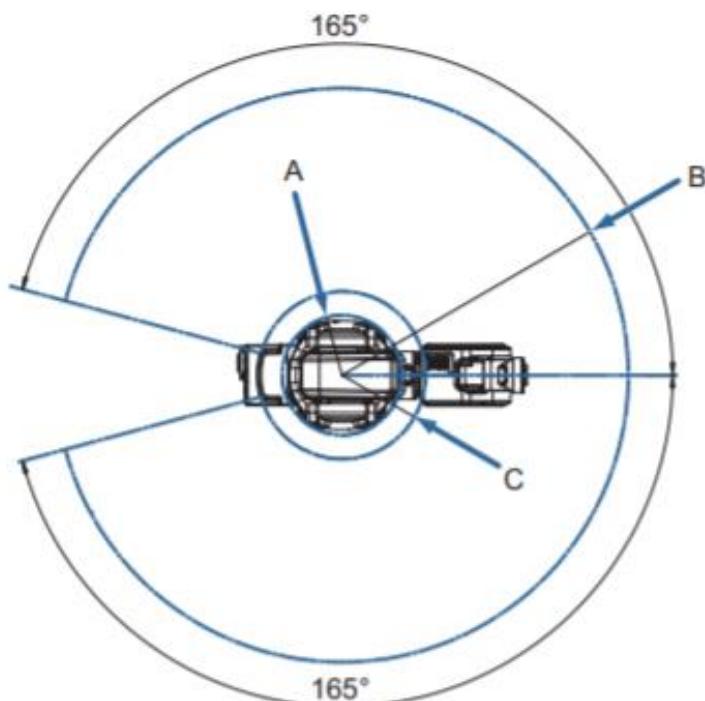
1 Descripción

1.8.1 Área de trabajo y tipo de movimiento

Continuación

Radio de giro

El radio de giro del robot se muestra en la figura.



xx09000000157

Variante de robot	Pos. A	Pos. B	Pos. C
IRB 120-3/0.6	R121 ⁱ	R580	R169.4

ⁱ Radio de giro mínimo del eje 1.

Movimiento del robot

En la tabla se especifican los tipos y áreas de movimiento de todos los ejes.

Ubicación del movimiento	Tipo de movimiento	Área de movimiento
Eje 1	Movimiento de rotación	De +165° a -165°
Eje 2	Movimiento del brazo	De +110° a -110°
Eje 3	Movimiento del brazo	De +70° a -110°
Eje 4	Movimiento de la muñeca	De +160° a -160°
Eje 5	Movimiento de doblado	De +120° a -120°
Eje 6	Movimiento de giro	De +400° a -400° (de forma predeterminada) De +242 a -242 revoluciones como máximo ⁱ

ⁱ El área de trabajo predeterminada para el eje 6 puede ampliarse mediante el cambio de valores de parámetros en el software.

La opción 610-1 "Independent axis" puede utilizarse para restablecer el cuentarrevoluciones tras el giro del eje (sin necesidad de "rebobinar" el eje).

1.8.3 Velocidad

Generalidades

Tipo de robot	Eje 1	Eje 2	Eje 3	Eje 4	Eje 5	Eje 6
IRB 120 - 3/0.6	250 °/s	250 °/s	250 °/s	320 °/s	320 °/s	420 °/s

Se requiere supervisión para evitar sobrecalentamientos en aplicaciones que requieren movimientos fuertes y frecuentes.

Resolución

Aprox. 0,01 ° en cada eje.

