



**Automatización y supervisión de una estación didáctica para el transporte y  
organización de envases para bebidas.**

Quimbita Panchi, Milton Alexis y Vargas Cañar, Andrés Damián

Departamento de Eléctrica, Electrónica y Telecomunicaciones

Carrera de Electrónica y Automatización

Trabajo de unidad de integración curricular, previo a la obtención del título de  
Ingeniero en Electrónica y Automatización

Ing. Pruna Panchi, Edwin Patricio Mgs.

5 marzo del 2024

Latacunga

### Trabajo de integración curricular Milt...

#### Scan details

Scan time:  
March 4th, 2024 at 2:56 UTC

Total Pages:  
36

Total Words:  
8984

#### Plagiarism Detection



Types of plagiarism		Words
Identical	0.7%	60
Minor Changes	1.1%	101
Paraphrased	0.3%	29
Omitted Words	0%	0

#### AI Content Detection



Text coverage		Words
AI text	0%	0
Human text	100%	8984

[Learn more](#)

#### Plagiarism Results: (7)

[Arduino Mega 2560](#) 1%

<https://importienda.com/es/arduino/31-arduino-mega-2560.html>

...

[Diseño e implementación de un sistema de transferencias de datos a tra...](#) 0.6%

<https://dspace.ups.edu.ec/bitstream/123456789/13460/1/ups-gt001796.pdf>

Ponce León, David

UNIVERSIDAD POLITÉCNICA SALESIANA SEDE GUAYAQUIL FACULTAD DE INGENIERÍAS CARRERA DE INGENIERÍA ELECTRÓNICA PROYECTO TÉCNICO PREVIO A LA...

[Arduino Mega 2560 REV3 - CREA ELECTRONICS](#) 0.6%

<https://creaelectronics.com/product/arduinomega-2560/>

Saltar al contenido INICIO COMPONENTES CONTÁCTANOS MI CUENTA CARRITO MI PEDIDO Buscar... Menú de n...

[MEMORIAS DE LOS PLC | PDF](#) 0.5%

<https://www.slideshare.net/dannyanderson19/memorias-de-los-plc>

Submit Search Upload MEMORIAS DE LOS PLC • 4 likes • 18,272 views D Danny Anderson Follow Tipos de memorias de un PLC Read less Read mo...

Firma:

Ing. Pruna Panchi, Edwin Patricio, Mgs.



Departamento de Eléctrica, Electrónica y Telecomunicaciones

Carrera de Electrónica y Automatización

#### Certificación

Certifico que el trabajo de unidad de integración curricular: **“Automatización y supervisión de una estación didáctica para el transporte y organización de envases para bebidas.”** fue realizado por los señores **Quimbíta Panchi, Milton Alexis** y **Vargas Cañar, Andrés Damián** el mismo que cumple con los requisitos legales, teóricos, científicos, técnicos y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, además fue revisado y analizada en su totalidad por la herramienta de prevención y/o verificación de similitud de contenidos; razón por la cual me permito acreditar y autorizar para que se lo sustente públicamente.

Latacunga, 4 de marzo de 2024

Firma:

Ing. Pruna Panchi, Edwin Patricio, Mgs.

C. C.: 0502651003



**ESPE**  
UNIVERSIDAD DE LAS FUERZAS ARMADAS  
INNOVACIÓN PARA LA EXCELENCIA

Departamento de Eléctrica, Electrónica y Telecomunicaciones

Carrera de Electrónica y Automatización

### Responsabilidad de Autoría

Nosotros, **Quimbita Panchi, Milton Alexis**, con cédula de ciudadanía n°0550012876 y **Vargas Cañar, Andrés Damián**, con cédula de ciudadanía n° 1850345669, declaramos que el contenido, ideas y criterios del trabajo de unidad de integración curricular: **“Automatización y supervisión de una estación didáctica para el transporte y organización de envases para bebidas.”** es de nuestra autoría y responsabilidad, cumpliendo con los requisitos legales, teóricos, científicos, técnicos, y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, respetando los derechos intelectuales de terceros y referenciando las citas bibliográficas.

Latacunga, 4 de marzo de 2024

**Quimbita Panchi, Milton Alexis**

C. C.: 0550012876

**Vargas Cañar, Andrés Damián**

C. C.: 1850345669



Departamento de Eléctrica, Electrónica y Telecomunicaciones

Carrera de Electrónica y Automatización

#### Autorización de Publicación

Nosotros, **Quimbíta Panchi, Milton Alexis**, con cédula de ciudadanía n° 0550012876 y **Vargas Cañar, Andrés Damián**, con cédula de ciudadanía n° 18503345669, autorizamos a la Universidad de las Fuerzas Armadas ESPE publicar el trabajo de unidad de integración curricular: “**Automatización y supervisión de una estación didáctica para el transporte y organización de envases para bebidas.**” en el Repositorio Institucional, cuyo contenido, ideas y criterios son de nuestra responsabilidad

Latacunga, 4 de marzo de 2024

Quimbíta Panchi, Milton Alexis

C. C.: 0550012876

Vargas Cañar, Andrés Damián

C. C.: 1850345669

## **Dedicatoria**

Este Proyecto está dedicado a mis padres ya que ellos fueron los pilares fundamentales, con su apoyo, con sus palabras de aliento que me daban más fuerzas para poder seguir luchando y poder lograr esta meta, y a un miembro de mi familia que, siempre estuvo ahí para mí, y que por azares de la vida ya no se encuentra el día de hoy viéndome lograr esta meta.

A mi hermana Deliany, a quien siempre estoy orgulloso de lo que hace y que con este trabajo que sea motivo de seguir adelante y que cumpla todas sus metas, y que, aunque no se lo demuestre todos los días, uno siempre logra cumplir sus metas con esfuerzo y dedicación cuando hace lo que a uno le gusta hacer.

***Milton Quimbita***

## **Agradecimiento**

En primer lugar, agradezco a Dios por cuidarme, darme salud y poder cumplir con una de las tantas metas que tengo.

También, agradezco a mis Padres Milton Quimbita y Rocio Panchi, ya que ellos estuvieron apoyándome durante toda mi carrera universitaria, dándome consejos, los cuales están dando frutos hoy por hoy, y como no olvidarme de mi hermana, la cual siempre ha estado ahí para mí, acompañándome en mis salidas.

Agradezco al Ing. Pruna, por darme esta oportunidad de realizar este proyecto junto con él, cómo Tutor; por las enseñanzas que a su debido momento nos compartió cuando era su alumno.

Agradezco, también a todos los docentes que impartieron sus conocimientos durante toda mi vida universitaria y estudiantil.

Agradezco a Josué, Andrés, Fernando, José a todos ellos y a los que faltan por nombrarles les agradezco por brindarme esa confianza de ser amigos durante la carrera universitaria.

Como no agradecer a M<sup>2</sup> y J<sup>2</sup>, los cuales son los que me han acompañado en las buenas y en las malas, fuera de mi vida universitaria, haciéndome olvidar de mis malos momentos.

***Milton Quimbita***

**Dedicatoria**

Este proyecto está dedicado para mis padres que me han apoyado en cada etapa de la vida universitaria siendo el soporte e inspiración de perseverancia, dando ese empuje cuando me encontraba en dificultades. Cada pedacito de este trabajo está hecho con su aliento, con las lágrimas que ha derramado por mí.

También dedico a mi hermana menor Nathaly para que este trabajo sea inspiración para que los sueños y las metas, teniendo paciencia, perseverancia, esfuerzo, trabajo duro y nunca desmaye esas ganas de cumplir sus metas.

***Andrés Vargas***



## **Agradecimiento**

En primer lugar, agradezco a Dios por cuidarme, protegerme y guiarme durante este camino académico.

Agradezco a mis padres Manuel Vargas y Lourdes Cañar, porque gracias a su apoyo incondicional, sus valores, enseñanzas y sobre todo sus consejos que han hecho conseguir esta meta en una etapa más de mi vida.

Agradezco a mis hermanos que siempre nos apoyamos en todo momento por siempre estar unidos mientras nos forjamos como profesionales.

Agradezco al Ing. Edwin Pruna por el apoyo como tutor de este trabajo, así también por sus enseñanzas en las aulas cuando fui su alumno en su momento, por compartir gratos momentos.

Agradezco a los docentes que han compartido su conocimiento y consejos durante toda la carrera universitaria para mi formación profesional.

Agradezco a Santiago, Jenyffer, Milton y José por ser parte de mi formación brindándome esa confianza de amigos, siempre apoyando a lo largo de esta vida universitaria, por compartir más allá de las aulas momentos inolvidables lleno de emociones, sonrisas, lágrimas.

De forma especial quiero agradecer a Fernando por ser un gran amigo que ha estado en las buenas y en las malas por ese apoyo fundamental para la realización de este trabajo, así como lo fue a largo de toda vida académica y sobre todo lo que compartimos fuera del aula que es lo más me llevo de él.

***Andrés Vargas***

**ÍNDICE DE CONTENIDO**

<b>Caratula .....</b>	<b>1</b>
<b>Reporte de verificación de contenido.....</b>	<b>2</b>
<b>Certificación .....</b>	<b>3</b>
<b>Responsabilidad de Autoría .....</b>	<b>4</b>
<b>Autorización de Publicación .....</b>	<b>5</b>
<b>Dedicatoria .....</b>	<b>6</b>
<b>Agradecimiento.....</b>	<b>7</b>
<b>Dedicatoria .....</b>	<b>8</b>
<b>Agradecimiento.....</b>	<b>9</b>
<b>Índice de contenido .....</b>	<b>10</b>
<b>Índice de figuras .....</b>	<b>13</b>
<b>Índice de tablas.....</b>	<b>17</b>
<b>Resumen.....</b>	<b>18</b>
<b>Abstract .....</b>	<b>19</b>
<b>Capítulo I: Introducción .....</b>	<b>20</b>
<b>Antecedentes.....</b>	<b>20</b>
<b>Planteamiento del problema y justificación .....</b>	<b>21</b>
<b>Objetivos.....</b>	<b>21</b>
<b><i>Objetivo general .....</i></b>	<b><i>21</i></b>
<b><i>Objetivos específicos.....</i></b>	<b><i>21</i></b>
<b>Hipótesis .....</b>	<b>21</b>
<b>Capítulo II: Fundamento Teórico.....</b>	<b>22</b>
<b>Automatización de procesos industriales .....</b>	<b>22</b>
<b>PLC.....</b>	<b>22</b>

Protocolo de comunicaciones Modbus TCP/IP .....	26
HMI .....	27
Diferentes tipos de lenguajes de programación en PLC .....	28
<i>Grafcet</i> .....	30
Robótica.....	31
Robot Industrial Manipulador .....	31
Configuraciones de un robot.....	32
Robot Cartesiano.....	32
Capítulo III: Desarrollo.....	33
Desarrollo de diagramas de conexión .....	33
<i>Conexión de sensores PE-RO5D</i> .....	33
<i>Conexión robot cartesiano eje x</i> .....	34
<i>Conexión PLC robot cartesiano eje z</i> .....	34
<i>Conexión de banda transportadora 1</i> .....	35
<i>Conexión de banda transportadora palet</i> .....	35
<i>Conexión PLC con elementos de tablero de control</i> .....	36
<i>Conexión de robot robotic Arm Kit con Arduino MEGA</i> .....	36
<i>Conexiones entre el panel móvil del controlador y el panel de Control</i> .....	37
<i>Conexión de red Modbus</i> .....	39
Configuración de los motores a pasos.....	39
Programación del Bloque Principal .....	43
Desarrollo de la programación (Grafcet) en modo automático.....	46
Configuración para la comunicación TCP/IP con el PLC y Arduino .....	55
Luces Indicadoras para cada sensor y actuador .....	60
Diagramas de flujo de la programación en la placa Arduino Mega .....	62
Diseño del HMI para la estación didáctica.....	67

<i>Symbol Factory</i> .....	67
<i>Desarrollo del HMI en el software InTouch</i> .....	68
<i>Comunicación del PLC con InTouch</i> .....	71
<b>Capítulo IV: Análisis de resultados</b> .....	<b>73</b>
Ejecutar pruebas de funcionamiento de la estación empleando el modo automático..	73
Resultados obtenidos del HMI.....	75
Resultados obtenidos de pruebas .....	78
<b>Capítulo V: Conclusiones y Recomendaciones</b> .....	<b>81</b>
Conclusiones.....	81
Recomendaciones.....	82
Trabajos Futuros .....	82
<b>Bibliografía</b> .....	<b>83</b>
<b>Anexos</b> .....	<b>85</b>

## ÍNDICE DE FIGURAS

<b>Figura 1</b> <i>Esquema del funcionamiento lógico y de computación de un PLC</i> .....	23
<b>Figura 2</b> <i>Memorias de un PLC</i> .....	24
<b>Figura 3</b> <i>Arduino Mega</i> .....	25
<b>Figura 4</b> <i>Ethernet Shield V1</i> .....	26
<b>Figura 5</b> <i>Red Modbus TCP / IP</i> .....	27
<b>Figura 6</b> <i>Interfaz Hombre-Maquina</i> .....	28
<b>Figura 7</b> <i>Programación Ladder</i> .....	29
<b>Figura 8</b> <i>Programación por funciones lógicas</i> .....	29
<b>Figura 9</b> <i>Etapa</i> .....	30
<b>Figura 10</b> <i>Acción asociada</i> .....	31
<b>Figura 11</b> <i>Robot manipulador</i> .....	32
<b>Figura 12</b> <i>Conexión de sensores</i> .....	33
<b>Figura 13</b> <i>Conexión controlador-driver-motor eje x</i> .....	34
<b>Figura 14</b> <i>Conexión controlador-driver-motor eje z</i> .....	34
<b>Figura 15</b> <i>Conexión controlador-driver-motor de la banda transportadora 1</i> .....	35
<b>Figura 16</b> <i>Conexión controlador-driver-motor de la banda transportadora de palets</i> .....	35
<b>Figura 17</b> <i>Conexión PLC – Panel de control</i> .....	36
<b>Figura 18</b> <i>Conexión de actuadores del robot manipulador y gripper</i> .....	37
<b>Figura 19</b> <i>Conexión plug and play</i> .....	38
<b>Figura 20</b> <i>Estructura de red Modbus TCP / IP</i> .....	39
<b>Figura 21</b> <i>Objeto tecnológico</i> .....	40
<b>Figura 22</b> <i>Adición del objeto tecnológico</i> .....	40
<b>Figura 23</b> <i>Creación del objeto tecnológico</i> .....	41
<b>Figura 24</b> <i>Configuración general del objeto tecnológico</i> .....	41
<b>Figura 25</b> <i>Configuración del accionamiento del objeto tecnológico</i> .....	42

<b>Figura 26</b> Configuración de la mecánica del objeto tecnológico .....	42
<b>Figura 27</b> Configuración de la velocidad y aceleración del motor .....	43
<b>Figura 28</b> Funciones de “Motion Control” .....	44
<b>Figura 29</b> Habilitación de las bandas transportadoras y los ejes .....	44
<b>Figura 30</b> Posicionamiento del Origen del eje Z .....	45
<b>Figura 31</b> Para de emergencia del Eje z.....	45
<b>Figura 32</b> Posicionamiento en el origen del Eje X .....	46
<b>Figura 33</b> Programación Grafcet etapa 1.....	47
<b>Figura 34</b> Etapas del grafcet en Ladder.....	48
<b>Figura 35</b> Envío de datos a las memorias para el desplazamiento de los ejes .....	48
<b>Figura 36</b> Programación para el movimiento de la banda 1.....	49
<b>Figura 37</b> Contador del Eje Z .....	49
<b>Figura 38</b> Movimiento del eje Z .....	50
<b>Figura 39</b> Programación para el posicionamiento del eje x.....	50
<b>Figura 40</b> Envío de los datos para la comunicación entre el PLC y Arduino .....	51
<b>Figura 41</b> Grafcet de la etapa 2. Transporte de los palets .....	51
<b>Figura 42</b> Programación de etapas de grafcet en Ladder.....	52
<b>Figura 43</b> Contadores para e desplazamiento de la banda.....	52
<b>Figura 44</b> Movimiento de la banda transportadora 2.....	53
<b>Figura 45</b> Diagrama Grafcet 3. Levantamiento del palet.....	53
<b>Figura 46</b> Programación de grafcet en Ladder .....	54
<b>Figura 47</b> Envío de datos a los registros.....	54
<b>Figura 48</b> Selección de bloque MB_CLIENT .....	55
<b>Figura 49</b> Creación de bloque de datos.....	56
<b>Figura 50</b> Configuración de frecuencia de registros .....	56
<b>Figura 51</b> Configuración MB_MODE .....	57

<b>Figura 52</b> <i>Escritura de MB_DATA_ADDR y MB_DATA_LEN</i> .....	57
<b>Figura 53</b> <i>Envío y recepción de datos en el bloque Datos_rec_env</i> .....	58
<b>Figura 54</b> <i>Bloque de datos para comunicación</i> .....	58
<b>Figura 55</b> <i>Configuración de Comunicación PLC-Arduino</i> .....	59
<b>Figura 56</b> <i>Envío de datos PLC hacia el Arduino</i> .....	59
<b>Figura 57</b> <i>Programación de indicadores de inicio, paro y reset</i> .....	60
<b>Figura 58</b> <i>Programación de indicadores de robot cartesiano</i> .....	61
<b>Figura 59</b> <i>Programación de indicadores bandas</i> .....	61
<b>Figura 60</b> <i>Programación de indicadores de los sensores</i> .....	62
<b>Figura 61</b> <i>Diagrama de flujo principal</i> .....	62
<b>Figura 62</b> <i>Subrutina void setup</i> .....	63
<b>Figura 63</b> <i>Subrutina Void loop</i> .....	64
<b>Figura 64</b> <i>Subrutina robot origin</i> .....	65
<b>Figura 65</b> <i>Subrutina robot palet</i> .....	66
<b>Figura 66</b> <i>Symbol Factory de InTouch</i> .....	67
<b>Figura 67</b> <i>Creación de una nueva aplicación</i> .....	68
<b>Figura 68</b> <i>HMI del proceso automatizado</i> .....	70
<b>Figura 69</b> <i>Creación del servidor OPC</i> .....	71
<b>Figura 70</b> <i>Creación de tags</i> .....	71
<b>Figura 71</b> <i>Creación de un Access Name</i> .....	72
<b>Figura 72</b> <i>Creación y enlace de los tags</i> .....	72
<b>Figura 73</b> <i>Funcionamiento del proceso de forma automática</i> .....	73
<b>Figura 74</b> <i>Proceso de llenado el del palet</i> .....	74
<b>Figura 75</b> <i>Funcionamiento del Robot manipulador organización de palet lleno</i> .....	74
<b>Figura 76</b> <i>Luces indicadoras encendidas del sensor 1 y sensor 3</i> .....	75
<b>Figura 77</b> <i>Movimiento del Robot Cartesiano en el HMI</i> .....	75

<b>Figura 78</b> Dos botellas paletizadas.....	76
<b>Figura 79</b> Comparación del Panel de Control con el HMI .....	77
<b>Figura 80</b> Supervisión de entrada y salidas de Productos .....	78



**ÍNDICE DE TABLAS**

<b>Tabla 1</b> <i>Íconos industriales vectorizados utilizados en el HMI</i> .....	69
<b>Tabla 2</b> <i>Número de envases perdidos - número de envases paletizados</i> .....	79
<b>Tabla 3</b> <i>Número de palets completos - Numero de palets incompletos</i> .....	80
<b>Tabla 4</b> <i>Datos Número de envases-número de envases paletizados</i> .....	<b>¡Error! Marcador no definido.</b>
<b>Tabla 5</b> <i>Número de Palets - Número de palets correctos</i> .....	<b>¡Error! Marcador no definido.</b>

## Resumen

El presente proyecto tiene como finalidad la automatización de un proceso didáctico de paletizado y traslación de botellas, conformado por 2 bandas transportadoras, una para la traslación de las botellas individuales y la otra para la traslación de los palet, un robot cartesiano con movimientos en el eje X y Z, para la paletización de las botellas en cada punto del palet, ambos elementos controlados por motores a paso, un robot antropomórfico de 5 GDL con un gripper como extremo operativo para la salida del palet, 4 sensores fotoeléctricos difusos para la detección de cada elemento en un punto de ubicación correspondiente, un panel de control con indicadores luminosos para conocer el estado del proceso. Para el control de todo el sistema, se usa un PLC Siemens S7-1200 con un módulo de Salidas Digitales, en donde estará la programación necesaria para la automatización de todo el proceso, por medio del uso de diagramas Grafset, y así, poder enviar las señales de control a cada uno de los actuadores por medio de la activación de cada una de las entradas del sistema.