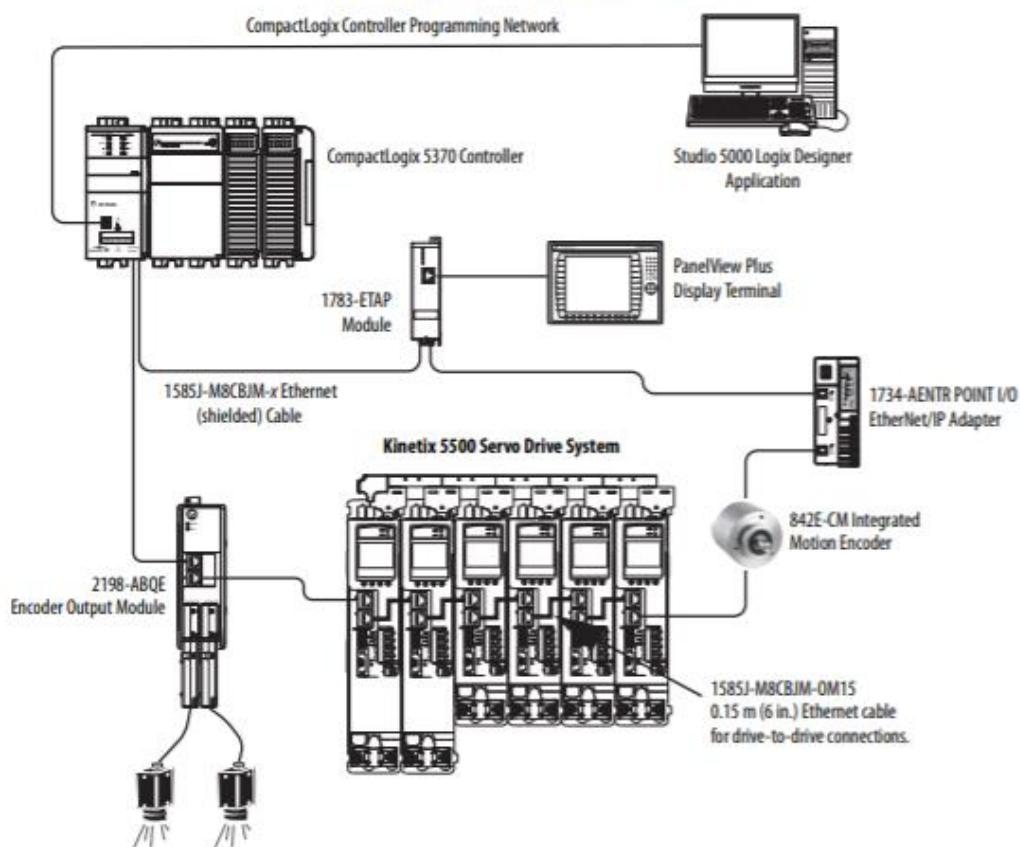
ANEXO B: TOPOLOGÍA EN ANILLO PARA KINETIX 5500

Ring Topology

In this example, the devices are connected by using ring topology. If only one device in the ring is disconnected, the rest of the devices continue to communicate. For ring topology to work correctly, a device level ring (DLR) supervisor is required (for example, the Bulletin 1783 ETAP device). DLR is an ODVA standard. For more information, refer to the EtherNet/IP Embedded Switch Technology Application Guide, publication [ENET-AP005](#).

Devices without dual ports, for example the display terminal, require a 1783-ETAP module to complete the network ring.