Adicionalmente, se usa una placa Arduino MEGA para el control de movimiento del robot antropomórfico que se conecta al PLC por medio del protocolo Modbus TCP/IP, con la ayuda de una placa Shield Ethernet adaptable con Arduino. Finalmente, la implementación de un HMI en el software Wonderware Intouch para la supervisión de todo el proceso, usando las variables creadas previamente en el dispositivo controlado, con ello, evaluar la cantidad de producto completamente terminado por medio de un análisis de resultados.

*Palabras Clave:* automatización industrial, control de procesos, protocolo Modbus TCP/IP, diagramas grafset.

### **Abstract**

The aim of this project is to automate a didactic process for palletising and moving bottles, consisting of 2 conveyor belts, one for moving the individual bottles and the other for moving the pallets, a Cartesian robot with movements in the X and Z axis, for palletising the bottles at each point on the pallet, both elements controlled by stepper motors, a 5 GDL anthropomorphic robot with a gripper as the operating end for the pallet exit, 4 diffuse photoelectric sensors for the detection of each element at a corresponding location point, a control panel with light indicators to know the status of the process. For the control of the whole system, a Siemens S7-1200 PLC is used with a Digital Outputs module, where the necessary programming for the automation of the whole process will be located, by means of the use of Grafcet diagrams, and thus, to be able to send the control signals to each one of the actuators by means of the activation of each one of the inputs of the system. Additionally, an Arduino MEGA board is used for the motion control of the anthropomorphic robot which is connected to the PLC via the Modbus TCP/IP protocol with the help of an Arduino adaptive Ethernet Shield board. Finally, the implementation of an HMI in the Woderware Intouch software for the supervision of the entire process, using the variables previously created in the controlled device, thereby evaluating the amount of completely finished product by means of a results analysis.

*Keywords:* industrial automation, process control, Modbus TCP/IP protocol, grafcet diagrams.

## Capítulo I

### Introducción

#### Antecedentes

La automatización de procesos industriales es de gran importancia en la actualidad, las tareas tediosas y peligrosas son realizadas generalmente por autómatas programables, aumentando la producción, productividad, eficiencia y calidad en las plantas industriales a nivel mundial.

Los sistemas de entrenamiento en automatización industrial son herramientas muy útiles en el proceso de enseñanza aprendizaje en automatización industrial, permitiendo a los usuarios realizar prácticas de laboratorio en sistemas a escala, similares a los diferentes procesos industriales existentes.

La estación didáctica es una herramienta que aporta a los estudiantes un enfoque de enseñanza estructurado y eficiente. Siguiendo un método didáctico, esta estación proporciona información de manera clara y concisa, permitiendo a los estudiantes adquirir conocimientos fundamentales de manera efectiva. Al implementar tecnologías y sistemas de automatización y supervisión, la estación didáctica optimiza el proceso de transporte y organización de los envases de bebidas, asegurando su flujo continuo y controlado. Esto no solo mejora la eficiencia y calidad del proceso, sino que también brinda a los estudiantes una experiencia de aprendizaje práctica y aplicada. A través de la estación didáctica, los estudiantes tienen la oportunidad de interactuar con tecnologías avanzadas y adquirir habilidades relevantes para el campo industrial. En resumen, la estación didáctica enriquece la educación al proporcionar un entorno de aprendizaje interactivo y tecnológico.

## **Planteamiento del problema y justificación**

La carrera de Ingeniería en Electrónica y Automatización de la Universidad de las Fuerzas Armadas ESPE Sede Latacunga actualmente no dispone de equipamiento adecuado para la enseñanza de las nuevas asignaturas en el área de Automatización Industrial, lo que dificulta el proceso de enseñanza aprendizaje práctico. Por lo mencionado es indispensable el contar con estaciones didácticas para la automatización de procesos industriales.

## **Objetivos**

### ***Objetivo general***

Automatizar y supervisar la estación didáctica del transporte y organización de envases para bebidas.

### ***Objetivos específicos***

- Diseñar la solución de automatización para el proceso de transporte y organización de envases para bebidas, mediante diagramas Grafcet.
- Programar el PLC, para el funcionamiento en modo automático de las etapas de la estación didáctica del transporte y organización de envases para bebidas.
- Desarrollar un HMI, para el monitoreo y supervisión de la estación didáctica, para la automatización del transporte y organización de envases para bebidas.
- Desarrollar los diagramas eléctricos de conexión de los equipos de automatización.
- Realizar pruebas de funcionamiento en modo automático del sistema de automatización del transporte y organización de envases para bebidas.

## **Hipótesis**

¿La automatización y supervisión de la estación didáctica para el transporte y organización de envases para bebidas, permitirá el funcionamiento en modo automático de la estación didáctica?

## Capítulo II

### Fundamento Teórico

#### Automatización de procesos industriales

La automatización industrial se define como el uso de equipos de tecnología (Software y hardware), sistemas computarizados, electromecánicos, electroneumáticos y electrohidráulicos que tengan la capacidad de llevar todo tipo de proceso de producción de manera automática es decir con la mínima intervención humana posible. (Adsrobotics, 2020)

#### PLC.

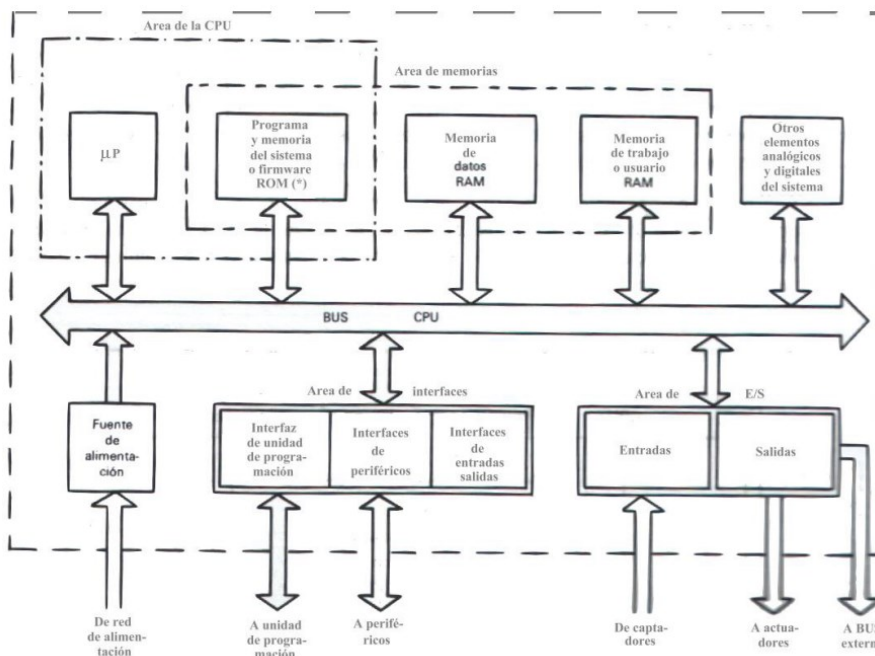
##### Controlador

Controlador Lógico Programable (PLC), es un dispositivo de control industrial, fundamentado en una computadora que usa instrucciones de programación para tomar acciones de encendido y apagado, esto para sustituir conexiones de lógica cableada por medio de relevadores (Martínez, 2015). La utilización se da en aquellas instalaciones donde se implementa o realiza maniobra, control, señalización, etc. Aplicado a procesos de fabricación industrial de cualquier tipo de transformaciones de materia prima en productos. Los PLC fueron diseñados como un dispositivo de reemplazo industrial, pero hoy se emplea en cualquier tipo de aplicación cubriendo las necesidades de los usuarios. Los PLCs están configurados de forma compacta, así como modularmente para poder expandirse con la posibilidad de satisfacer las necesidades de la industria en futuras expansiones. (Vallejo, 2018)

## Estructura Interna de un Controlador Lógico Programable

Figura 1

Esquema del funcionamiento lógico y de computación de un PLC.



*Nota.* La Figura 1 detalla la estructura interna de un PLC para el procesamiento de sus operaciones clasificada por áreas y periféricos del hardware. Obtenido de (Martínez, 2015).

### Unidad Central de Procesos (CPU)

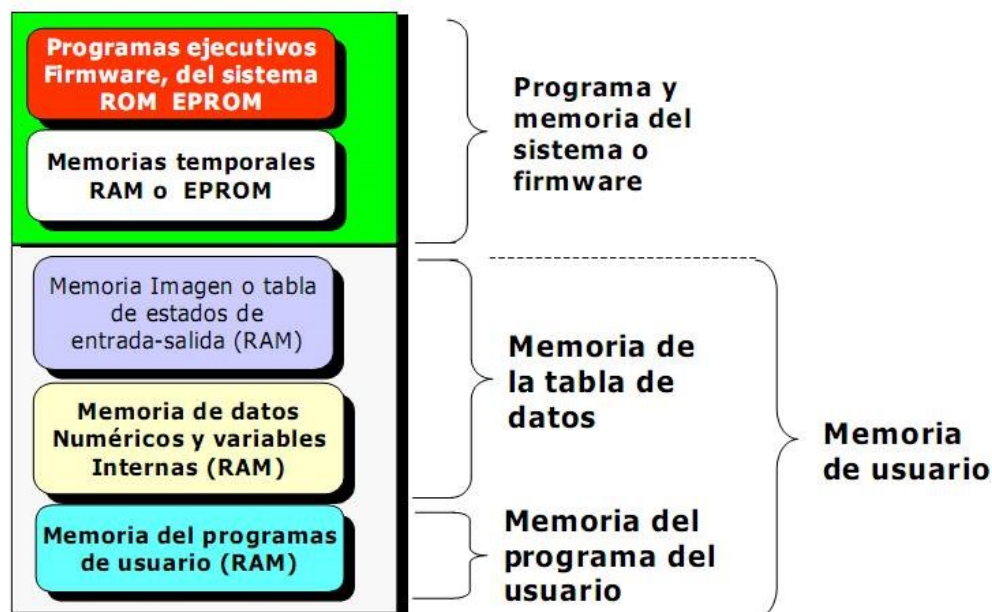
Esta unidad se encarga de recibir las órdenes de usuario por medio de una consola de programación y el módulo de entradas, la cual procesa y envía respuestas al módulo de salida. En la memoria se encuentra cargado el programa destinado a controlar el proceso.

### Memoria del PLC

Los PLC tienen la capacidad de almacenar y retirar información para lo cual cuentan con memorias. El sistema operativo como el programa de aplicación, las tablas o registros de variables o en bits internos están relacionados con las memorias.

**Figura 2**

Memorias de un PLC



*Nota.* La Figura 2 se detalla la configuración de memorias dentro de un PLC. Obtenido de (Anderson, 2018).

### Entradas y Salidas

Las entradas y salidas es la parte que se encarga de vincular el campo con el controlador programable, su función es recibir las señales de los transmisores o sensores para ser procesados por la CPU en el caso de las entradas, y dar órdenes en las salidas. (Pablo, 2008)

### Tarjeta Arduino Mega

El Arduino Mega 2560 es una placa microcontroladora basada en el ATmega2560. Tiene 54 pines de entrada/salidas digitales (de los cuales 15 se pueden usar como salidas PWM), 16 entradas analógicas, 4 UART (puertos serie de hardware), un oscilador de cristal de 16 MHz, una conexión USB, un conector de alimentación, un encabezado ICSP, y un botón de reinicio. Contiene todo lo necesario para soportar el microcontrolador; simplemente conéctelo a una computadora con un cable USB o enciéndalo con un adaptador de CA a CC o una batería para





**Figura 4***Ethernet Shield V1*

*Nota.* esta figura representa la conexión por los pines de shield ethernet y un Arduino Mega.

Obtenido de (Sainsmart, 2024)

**Protocolo de comunicaciones Modbus TCP/IP**

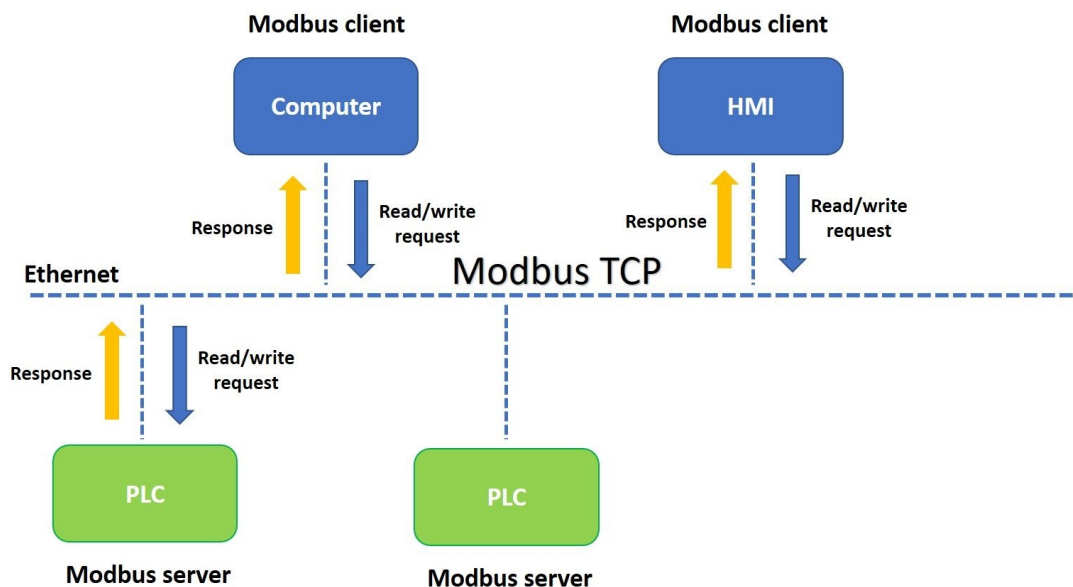
Es el protocolo Modbus RTU con interfaz TCP que se ejecuta en Ethernet, la estructura de envío y recepción de datos Modbus es el protocolo de aplicación que define las normas para la organización e interpretación de los datos sin tomar el tipo de medio de transmisión de datos.

El protocolo TCP / IP permite a los bloques de datos binarios que se intercambien entre ordenadores como se observa en la Figura 5, la función de TCP es asegurar que todos los paquetes de datos se reciben correctamente y que IP asegure que los mensajes se aborden correctamente.

Cabe recalcar que el protocolo TCP / IP es solo un protocolo de transporte, y no define que significa los datos o como los datos se ha de interpretar, la interpretación le corresponde al protocolo de aplicación es decir Modbus. ( QualiaTech Corp Company, 2021)

**Figura 5**

*Red Modbus TCP / IP*



*Nota.* esta figura representa la comunicación Modbus TCP entre varios dispositivos como PLC, computadora HMI. Obtenido de (Plcynergy, 2023)

## HMI

Un sistema HMI viene de las siglas en ingles de Interfaz Humano-Máquina, se describe como una ventana que tiene funciones como monitorización, supervisión, alarmas y control de un proceso, estos paneles o software están, ya se sea en un computador o pantallas desarrollado por los diversos fabricantes un ejemplo didáctico se puede observar en la Figura 6 (Vestertraining, 2024).

En caso de control es la capacidad de ajustar ciertas variables de proceso dentro de los límites que cada variable permite, pues bien, el operador puede modificar, cambiar, ajustar en

el HMI, pero a la vez está enviando los valores ajustados a un controlador donde realiza las operaciones programadas para cumplir el pedido del operador.

Las señales del proceso son conducidas hacia los HMIs mediante tarjetas electrónicas de entradas y salidas en la en los computadores, controladores lógicos programables (PLCs), RTU o DRIVERS, todos estos deben tener comunicación en HMI y los dispositivos mencionados. (Marcia Carrion, 2012)

## Figura 6

### *Interfaz Humano-Maquina*



## Diferentes tipos de lenguajes de programación en PLC

Para poder programar en el PLC existen diferentes tipos de lenguaje de programación, el cual consiste en instrucciones ordenadas (disponibles del PLC) que resuelven una tarea de control.

En los lenguajes de programación del PLC se pueden clasificar de la siguiente manera:

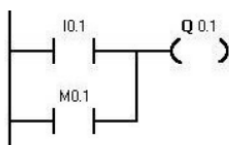
- Informáticos:
  - Bajo Nivel: Lista de instrucciones.
  - Alto Nivel.

- Gráficos:
  - Diagramas de escalera (Ladder).
  - Flujo gramas.
  - Diagramas de Funciones lógicas.
  - Grafcet.

El lenguaje de programación que tiene base las especificaciones del personal de mantenimiento de los antiguos armarios de relés pudieran operar el Controlador Lógico Programable es el lenguaje de programación Ladder. Un ejemplo claro de dicha programación se puede observar en la Figura 7 obtenida de (Arrieta Arvilla & Gonzales, 2003)

### Figura 7

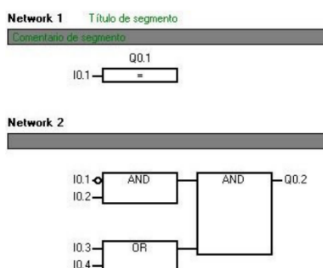
*Programación Ladder*



Otra forma de programar a los PLCs, como son los diagramas de funciones lógicas tiene base en la electrónica digital, en donde se pueden encontrar bloques de programación que hacen referencia a compuertas lógicas, temporizadores, contadores, comparadores, etc. Un ejemplo claro de programación con diagramas de funciones lógicas se puede observar en la Figura 8 obtenida de (Arrieta Arvilla & Gonzales, 2003)

### Figura 8

*Programación por funciones lógicas.*



## **Grafcet**

Este tipo de programación se basa en un modelado de sistemas autónomos secuenciales. La programación Grafcet no busca minimizar las funciones lógicas que modelan la dinámica del sistema. Una de las cualidades que mejor representa a la programación Grafcet, es que es una herramienta poderosa y sobre todo metodológica. (Arrieta Arvilla & Gonzales, 2003)

Las características que ofrece la programación Grafcet son las siguientes:

- Una programación TOP DOWN (programación descendente), es el desarrollo a través de diversas etapas que abarca desde lo general hasta llegar a lo particular.
- Introduce el concepto de diseño estructurado, en donde las diversas etapas del control autónomo tienen un cierto orden, mediante el forzado de eventos de mayor orden en modelos Grafcet. (Arrieta Arvilla & Gonzales, 2003)

Para el desarrollo de esta programación se debe tener en cuenta los siguientes conceptos y elementos gráficos:

- **Etapas:** es aquella situación del sistema, la cual forma parte de todo o una sola parte del control, la cual no depende de entradas y las salidas del proceso. (Arrieta Arvilla & Gonzales, 2003)

La Figura 9 es la representación de la etapa en programación Grafcet.

### **Figura 9**

*Etapas*

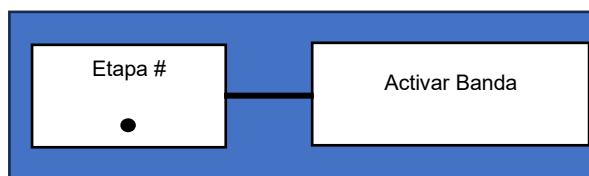


- **Acción asociada:** este elemento se enfoca en las operaciones que se van a realizar, siempre y cuando la etapa este activada. Las acciones que se van a realizar, se conectan en el lado derecho con la figura de un rectángulo, dentro del rectángulo debe estar indicado las operaciones que se van a realizar de manera literal o simbólica. (Arrieta Arvilla & Gonzales, 2003).

Como se puede observar en la Figura 10.

### Figura 10

*Acción asociada*



### Robótica

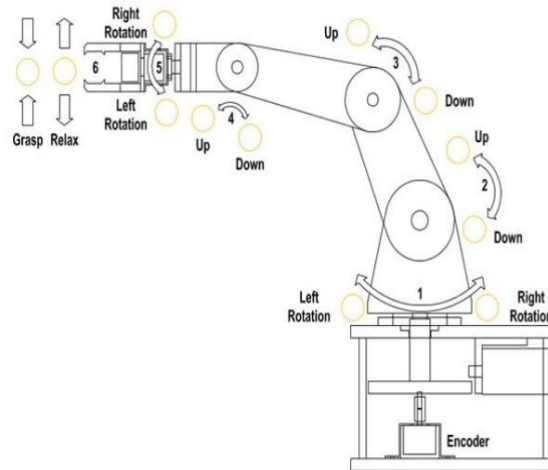
#### Robot Industrial Manipulador

La definición de robótica según la RIA (Asociación de Industrias Robóticas) es un manipulador multifuncional reprogramable, capaz de mover materias, piezas, herramientas o dispositivos especiales, según trayectorias variables, programadas para realizar tareas diversas.

La definición por la IFR (Federación Internacional de Robótica) en su informe técnico ISO/TR 83737 (septiembre 1988) distingue entre robot industrial de manipulación se entiende a una máquina de manipulación automática, reprogramable y multifuncional con tres o más ejes que puede proporcionar y orientar materias, piezas, herramientas o dispositivos especiales para la ejecución de trabajos diversos en las diferentes etapas de la producción industrial, ya sea en una posición fija o en movimiento. (Barrietos A., 1997)

## Figura 11

### Robot manipulador



*Nota.* La Figura 11 es la representación estructural de un robot manipulador. Obtenido de (DE LORENZO OF AMERICA CORP. S.A)

## Configuraciones de un robot

### Robot Cartesiano

Este tipo de robots presentan una estructura articulada, es decir se constituye por elementos o eslabones que facilitan el movimiento el cual puede ser de traslación vertical, horizontal y transversal los cuales se puede combinar con movimiento de giro sobre el mismo eje.

Todo robot sin importar el tipo de configuración se debe tener en cuenta los grados de libertad (GDL) que significa el movimiento independiente en un espacio tridimensional pueden realizar una articulación. La especificación del punto se efectúa mediante coordenadas cartesianas (x, y, z) y la precisión es uniforme en todo el espacio de trabajo. (López J., 2007)



## Capítulo III

### Desarrollo

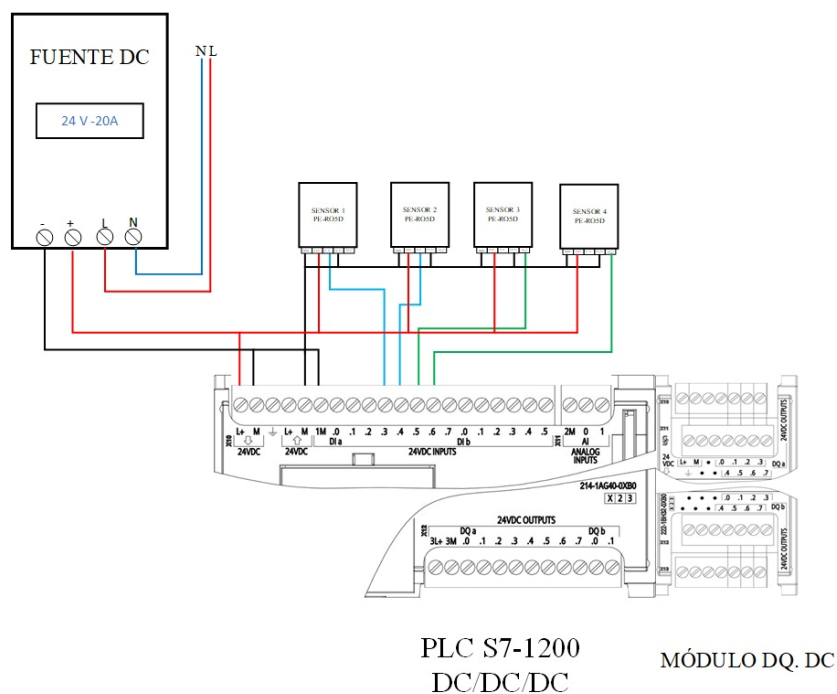
#### Desarrollo de diagramas de conexión

#### Conexión de sensores PE-RO5D

En la Figura 12 se muestra la conexión de los equipos de medición PE-RO5D con el controlador, el sensor 1 y sensor 2 está configurado normalmente abierto conectado a la entrada 3 y 4 respectivamente del controlador, el sensor 3 y sensor 4 está configurado normalmente cerrado conectado a la entrada 5 y 6 respectivamente.

**Figura 12**

Conexión de sensores

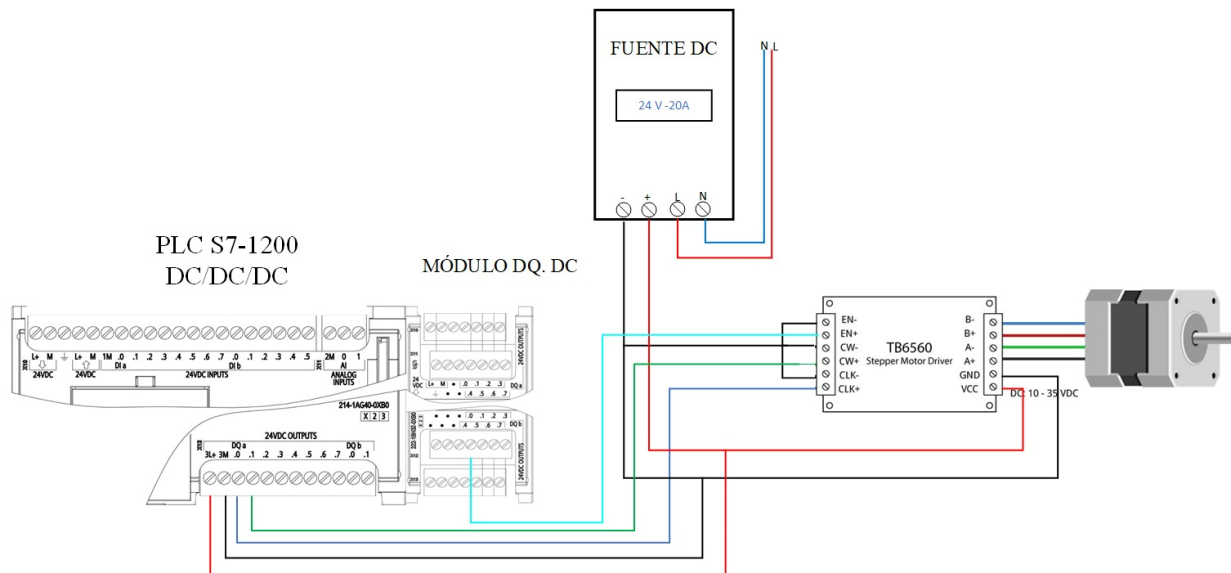


En la Figura 13 se observa la conexión del motor Nema 17 con un PLC siemens 1200 DC/DC/DC mediante el driver TB6560 donde el CLK+ y CW+, está conectado con la salida Q0.0 y Q0.1 respectivamente, el EN+ conectado a la salida QB.0 del módulo DQ 222.

### Conexión robot cartesiano eje x

Figura 13

Conexión controlador-driver-motor eje x

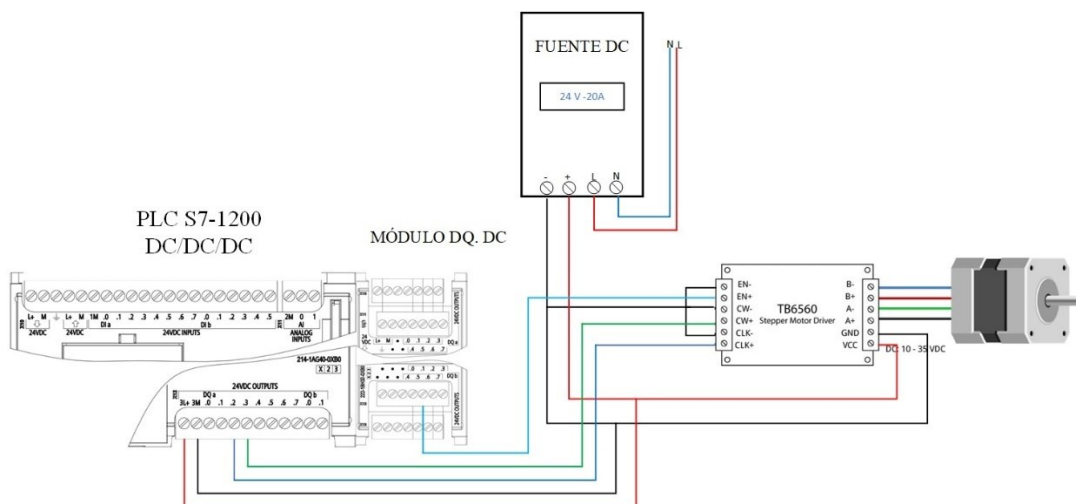


En la Figura 14 se observa la conexión del motor Nema 17 con un PLC siemens 1200 DC/DC/DC mediante el driver TB6560 donde el CLK+ y CW+, está conectado con la salida Q0.2 y Q0.3 respectivamente, el EN+ conectado a la salida QB.1 del módulo DQ 222.

### Conexión PLC robot cartesiano eje z

Figura 14

Conexión controlador-driver-motor eje z

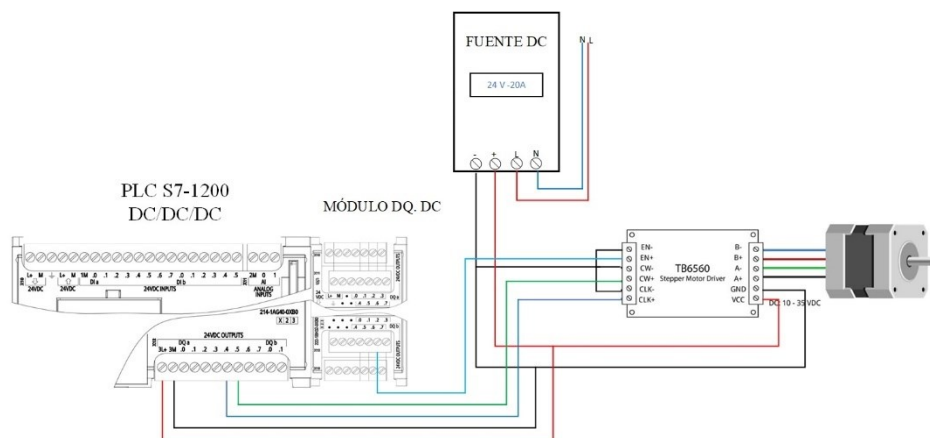


En la Figura 15 se observa la conexión del motor Nema 17 con un PLC siemens 1200 DC/DC/DC mediante el driver TB6560 donde el CLK+ y CW+, está conectado con la salida Q0.4 y Q0.5 respectivamente, el EN+ conectado a la salida QB.2 del módulo DQ 222.

### **Conexión de banda transportadora 1**

**Figura 15** Conexión controlador-driver-motor de la banda transportadora 1

*Conexión controlador-driver-motor de la banda transportadora 1*



En la Figura 16 se observa la conexión del motor Nema 17 con un PLC siemens 1200 DC/DC/DC mediante el driver TB6560 donde el CLK+ y CW+, está conectado con la salida Q0.6 y Q0.7 respectivamente, el EN+ conectado a la salida QB.3 del módulo DQ 222.

### **Conexión de banda transportadora palet**

**Figura 16**

*Conexión controlador-driver-motor de la banda transportadora de palets*

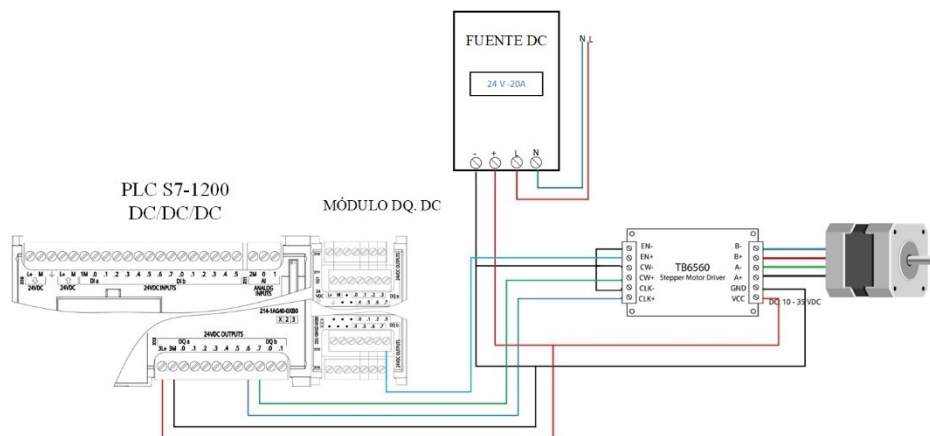
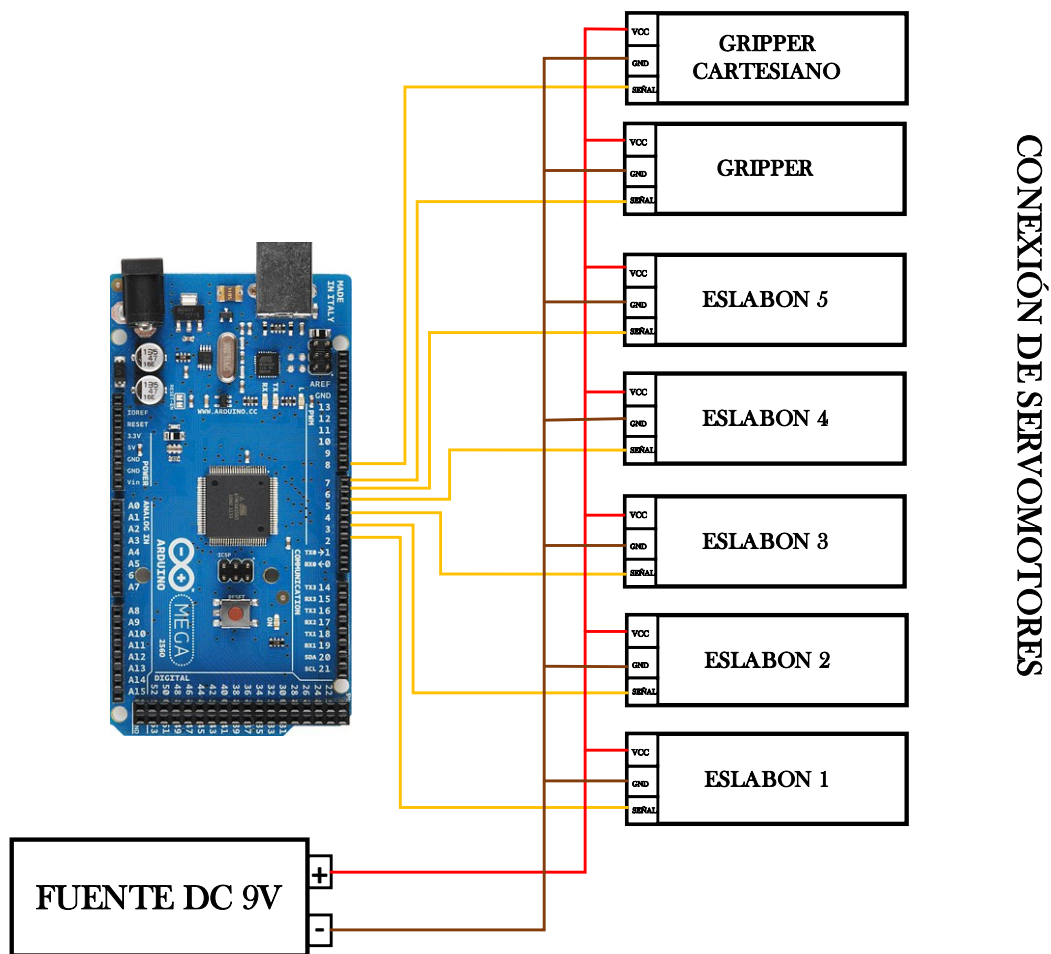