Figure 8 - Kinetix 5500 Ring Communication Installation



ANEXO C: FICHA TÉCNICA KINTEIX 5500

Kinetix Servo Drives Specifications

Technical Specifications - Kinetix 5500 Servo Drives

Kinetix 5500 Input Power Specifications

Attribute	2198-H003-ERS 2198-H003-ERS2	2198-H008-ERS 2198-H008-ERS2	2198-H015-ERS 2198-H015-ERS2	2198-H025-ERS 2198-H025-ERS2	2198-H040-ERS 2198-H040-ERS2	2198-H070-ERS 2198-H070-ERS2
AC input voltage	195...264V rms, single-phase (240V nom) 195...264V rms, three-phase (240V nom) 324...528V rms, three-phase (480V nom)			195...264V rms, three-phase (240V nom) 324...528V rms, three-phase (480V nom)		
AC input frequency	47...63 Hz					
Main AC input current ⁽¹⁾ 324...528V rms, three-phase 195...264V rms, single-phase	1.0 A 0.7 A	2.60 A 1.70 A	5.20 A 3.40 A	8.30 A N/A	13.4 A N/A	23.7 A N/A
Max inrush (0-pk)	15.0 A			30.0 A		
Peak AC input current 324...528V rms, three-phase 195...264V rms, single-phase	3.0 A 2.1 A	7.8 A 5.1 A	15.6 A 10.2 A	24.9 A N/A	40.2 A N/A	71.1 A N/A
Line loss ride through	20 ms					
Control power DC input voltage	24V DC \pm 10%					
Control power DC input current ^{(1) (2)} (non-brake motors)	0.4 A _{DC}			0.8 A _{DC}		
Nominal bus output voltage	276...747V DC					
Continuous output current to bus 195...264V rms, single-phase 324...528V rms, three-phase	0.67 A _{DC} 1.0 A _{DC}	1.60 A _{DC} 2.40 A _{DC}	3.27 A _{DC} 4.90 A _{DC}	N/A 7.80 A _{DC}	N/A 12.7 A _{DC}	N/A 22.50 A _{DC}
Peak output current to bus ⁽³⁾ 195...264V rms, single-phase 324...528V rms, three-phase	2.0 A _{DC} 3.0 A _{DC}	4.8 A _{DC} 7.2 A _{DC}	9.8 A _{DC} 14.7 A _{DC}	N/A 23.4 A _{DC}	N/A 38.1 A _{DC}	N/A 67.5 A _{DC}
Continuous output power to bus Nom (230V rms, single-phase) Nom (230V rms, three-phase) Nom (480V rms, three-phase)	0.2 kW 0.3 kW 0.6 kW	0.5 kW 0.8 kW 1.6 kW	1.0 kW 1.6 kW 3.2 kW	N/A 2.5 kW 5.2 kW	N/A 4.0 kW 8.4 kW	N/A 7.2 kW 14.9 kW
Peak output power to bus Nom (230V rms, single-phase) Nom (230V rms, three-phase) Nom (480V rms, three-phase)	0.6 kW 1.0 kW 1.9 kW	1.6 kW 2.4 kW 4.9 kW	3.2 kW 4.9 kW 9.7 kW	N/A 7.8 kW 15.6 kW	N/A 12.7 kW 25.3 kW	N/A 22.4 kW 44.8 kW
DC input voltage (common bus follower)	276...747V DC					
DC input current (common bus follower)	1.0 A _{DC}	2.4 A _{DC}	4.9 A _{DC}	7.8 A _{DC}	12.7 A _{DC}	22.5 A _{DC}
Bus overvoltage	240V, nom AC input	440V DC				
	480V, nom AC input	810V DC				
Bus undervoltage	240V, nom AC input	138V DC				
	480V, nom AC input	275V DC				
Efficiency	97%					
Capacitive energy absorption	11.54 J		19.58 J	39.15 J	58.73 J	104.87 J
Short-circuit current rating	200.000 A (rms) symmetrical					

(1) All drives are limited to 1 power cycle per minute.

(2) For current values when motors include a holding brake and additional information, refer to [Control Power Current Specifications](#) on page 49.

(3) Peak output current duration equals 1.0 second.

Kinetix 5500 Output Power Specifications

Attribute	2198-H003-ERS 2198-H003-ERS2	2198-H008-ERS 2198-H008-ERS2	2198-H015-ERS 2198-H015-ERS2	2198-H025-ERS 2198-H025-ERS2	2198-H040-ERS 2198-H040-ERS2	2198-H070-ERS 2198-H070-ERS2
Bandwidth ⁽¹⁾ Velocity loop, max Current loop	300 Hz 1000 Hz					
PWM frequency	8 kHz	4 kHz				
Continuous output current (rms)	1.0 A	2.5 A	5.0 A	8.0 A	13.0 A	23.0 A
Continuous output current (0-pk)	1.4 A	3.5 A	7.1 A	11.3 A	18.4 A	32.5 A
Peak output current (rms) ⁽²⁾	2.5 A	6.25 A	12.5 A	20.0 A	32.5 A	57.5 A
Peak output current (0-pk) ⁽²⁾	3.5 A	8.8 A	17.7 A	28.3 A	45.9 A	81.3 A
Continuous power out (nom) 195...264V rms, single-phase 195...264V rms, three-phase 324...528V rms, three-phase	0.2 kW 0.3 kW 0.6 kW	0.5 kW 0.8 kW 1.6 kW	1.0 kW 1.5 kW 3.2 kW	N/A 2.4 kW 5.1 kW	N/A 4.0 kW 8.3 kW	N/A 7.0 kW 14.6 kW
Internal shunt resistance	100 Ω		60 Ω			40 Ω
Internal shunt power	30 W		50 W			75 W
Shunt on	775V plus 30V x bus regulator capacity/utilization ⁽³⁾					
Shunt off	765V plus 30V x bus regulator capacity/utilization ⁽³⁾					

(1) Bandwidth values vary based on tuning parameters and mechanical components.