Figura 18

Conexión de actuadores del robot manipulador y gripper

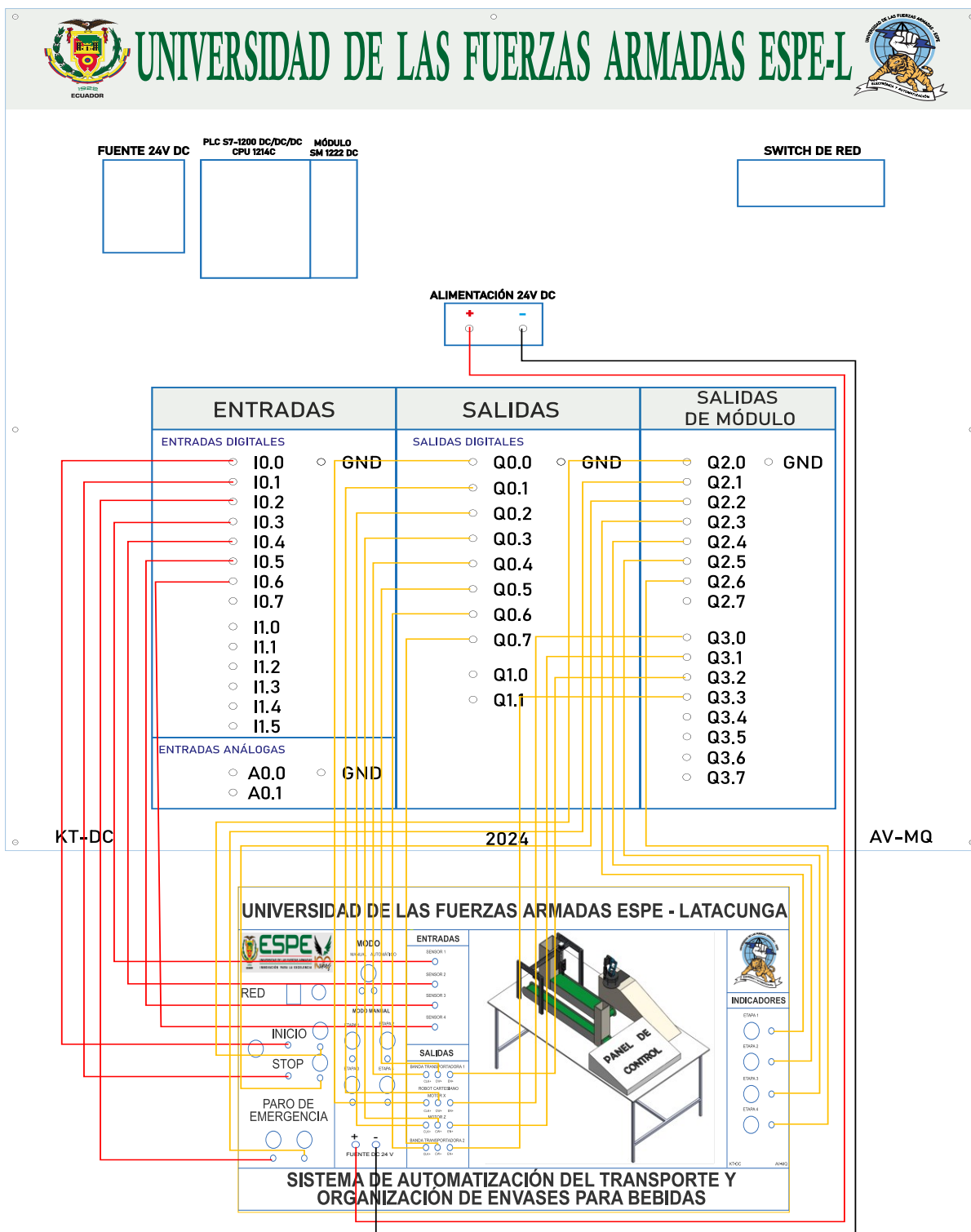


### Conexiones entre el panel móvil del controlador y el panel de Control

Con la Figura 19 se presenta un diagrama conexiones para el correcto funcionamiento del modo automático del proceso, las conexiones entre el Panel de control y el panel móvil de entradas y salidas del controlador es mediante cables con terminales tipo banana.

Figura 19

Conexión plug and play

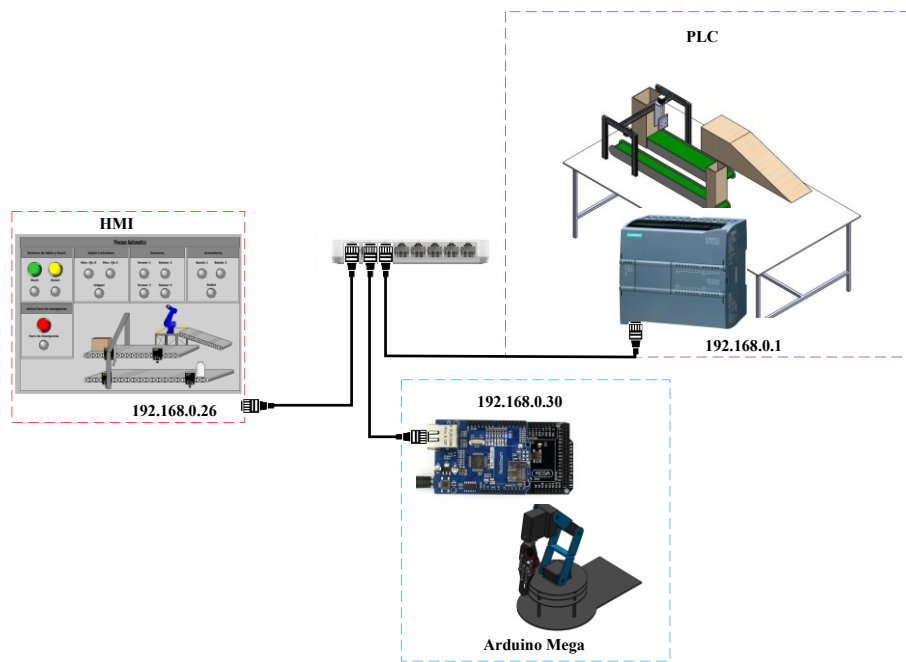


### Conexión de red Modbus

La Figura 20 representa la conexión mediante terminales ethernet la conexión entre controlador (PLC), Arduino Mega y HMI (PC laptop) con la ayuda de switch marca D-Link de uso ofimático de 8 puertos, se permite el uso de este tipo de equipo al ser una estación didáctica.

### Figura 20

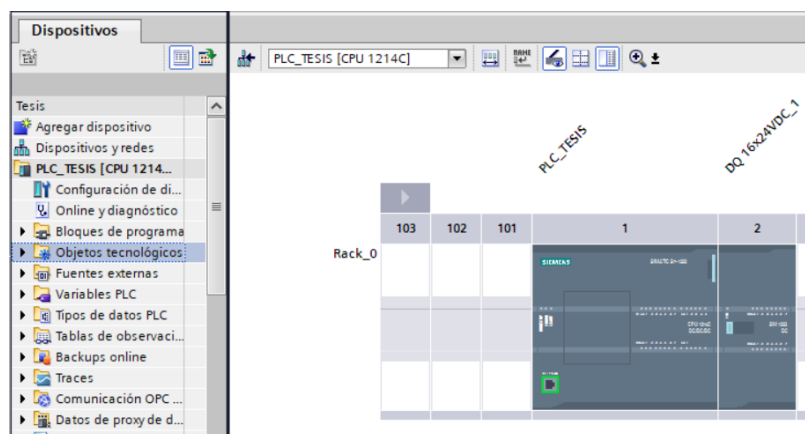
*Estructura de red Modbus TCP / IP*



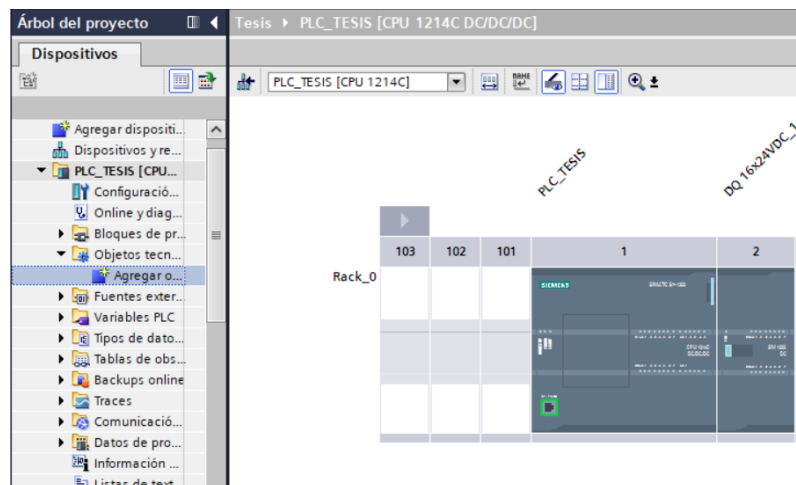
### Configuración de los motores a pasos.

Para el control de los motores a pasos en modo automático, primero realizar, la creación de objetos tecnológicos dentro de la programación del PLC, los pasos que se deben seguir son los siguientes.

1. Una vez agregado el PLC en el proyecto, seleccionar la opción de objetos tecnológicos como se muestra la Figura 21.

**Figura 21***Objeto tecnológico*

2. Dentro de esa opción agregar el objeto como se muestra en la Figura 22.

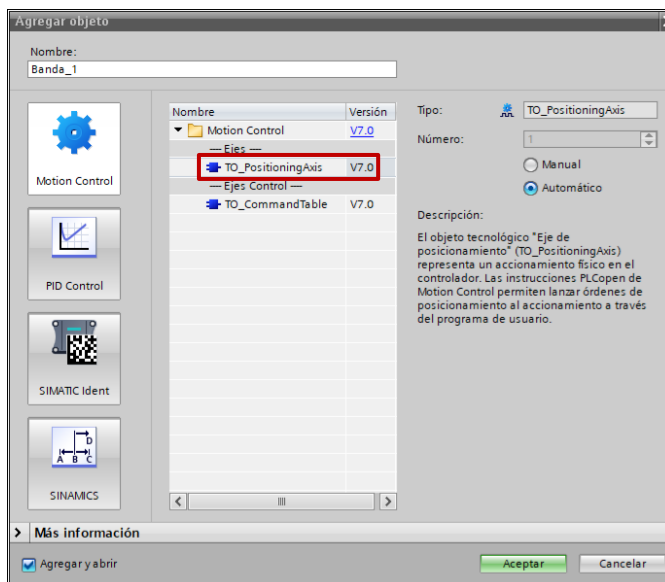
**Figura 22***Adición del objeto tecnológico*

3. Después de hacer clic en agregar objeto, se muestra la siguiente pantalla, donde se asigna un nombre al objeto tecnológico y en el lado izquierdo se elige la opción de Motion Control. Dentro de dicha opción, se encuentran dos alternativas adicionales, y para configurar los motores se selecciona la primera opción, tal como se indica en la Figura 23.



Figura 23

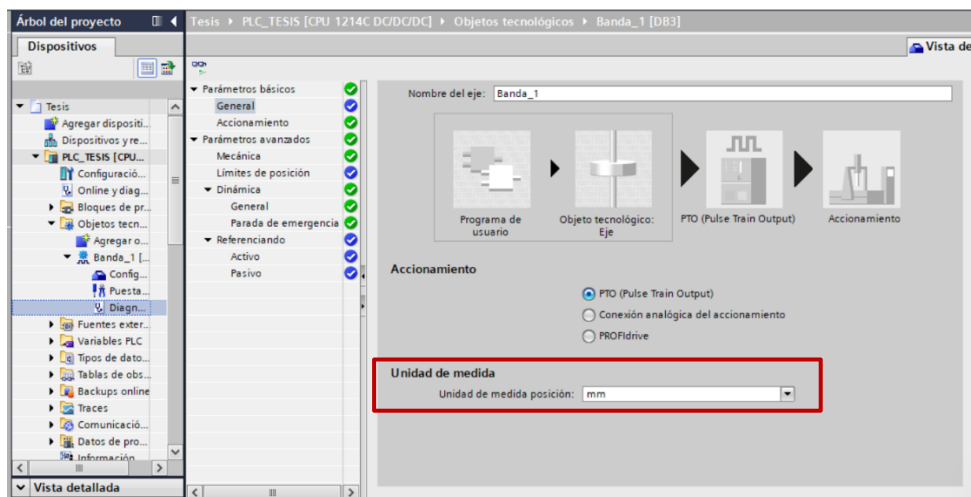
Creación del objeto tecnológico



- Una vez creado el objeto tecnológico, se procede a dirigirse a la configuración del mismo. En la sección “general”, se elige la unidad de medida con la que se controlarán los motores a pasos, según se muestra en la Figura 24. Para el caso en cuestión, la unidad será en milímetros (mm).

Figura 24

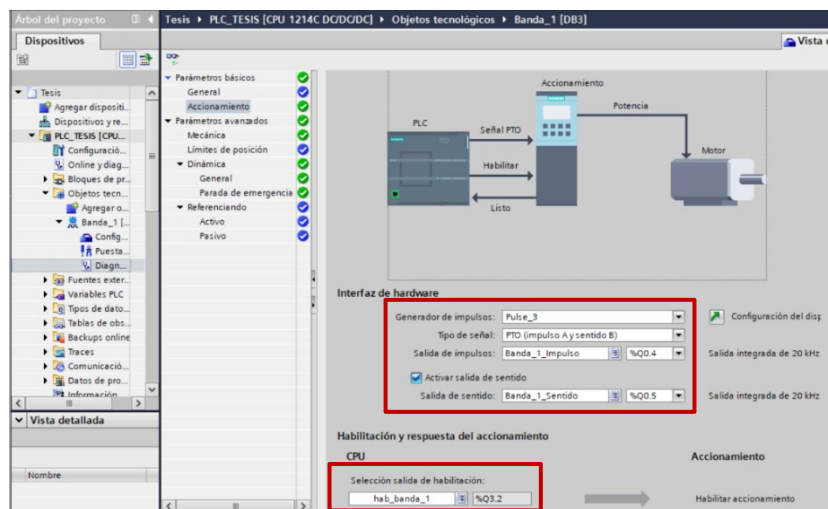
Configuración general del objeto tecnológico



En el apartado de accionamiento, se elige la salida de las señales de control (tren de pulsos, dirección y habilitación del motor), tal como se indica en la Figura 25.

**Figura 25**

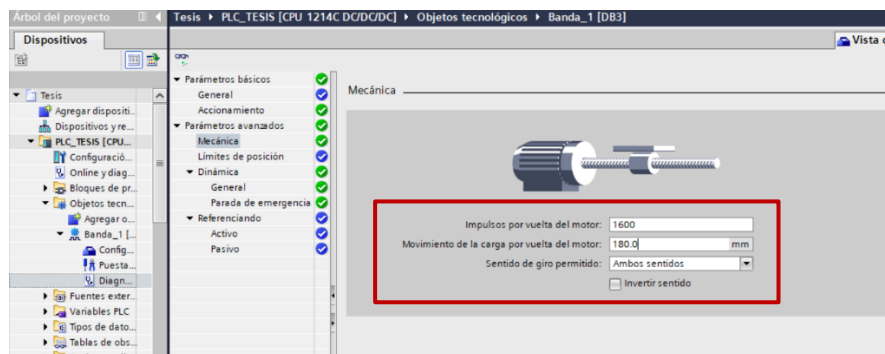
*Configuración del accionamiento del objeto tecnológico*



5. Por último, en el apartado de mecánica, en el campo de impulsos por vuelta se introduce el valor de los pulsos que emite el motor a pasos divididos por el número de micro pasos proporcionados por el controlador. En este caso, el valor a utilizar en dicho campo es de 1600. En el espacio de movimiento de la carga por vuelta del motor, se indica cuántos milímetros se desplaza en una vuelta completa del motor. Todos los detalles mencionados anteriormente se pueden observar en la Figura 26.

**Figura 26**

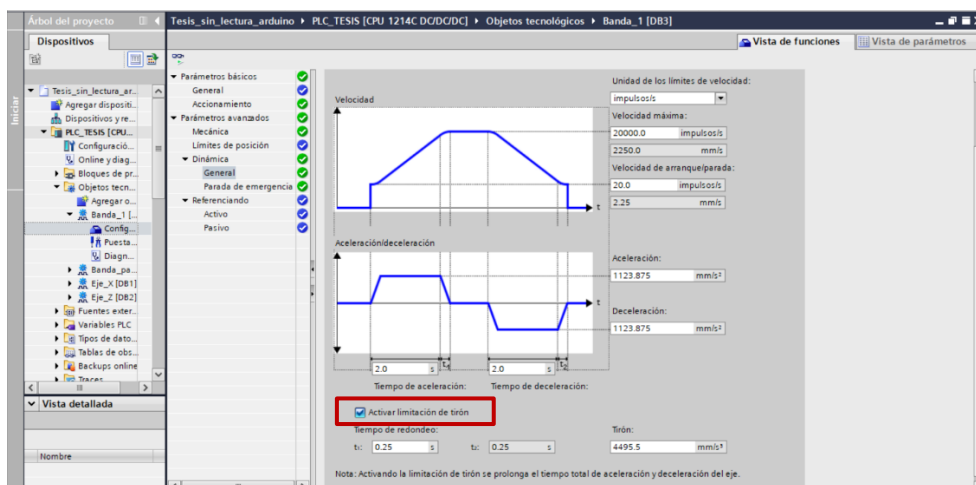
*Configuración de la mecánica del objeto tecnológico*



- En la opción dentro de "Parámetros avanzados", y en "Dinámica", se elige la opción de General. Dentro de esta opción, se puede ajustar la velocidad máxima a la que puede operar el motor. Además, se selecciona la opción de "Activar limitación de tirón", la cual permite realizar una aceleración/desaceleración progresiva del motor, evitando cambios abruptos de velocidad. Todo lo mencionado anteriormente se puede observar en la Figura 27.

**Figura 27**

*Configuración de la velocidad y aceleración del motor*



- Se repite el proceso nuevamente para el motor del Eje x, motor del Eje Z y para el motor que lleva las botellas y para el motor del palet.

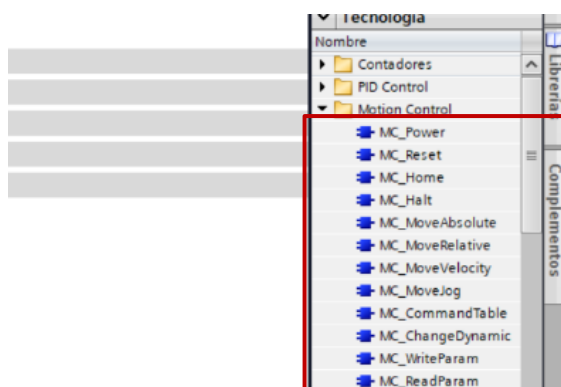
## Programación del Bloque Principal

Una vez realizada la configuración de todos los objetos tecnológicos, se procede a realizar el programa autónomo del proceso, siguiendo los pasos que se presentan a continuación

- En primera instancia se inicializa los motores dentro del bloque de función "Inicio\_motores"; las funciones utilizadas en el bloque antes mencionado, se encuentran en la carpeta "Motion Control" ubicada en la opción de "Tecnología" como se muestra en la Figura 28.

Figura 28

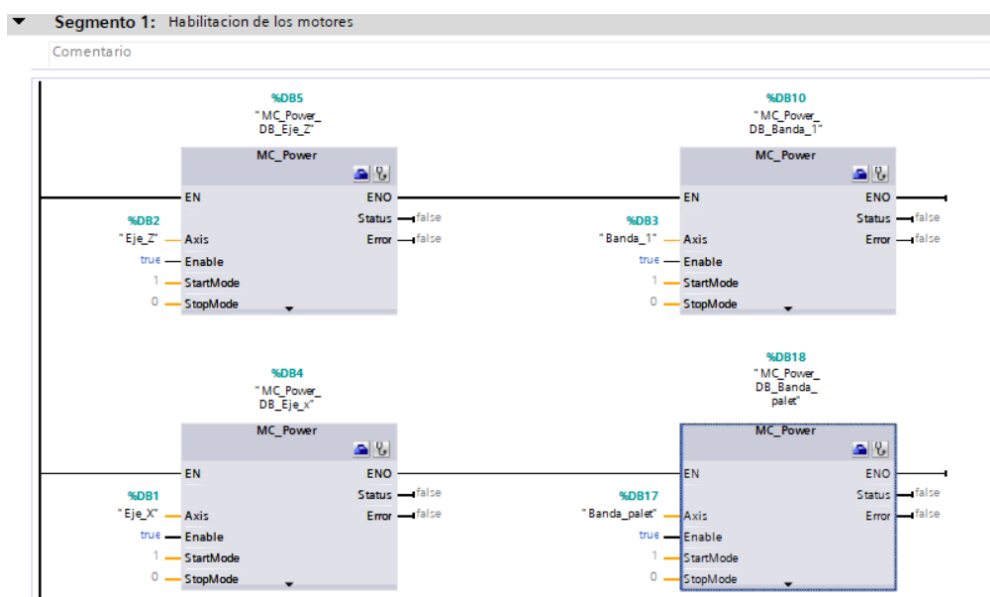
Funciones de "Motion Control"



2. Con las funciones mencionadas anteriormente, se procede a escoger la función "MC\_HOME" para habilitar el accionamiento de los motores seleccionando el nombre de las bandas y de los ejes en parámetro "Axis" (Figura 29) el cual se va a ejecutar al momento de arrancar el PLC.

Figura 29

Habilitación de las bandas transportadoras y los ejes

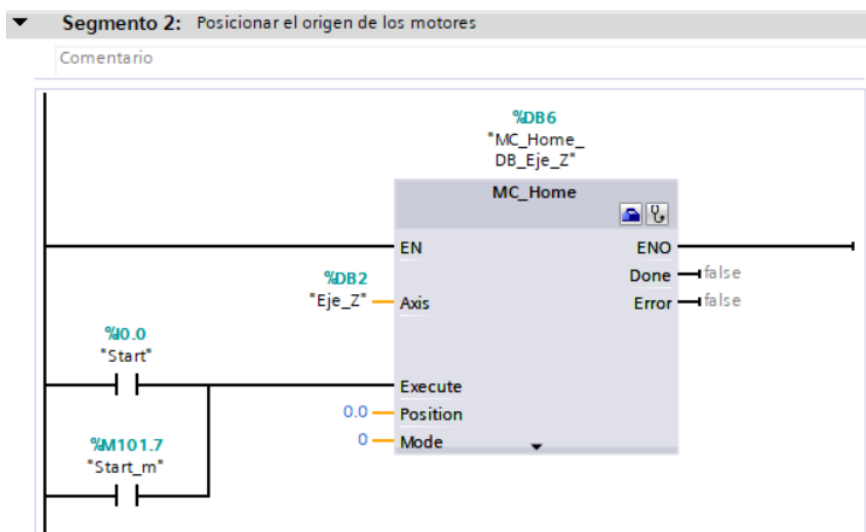


3. Para posicionar los motores en el origen 0, se va a utilizar la función "MC\_Home", el cual se va a ejecutar cuando se presione el botón de Inicio del panel de control, o al

botón de Inicio del HMI (Figura 30), este procedimiento se realiza para los 4 objetos tecnológicos creados.

**Figura 30**

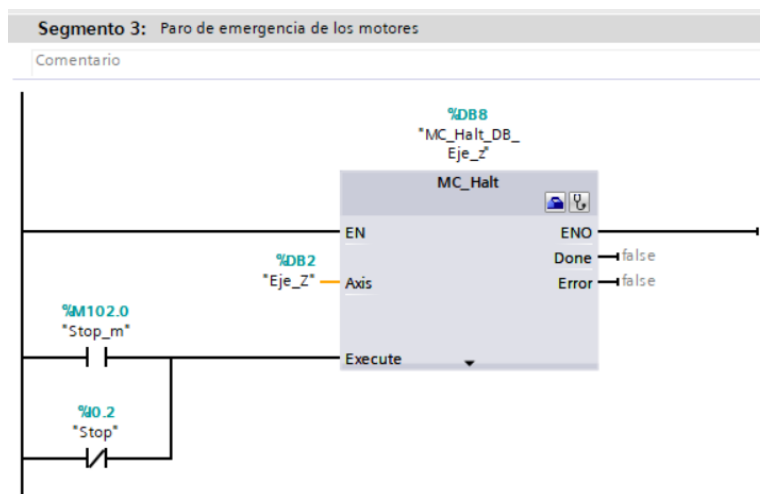
*Posicionamiento del Origen del eje Z*



4. Para el Paro de emergencia de los motores, se utilizará la función “MC\_Halt”, los cuales van a ser accionados mediante el botón de paro de emergencia del panel de control o el botón de paro de emergencia del HMI (Figura 31), este procedimiento se realiza para los 4 objetos tecnológicos creados.

**Figura 31**

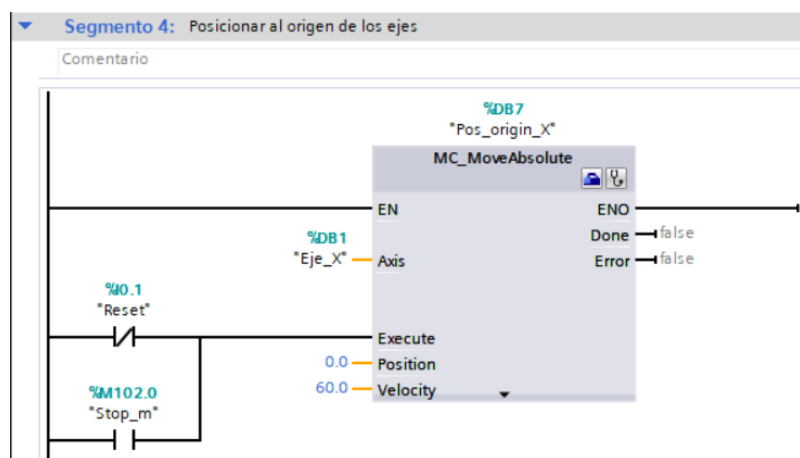
*Para de emergencia del Eje z*



- Para posicionar los ejes en el origen, se va a utilizar la función “MC\_MoveAbsolute” la cual se va a ejecutar al momento de presionar el botón de reset del panel de control o el botón de reset del HMI (Figura 32) este procedimiento se repite para el eje Z.

**Figura 32**

*Posicionamiento en el origen del Eje X*

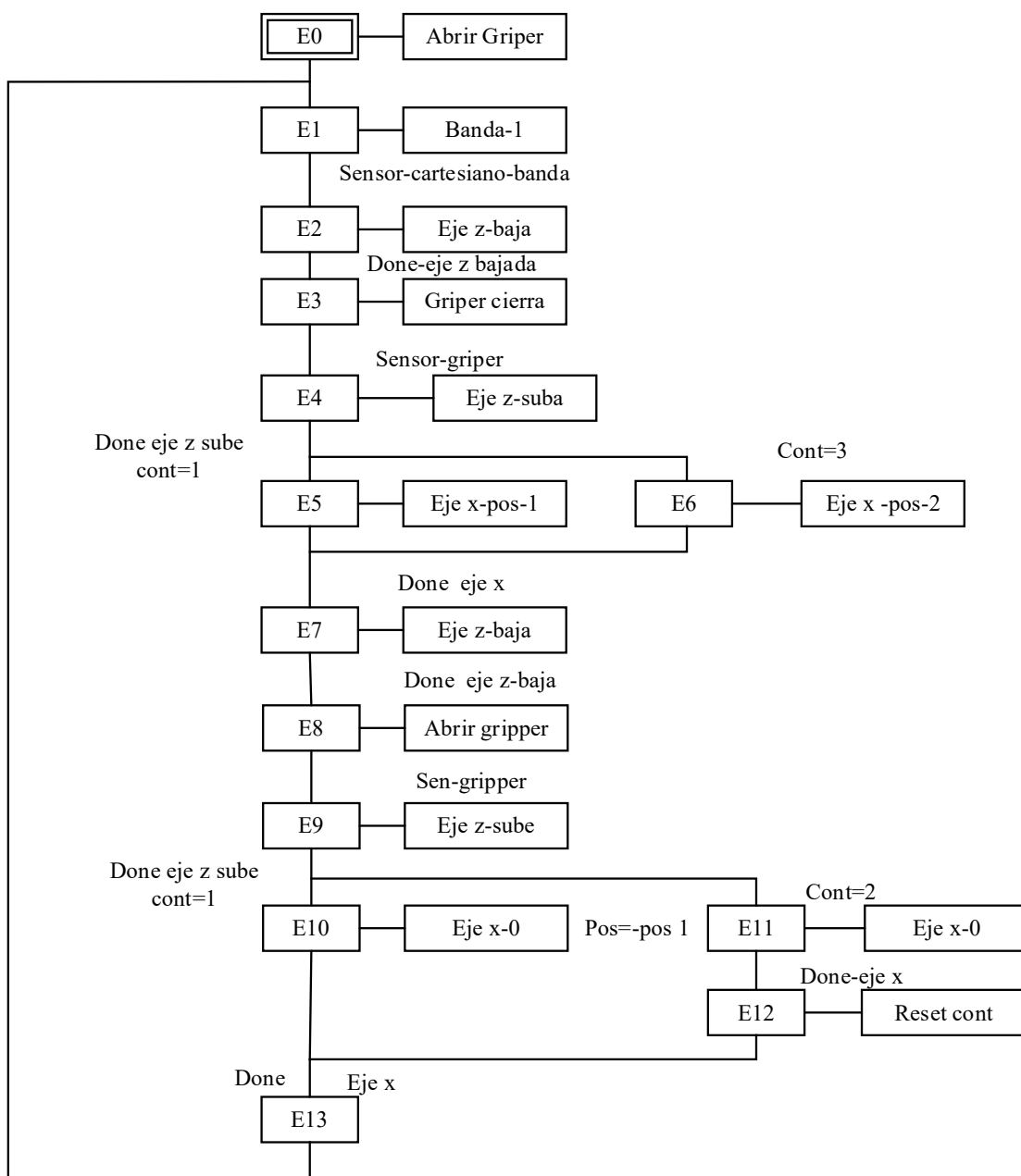


### **Desarrollo de la programación (Grafcet) en modo automático.**

Una vez hecha la configuración de los motores se procede a realizar la programación del modo automático, por medio de la lógica de programación Grafcet. El programa se divide en 3 etapas, la Figura 33 indica la primera etapa, la cual, se concentra en la traslación de las botellas desde el punto inicial de la banda transportadora hasta llegar al robot cartesiano, el cual, se encarga de posicionar las botellas en dos diferentes posiciones en el eje x, la segunda etapa (Figura 41) se concentra en el transporte de los palets desde un punto inicial hasta el lugar de posicionamiento del eje x para la colocación de las botellas dentro de los mismo. Por último, la tercera etapa (Figura 45), el levantamiento del palet mediante un robot manipulador antropomórfico de 5GDL.

Figura 33

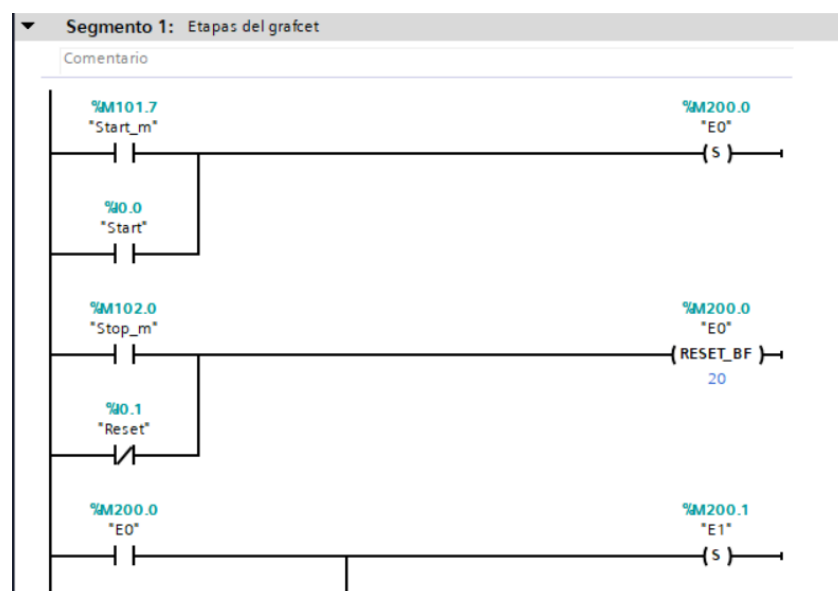
## Programación Grafcet etapa 1



1. En el segmento 1 (Figura 34) del bloque de programación llamado "Aut\_Banda1" se encuentra la programación en Ladder por etapas.

**Figura 34**

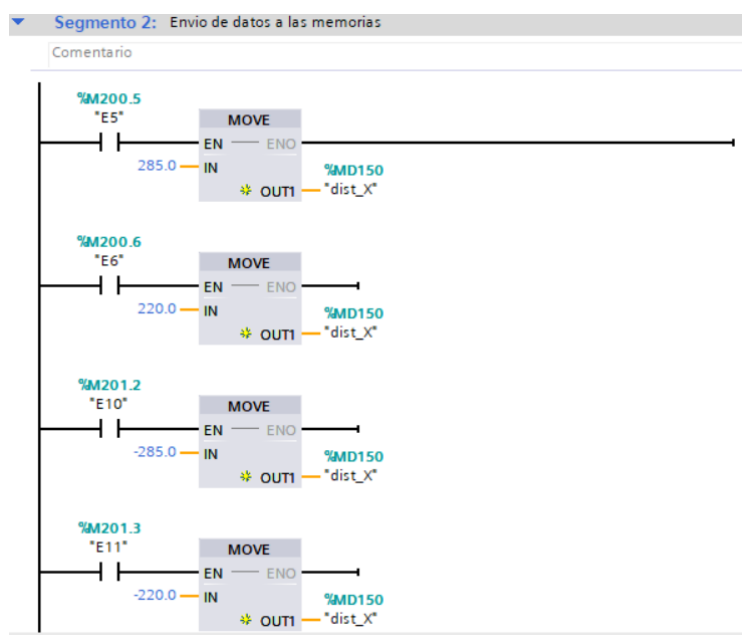
*Etapas del grafcet en Ladder*



2. En el segmento 2 (Figura 35) se encuentra el envío de los datos necesarios hacia las marcas del PLC, para generar el movimiento de los ejes.