(2) Peak current duration (T_{PKmax}) equals 1.0 second.

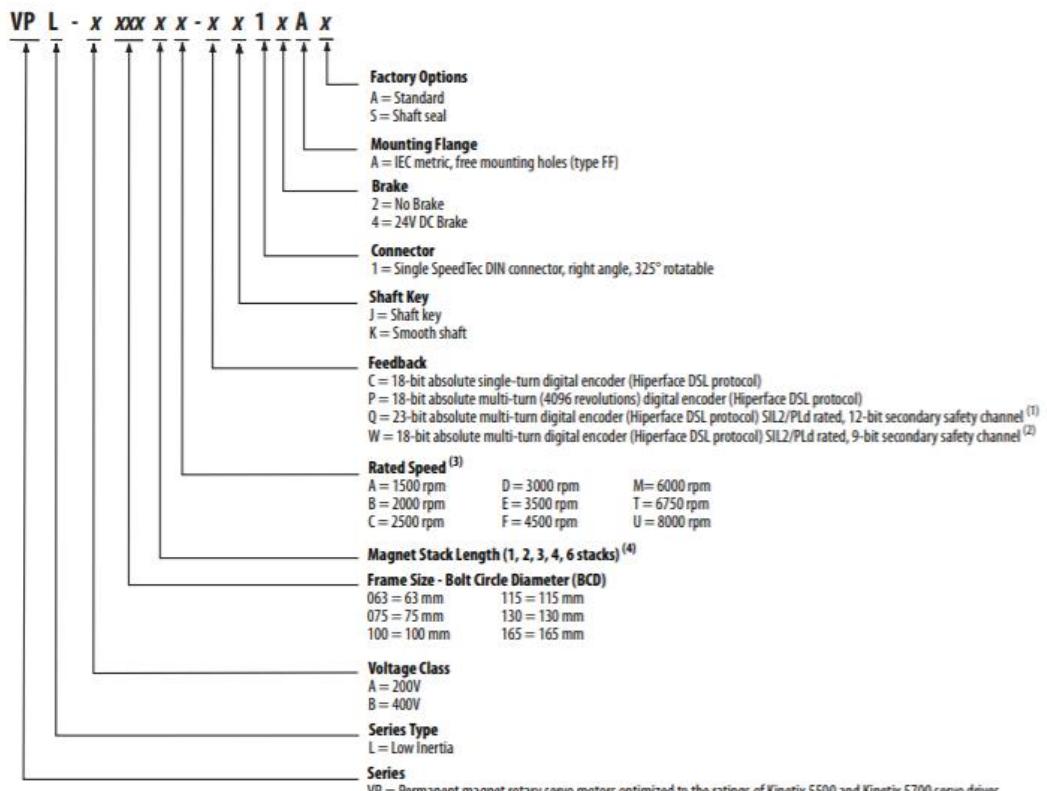
(3) The shunt on and shunt off voltages increase during periods of shunting activity to promote sharing of shunt power in multi-axis configurations. Shunt utilization is equivalent to the BusRegulatorCapacity tag in the Logix Designer application.

ANEXO D: NÚMERO DE CATÁLOGO DEL SERVOMOTOR

SERVO VPL-A1001M-CJ12AA

Kinetix VP Low Inertia Servo Motors with 063...165 mm Frame Size

Catalog Number Explanation



(1) This encoder option is available with only VPL-A/B100xx, VPL-A/B115xx, VPL-A/B130xx, and VPL-B165xx motor frame sizes.

(2) This encoder option is available with only VPL-B063xx and VPL-B075xx motor frame sizes.

(3) Rated speed hierarchy is only for comparative purposes. Use Motion Analyzer software to size and select motors for your application, and/or the torque/speed curves in the Kinetix 5500 Drive System Design Guide, publication [KNX-RM009](#) and the Kinetix 5700 Drive System Design Guide, publication [KNX-RM010](#).

(4) Refer to [Motor Dimensions \(063 mm and 075 mm frame sizes\)](#) on page 7 and [Motor Dimensions \(100...165 mm frame sizes\)](#) on page 8 for dimensional changes (L, LB, LD, and LE) that result from the number of magnet stacks.