**Figura 35**

*Envío de datos a las memorias para el desplazamiento de los ejes*

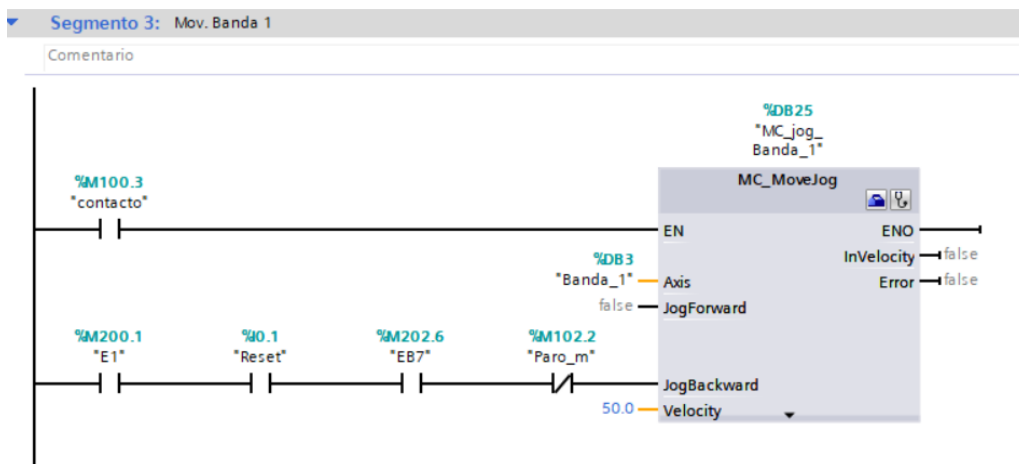




- En el segmento 3 (Figura 36) se encuentra la programación para el movimiento de la banda 1.

**Figura 36**

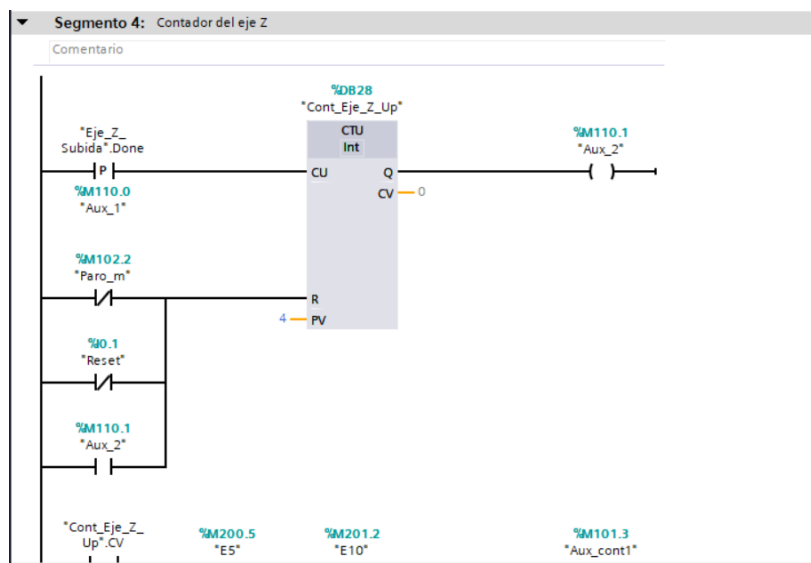
*Programación para el movimiento de la banda 1.*



- En el segmento 4 (Figura 37) se encuentra un contador para determinar la cantidad de veces que se eleva el eje Z, el cual se utiliza para activar diferentes etapas dependiendo el número de veces que se levantó el eje Z.

**Figura 37**

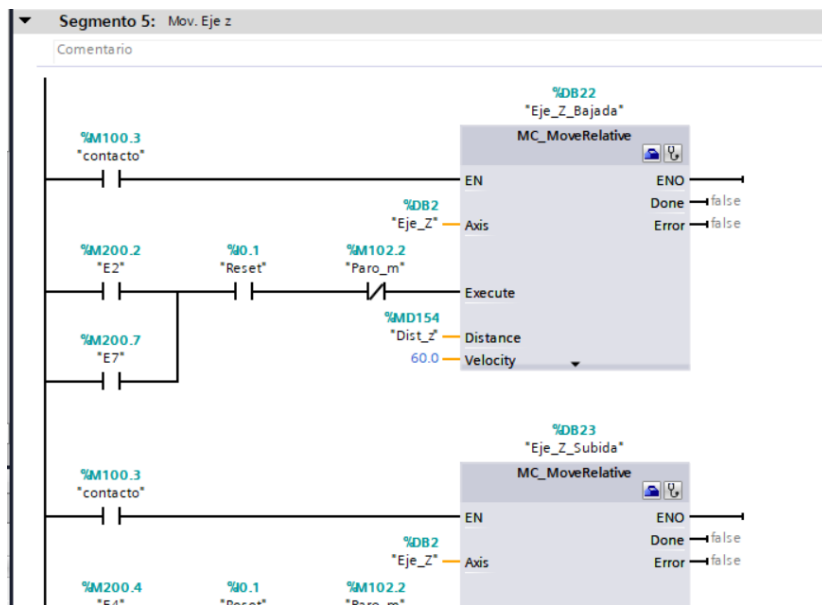
*Contador del Eje Z*



5. En el segmento 5 (Figura 38) se encuentra la programación para el movimiento del eje Z en cada etapa que sea necesario.

**Figura 38**

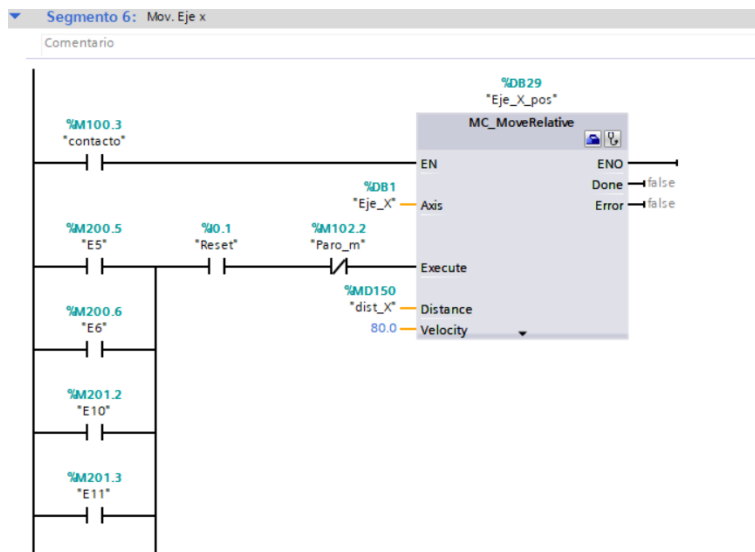
*Movimiento del eje Z*



6. En el segmento 6 (Figura 39) se encuentra la programación para el posicionamiento del eje x en las diferentes etapas en las que se necesita.

**Figura 39**

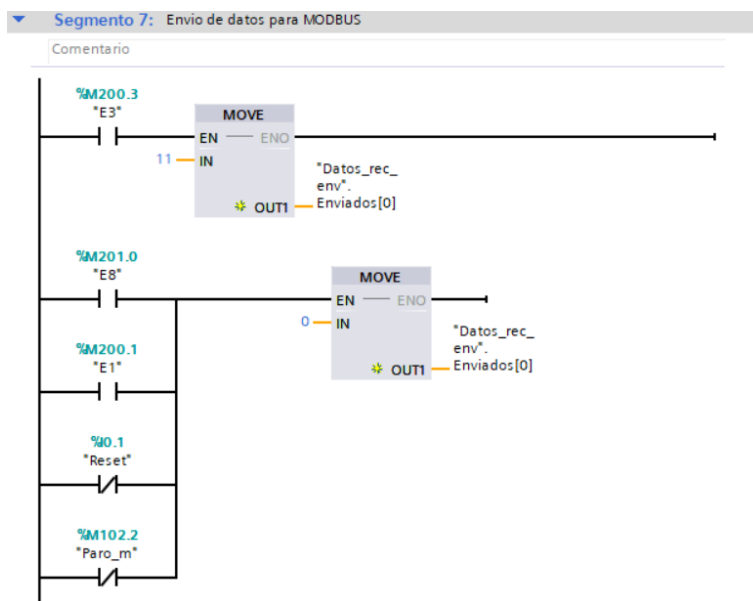
*Programación para el posicionamiento del eje x*



7. En el segmento 7 (Figura 40) se encuentra el envío de los datos para la comunicación entre el PLC y Arduino mediante comunicación Modbus TCP/IP.

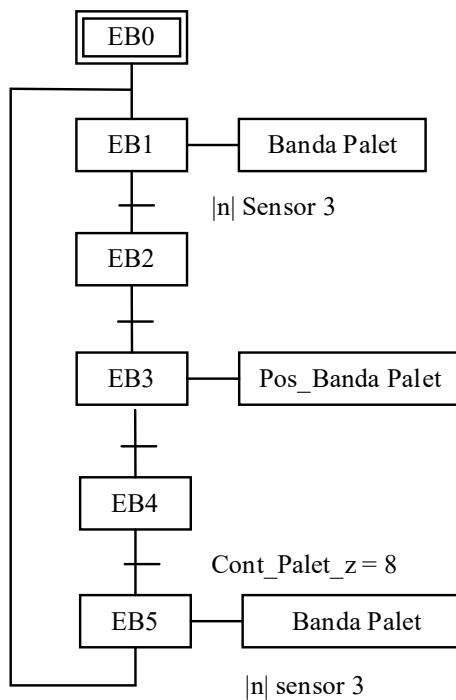
**Figura 40**

*Envío de los datos para la comunicación entre el PLC y Arduino*



**Figura 41**

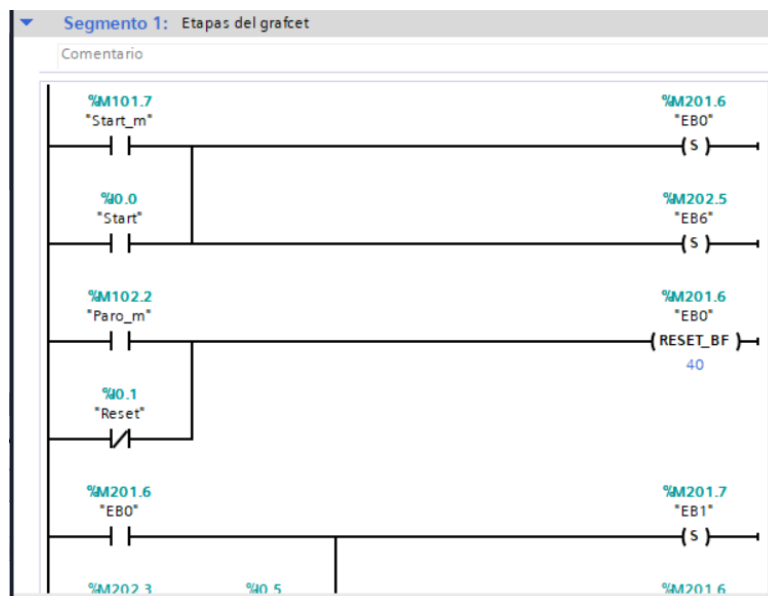
*Diagrama graficet 2. Transporte de los palets*



1. En la Figura 42 se ilustra la programación en lenguaje Ladder correspondiente a cada etapa del diagrama grafcet presentado en la figura anterior en el software Tia Portal

**Figura 42**

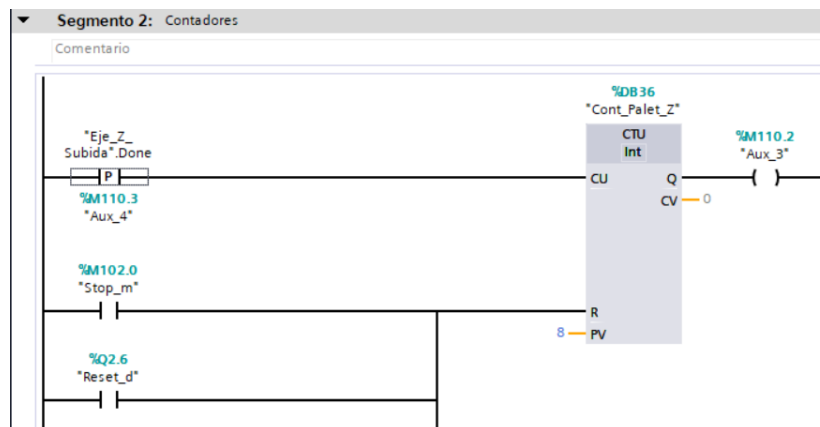
*Programación de etapas de grafcet en Ladder*



2. En la Figura 43 se presente la programación del contador necesario para poder cuantificar la cantidad de palets presentes en la banda correspondiente a la etapa 5 del diagrama grafcet presentado en la Figura 41.

**Figura 43**

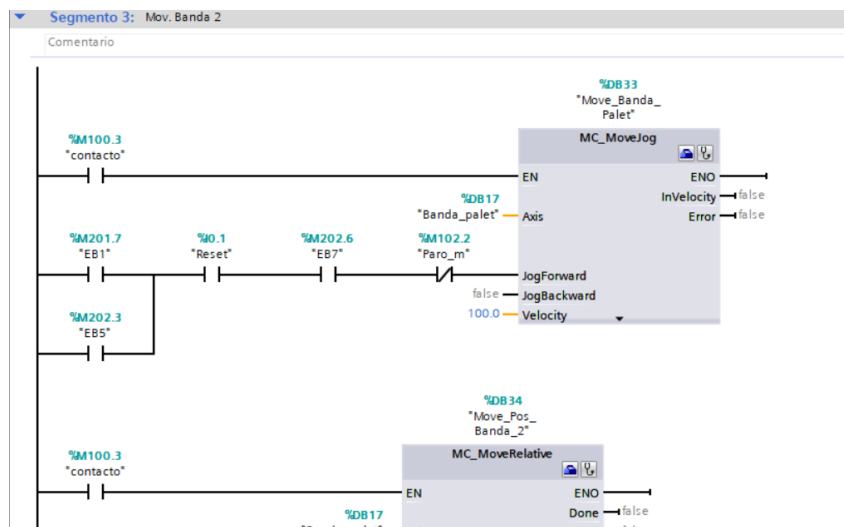
*Contadores para el desplazamiento de la banda*



3. La ilustración de la Figura 44 presenta la programación necesaria para generar el movimiento en el motor correspondiente a la banda transportadora 2.

### Figura 44

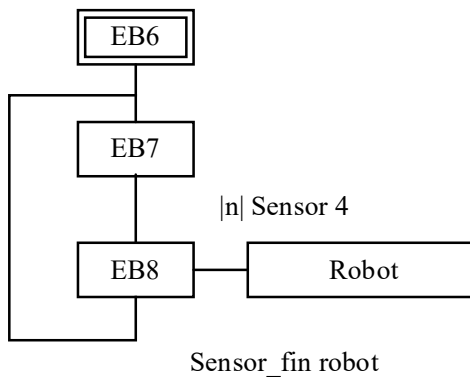
#### Movimiento de la banda transportadora 2



Con esto, finaliza la programación del diagrama grafcet de la etapa 2. Posteriormente, se procede a realizar la programación requerida para el diagrama grafcet 3, presentado en la Figura 45.

### Figura 45 Diagrama Grafcet 3. Levantamiento del palet

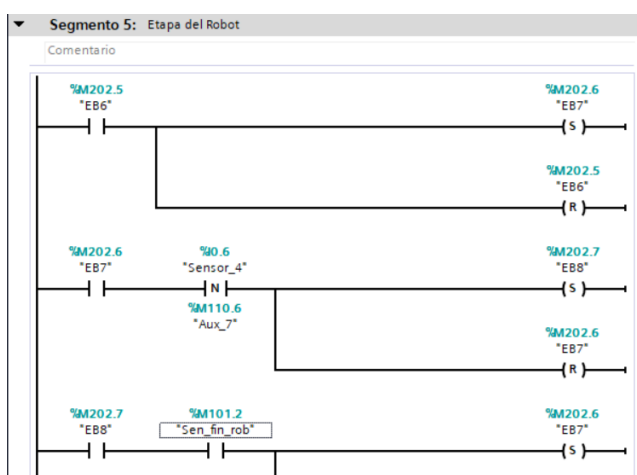
#### Diagrama Grafcet 3. Levantamiento del palet



1. En primera instancia, se procede a programar en lenguaje Ladder las condiciones necesarias para pasar de una etapa a otra del diagrama grafcet ilustrado anteriormente, como se indica en la Figura 46.

**Figura 46**

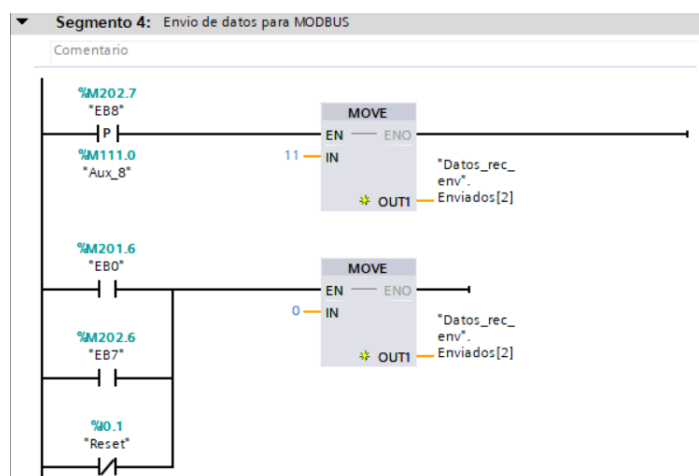
*Programación de grafcet en Ladder*



2. En la Figura 47 se presenta la programación necesaria para el envío de datos hacia los registros en la comunicación Modbus TCP/IP.

**Figura 47**

*Envío de datos a los registros*



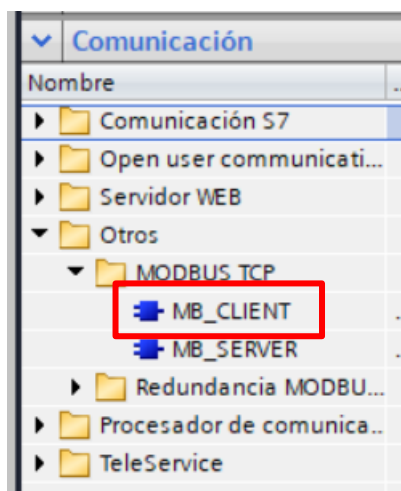
## Configuración para la comunicación TCP/IP con el PLC y Arduino

Para poder manipular la apertura del extremo operativo que se encuentra en el robot cartesiano, y poder mover dicho robot, es necesario controlar los servomotores que actúan como actuadores controlados por la placa Arduino, dicho control de movimiento se encuentra dentro de la programación de la placa antes mencionada. Los pasos a seguir para poder establecer la comunicación entre estos dos dispositivos son los siguientes:

1. La función que se va a utilizar para poder establecer la comunicación Modbus TCP/IP se llama "MB\_CLIENT"; la cual se encuentra en la carpeta "Otros", la carpeta se encuentra dentro de la opción de "Comunicación". Como se puede ver en la Figura 48.

### Figura 48

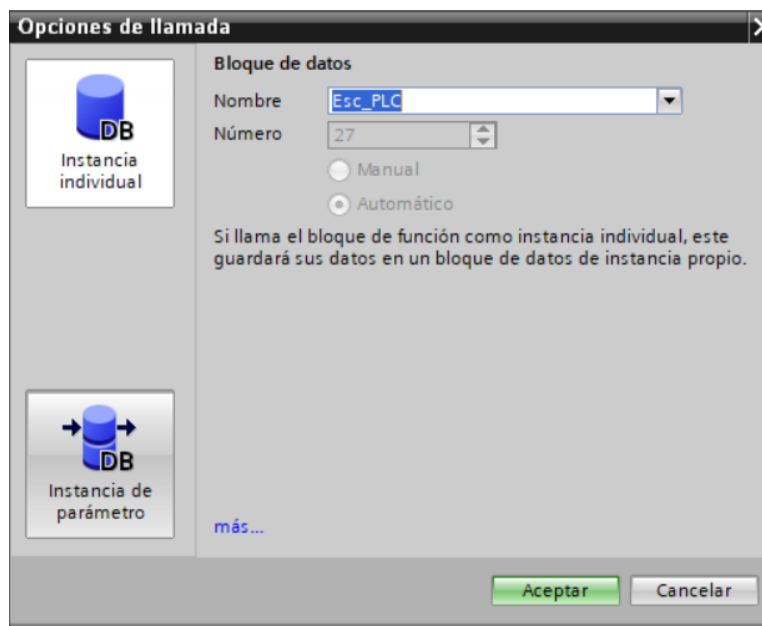
*Selección de bloque MB\_CLIENT*



2. Una vez insertado la función, dentro del bloque de programación "Com\_PLC\_Ard", se pone de nombre al bloque de datos "Esc\_PLC" representado en la Figura 49.

Figura 49

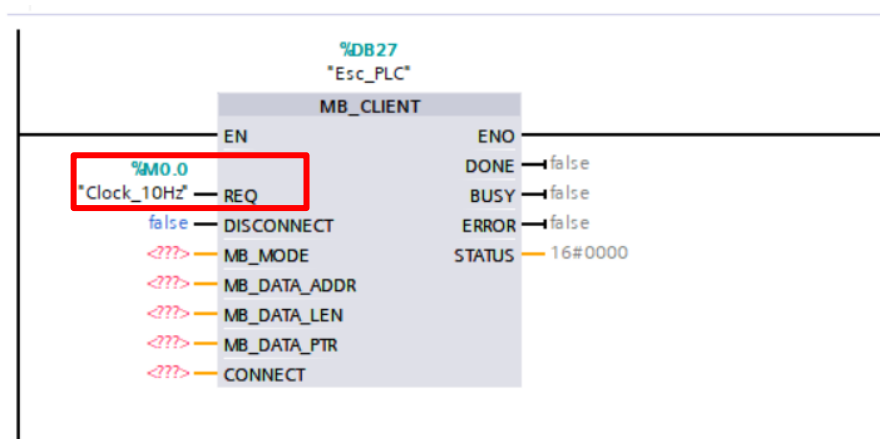
Creación de bloque de datos



- Ya listo con la función, se procede a realizar la configuración del bloque, la cual va estar siempre mandando la solicitud de escritura de registros cada 100 ms como se muestra en la Figura 50.

Figura 50

Configuración de frecuencia de registros

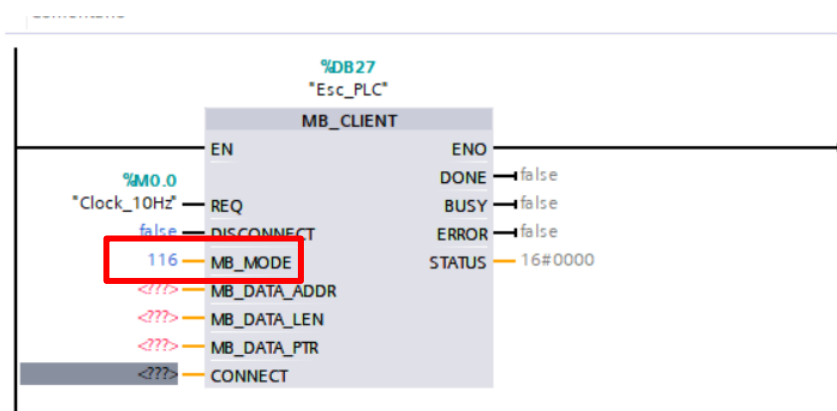




4. Para poder realizar la escritura de los registros se coloca el numero 116 dentro de la opción MB\_MODE (el modo 116, ayuda a escribir registros de parada, desde el registro 0 hasta 65.535); lo anteriormente mencionado se muestra en la Figura 51.

**Figura 51**

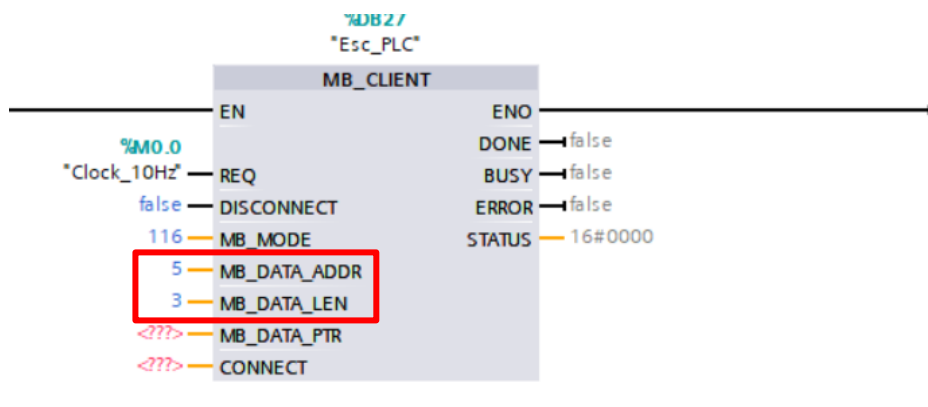
*Configuración MB\_MODE*



5. Para indicar desde que numero de registro y hasta que registro vamos a escribir desde el PLC, se rellenan en las dos siguientes opciones como se muestra en la Figura 52.

**Figura 52**

*Escritura de MB\_DATA\_ADDR y MB\_DATA\_LEN*



6. Para la escritura de los registros se crea un bloque de datos con el nombre de “Datos\_rec\_env”, dichos datos van a ser de tipo enteros como se observa en la Figura 53.

**Figura 53**

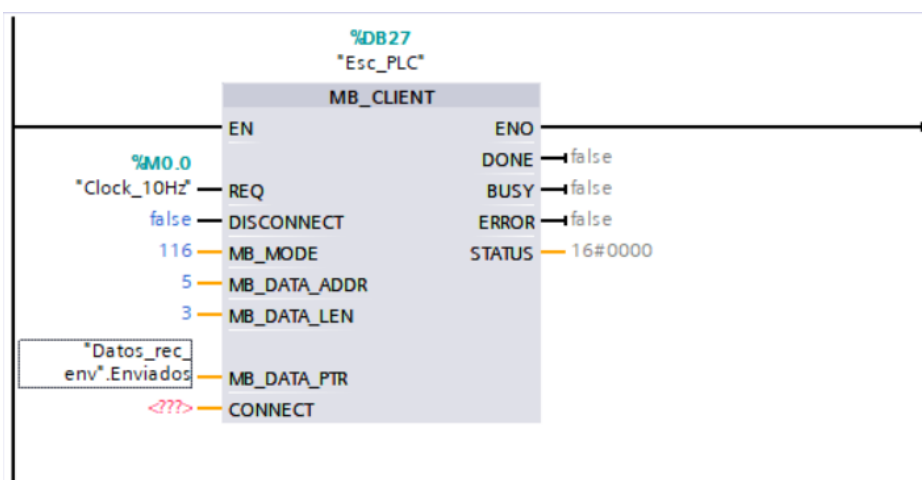
*Envió y recepción de datos en el bloque Datos\_rec\_env*

Datos_rec_env									
	Nombre	Tipo de datos	Valor de arranq...	Remanen...	Accesible d...	Escrib...	Visible en ..	Valor de a..	Cor
1	Static			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
2	Recibidos	Array[0..9] o...		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
3	Enviados	Array[0..9] of Int		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
4	Enviados[0]	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
5	Enviados[1]	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
6	Enviados[2]	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
7	Enviados[3]	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
8	Enviados[4]	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
9	Enviados[5]	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
10	Enviados[6]	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
11	Enviados[7]	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
12	Enviados[8]	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
13	Enviados[9]	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	

7. Una vez creado el bloque, dicho bloque de datos se coloca en la siguiente opción para la configuración de la comunicación como se observa en la Figura 54.

**Figura 54**

*Bloque de datos para comunicación*



8. Para saber a qué servidor escribir los registros se crea un bloque de datos llamado “Config\_Com\_PLC\_Ard” Figura 55 la cual se encuentran todas las configuraciones necesarias para establecer la comunicación.

**Figura 55**

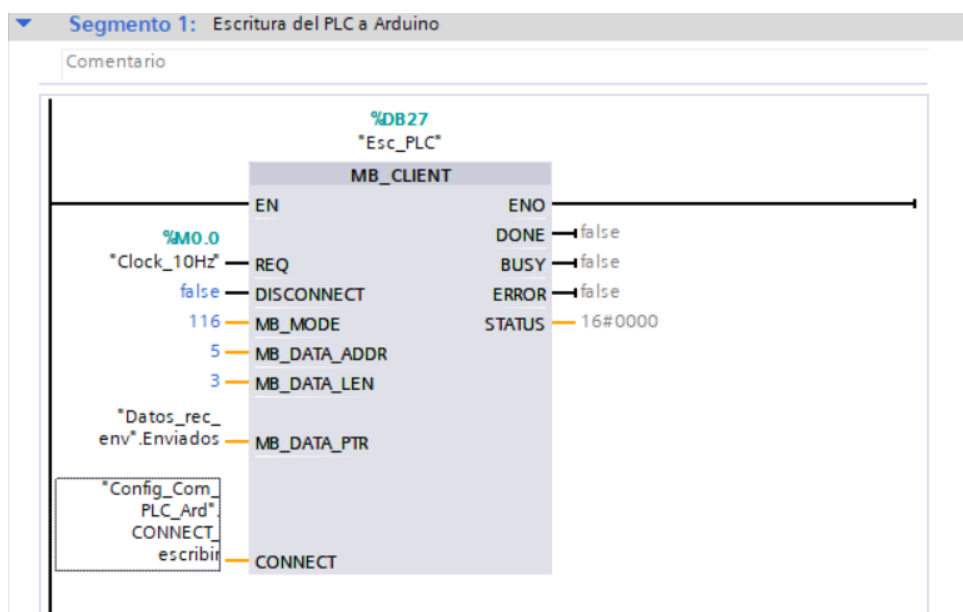
*Configuración de Comunicación PLC-Arduino*

	Nombre	Tipo de datos	Valor de arranq...	Remanen...	Accesible d...	Escrib...	Visible en ...	Valor de a..	Comentario
1	Static								
2	Leer_REQ	Bool	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
3	CONNECT_leer	TCON_IP_v4			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
4	CONNECT_escribir	TCON_IP_v4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
5	Interfaceld	HW_ANY	64	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	HW-identifier of IE-interface submodule
6	ID	CONN_OUC	91	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	connection reference / identifier
7	ConnectionType	Byte	11	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	type of connection: 11=TCP/IP, 19=UDP (1
8	ActiveEstablished	Bool	TRUE	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	active/passive connection establishment
9	RemoteAddress	IP_V4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	remote IP address (IPv4)
10	ADDR	Array[1..4] of Byte		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	IPv4 address
11	ADDR[1]	Byte	192	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	IPv4 address
12	ADDR[2]	Byte	168	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	IPv4 address
13	ADDR[3]	Byte	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	IPv4 address
14	ADDR[4]	Byte	30	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	IPv4 address
15	RemotePort	UInt	502	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	remote UDP/TCP port number
16	LocalPort	UInt	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	local UDP/TCP port number
17	Esc_req	Bool	false	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	

9. Ya con el bloque de datos creado con las configuraciones quedaría de la siguiente manera obsérvese la Figura 56.

**Figura 56**

*Envío de datos PLC hacia el Arduino*

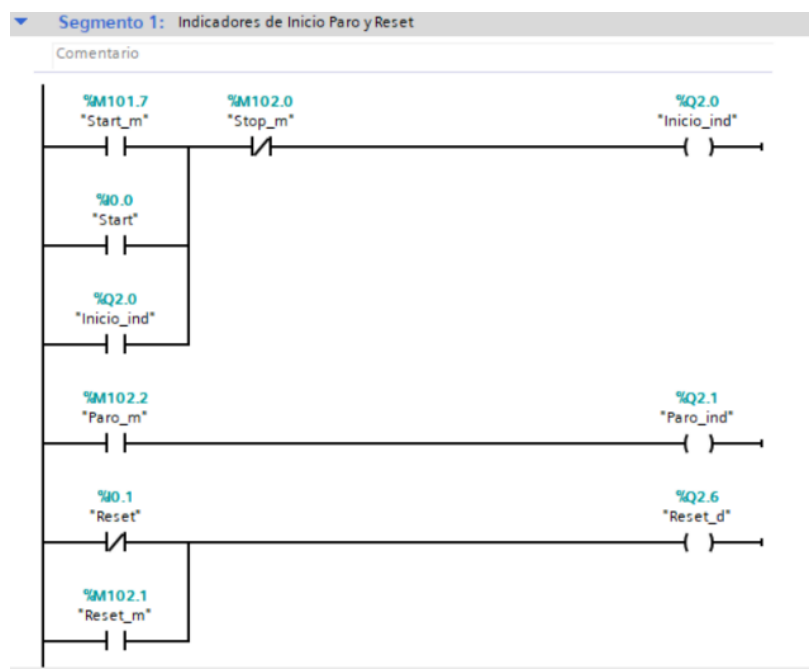


## Luces Indicadoras para cada sensor y actuador

Como parte de las normativas empleadas para el presente trabajo, se requiere de indicadores luminosos, los cuales, brinden información para conocer el estado en el que se encuentra el proceso, para ello se programa como se indica a continuación, se empieza con los indicadores de inicio, paro y reset como se indica en la Figura 57.

### Figura 57

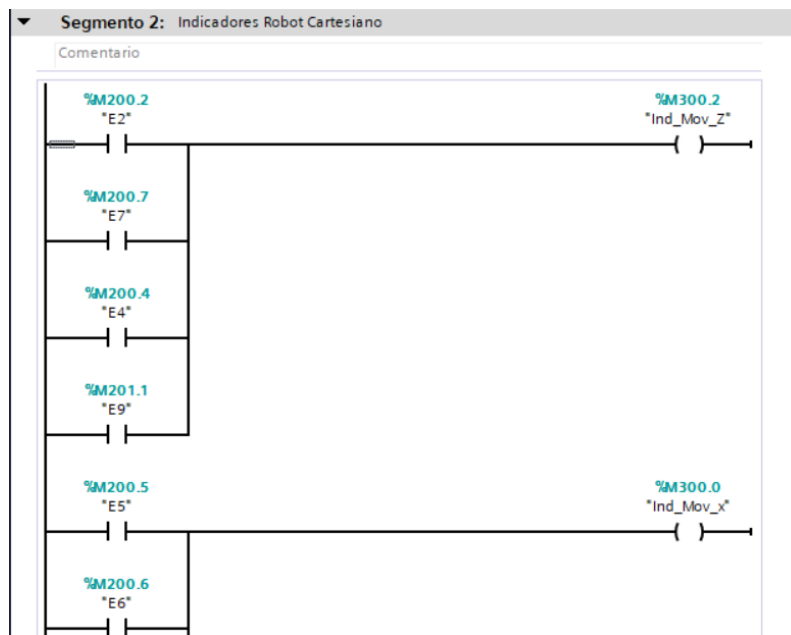
*Programación de indicadores de inicio, paro y reset*



Posteriormente, se procede a programar los indicadores del robot cartesiano, quienes, brindaran información acerca del movimiento que esté ejecutando, sea en el eje X o en el eje Z, como se indica en la Figura 58

**Figura 58**

*Programación de indicadores de robot cartesiano*



Culminando con los actuadores, programamos indicadores de las bandas transportadoras, presentado en la Figura 59.

**Figura 59**

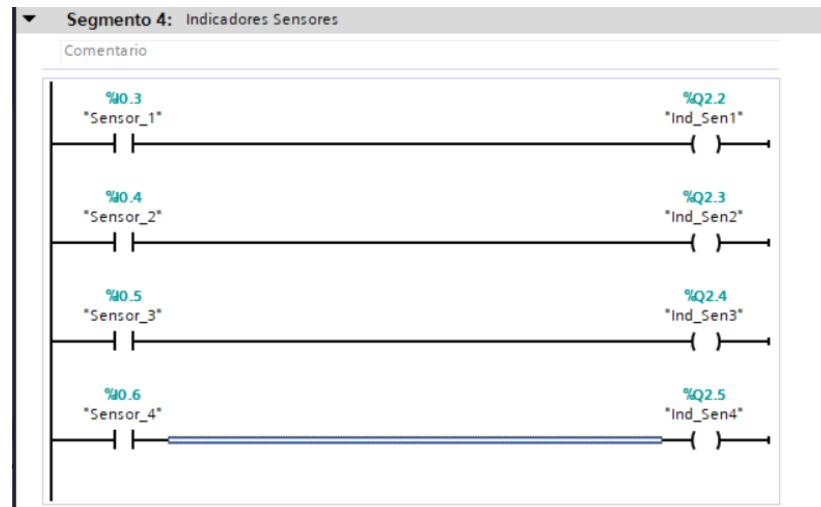
*Programación de indicadores bandas*



Consecuentemente, se procede a programar los indicadores correspondientes a los sensores, lo cuales brindarán la información cada que un sensor ha sido activado.

**Figura 60**

*Programación de indicadores de los sensores*

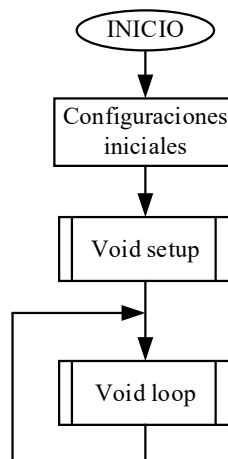


### Diagramas de flujo de la programación en la placa Arduino Mega

Para la programación de la placa Arduino, es necesario plantear un diagrama de flujo requerido para implantar la lógica de programación a seguir, en este caso, se plantea la ejecución del programa de la siguiente manera:

**Figura 61**

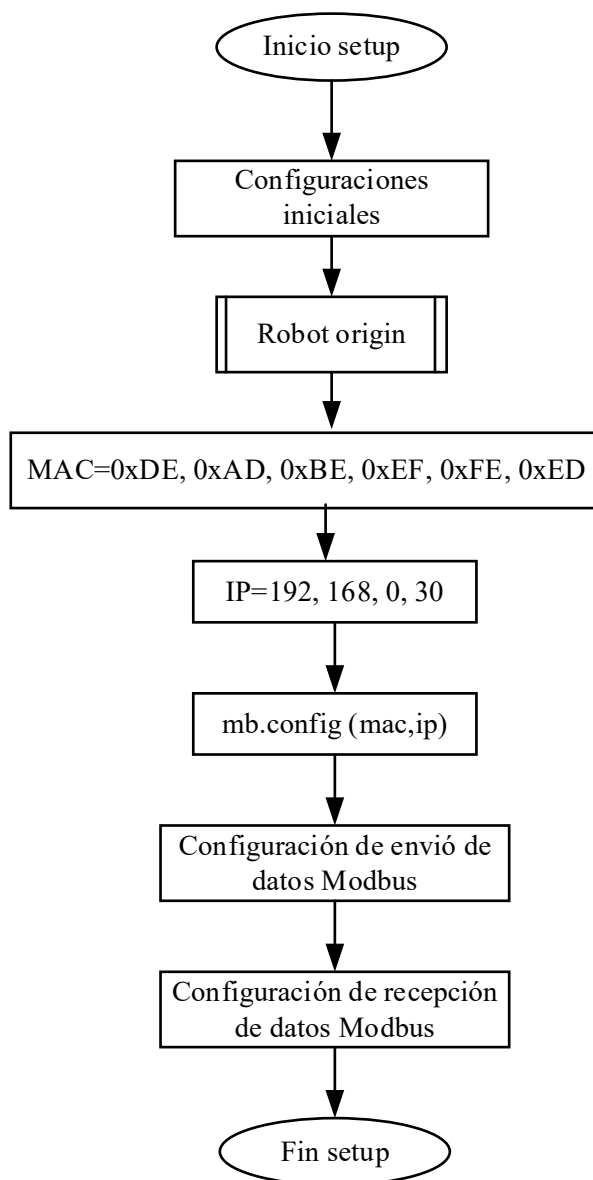
*Diagrama de flujo principal*



En la Figura 61 se muestra el diagrama de flujo de la programación en Arduino para el control de los servomotores que posee el robot antropomórfico manipulador, donde se desarrolla mediante subrutinas que se describen a continuación.

### Figura 62

#### Subrutina void setup

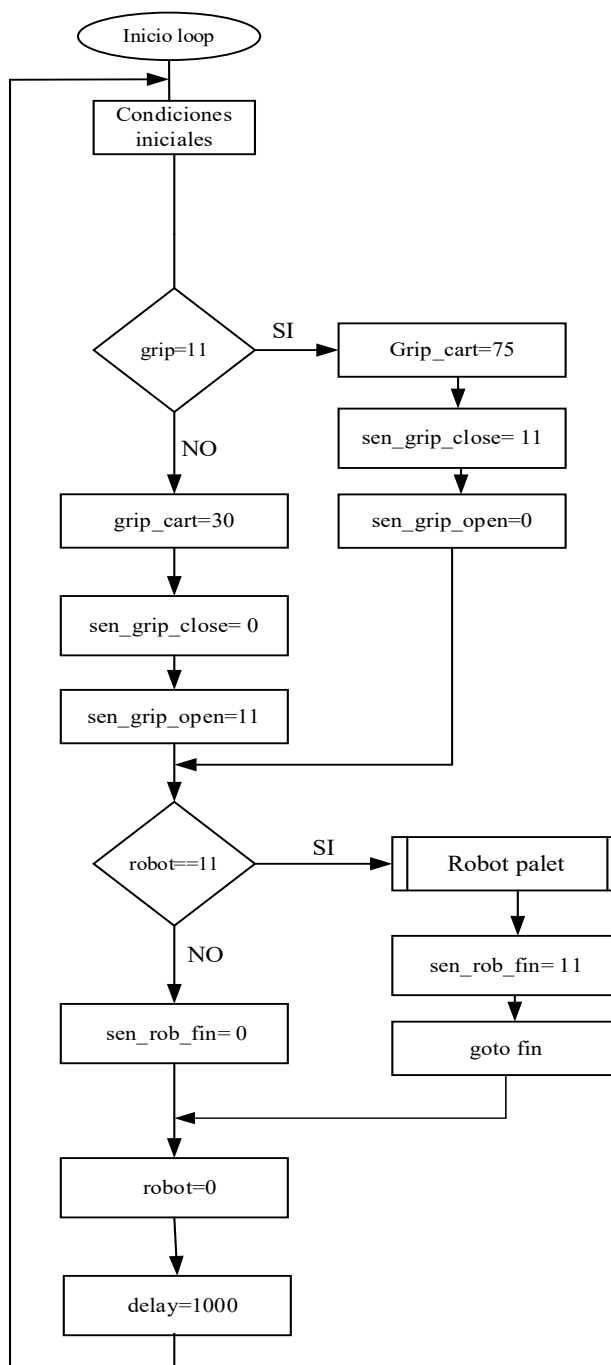


La Figura 62 describe la subrutina void setup en esta sección del programa se realiza las configuraciones iniciales como: llamar a la subrutina de posición home del robot,

configuración de la comunicación modbus TCP / IP, Asignar MAC e IP de la tarjeta Arduino Mega.

**Figura 63**

*Subrutina Void loop*

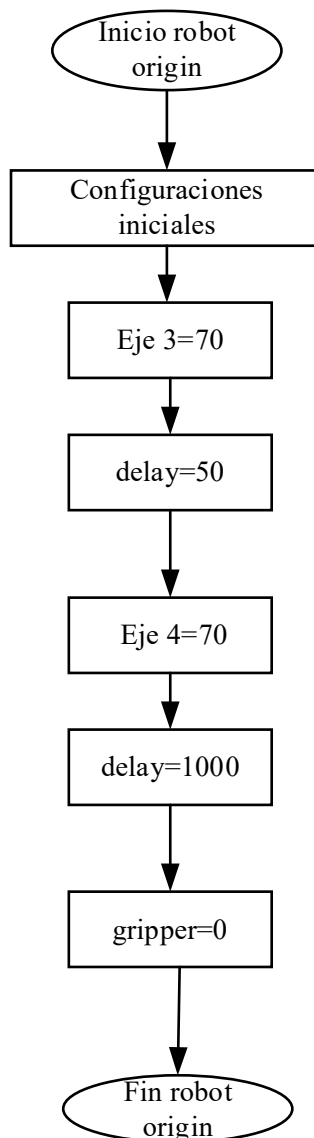




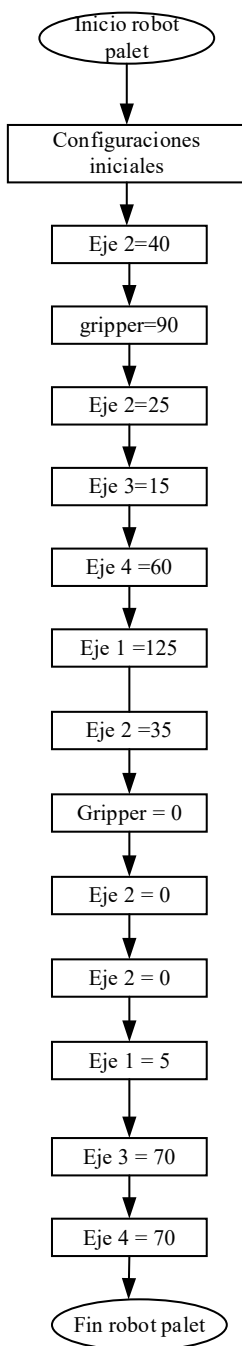
En la Figura 63 representa la subrutina del void loop donde realiza la tarea de paletizado cuando llega un mensaje desde el PLC hacia el Arduino mediante la shield Ethernet.

**Figura 64**

*Subrutina robot origin*



En la figura 64 se describe la subrutina robot origin la cual es posición home o en los grados que permanecerán los eslabones del robot mientras espera la señal de activación para entrar en la subrutina de paletizado.

**Figura 65***Subrutina robot palet*

En la Figura 65 se describe la subrutina del paletizado donde se escribe los grados que se mueven los motores para recoger el palet y colocar en la zona final de posicionamiento de los palets.

## Diseño del HMI para la estación didáctica

La supervisión de la estación didáctica, se realiza mediante una pantalla HMI, la cual, constará de botones de inicio, stop y reset, luces indicadoras de cada uno de los sensores que se encuentran dentro del proceso y luces indicadoras de las activaciones de los motores de las bandas y de los ejes del robot cartesiano.

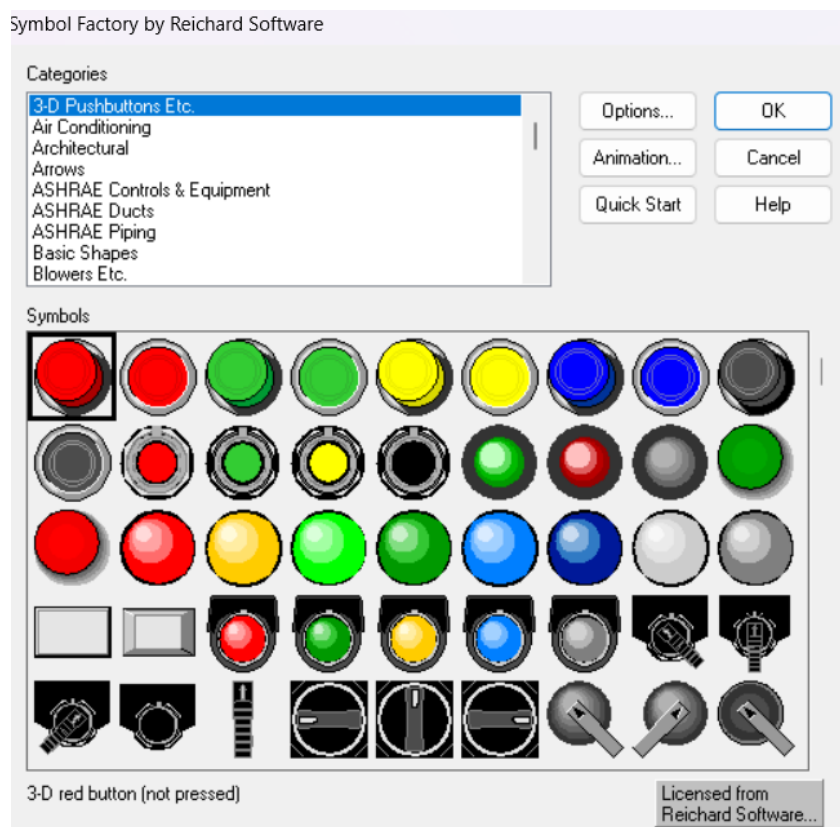
### ***Symbol Factory***

Symbol Factory es una biblioteca que permite crear íconos con los que se pueden interactuar, estos, son objetos industriales vectorizados que no pierden calidad de imagen al momento de redimensionar.

La Figura 66 indica todos los componentes industriales que se encuentran dentro de la biblioteca de íconos para la creación de HMI y Scada, dentro del software de InTouch.

### **Figura 66**

#### *Symbol Factory de InTouch*



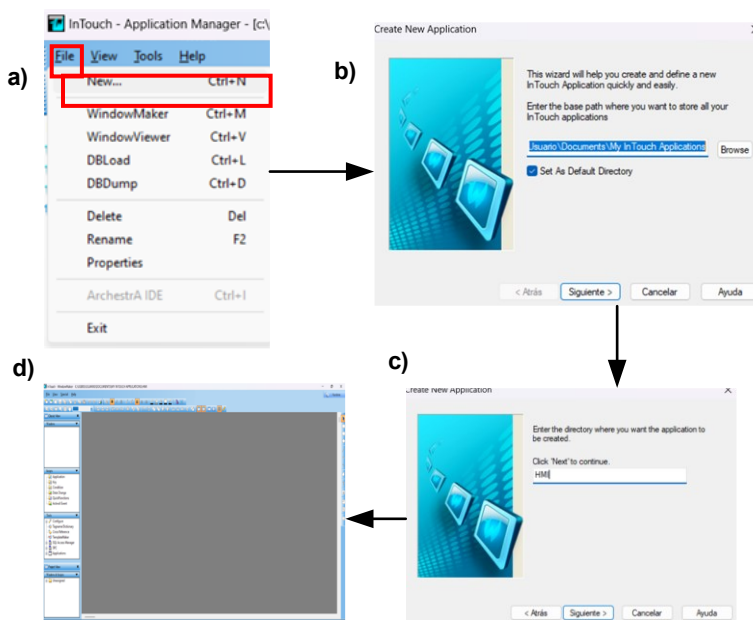
## Desarrollo del HMI en el software InTouch

Para poder crear el HMI dentro del software de InTouch, se debe seguir los siguientes pasos:

1. Una vez abierto el software, se procede a crear una nueva aplicación dando click en la opción de “File, new” o por teclado con Ctrl+N, lo dicho anteriormente se muestra en el literal a) la Figura 67.

**Figura 67**

*Creación de una nueva aplicación*







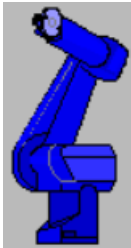
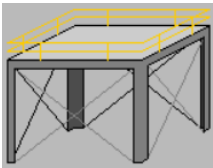
2. Ya una vez hecho el paso anterior, saltará una ventana emergente, la cual indica la ubicación en donde se va a guardar la aplicación. Para este caso lo vamos a dejar en la ubicación que viene por defecto, como se puede observar en el literal b) de la Figura 67 .
3. El siguiente paso a seguir es ponerle un nombre a la nueva aplicación que se está creando, para este caso se llamará HMI, lo cual es representado en el literal c) de la Figura 67, luego de dar el nombre click en siguiente.

4. Una vez creada la nueva aplicación saltará una nueva ventana literal d) de la Figura 67, la cual permite crear el HMI.
5. Ya dentro del programa de WindowMaker se creó una nueva ventana, y dentro de ella se coloca todos los componentes necesarios con la herramienta de Symbol Factory.
6. Los íconos industriales vectorizados que se utilizaron para la creación del HMI. Son los siguientes.

**Tabla 1**

*Íconos industriales vectorizados utilizados en el HMI*

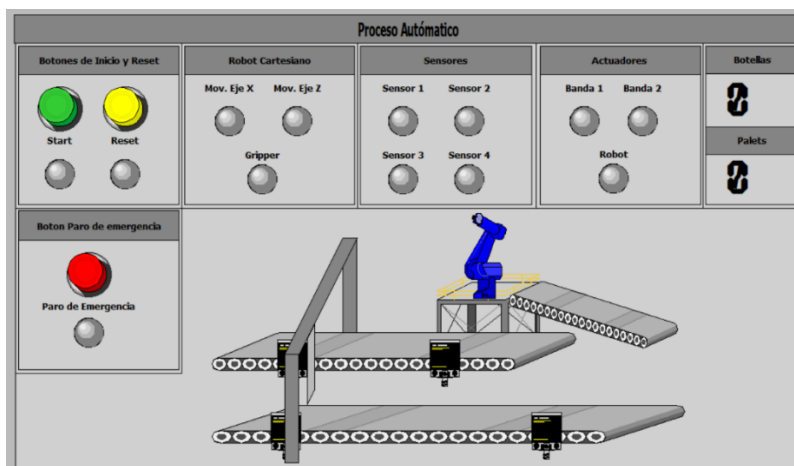
<b>Símbolo</b>	<b>Descripción</b>	<b>Símbolo</b>	<b>Descripción</b>
	Botón de encendido sin pulsar		Botón de encendido pulsado
	Botón de paro de emergencia sin pulsar		Botón de paro de emergencia pulsado
	Botón de Reset sin pulsar		Botón de reset pulsado
	Luz Indicadora apagada		Luz Indicadora encendida para sensores, actuadores y de start
	Luz Indicadora encendida para Reset		Luz Indicadora encendida para Parado de Emergencia

Símbolo	Descripción	Símbolo	Descripción
	Representación gráfica del palet		Representación gráfica de la botella.
	Representación gráfica de los sensores		Representación gráfica de las bandas
	Representación gráfica del Robot antropomórfico		Representación gráfica del soporte en donde se encuentra el robot antropomórfico.

7. La pantalla del HMI, representa un proceso batch de paletizado de botellas, el cual consta con luces indicadores de los sensores, indicadores de accionamiento del robot cartesiano, indicadores de accionamiento de las bandas y del robot. Para el control se tiene botones de Start, reset y paro de emergencia. Todo lo anteriormente mencionado se observa en la Figura 68

**Figura 68**

*HMI del proceso automatizado.*



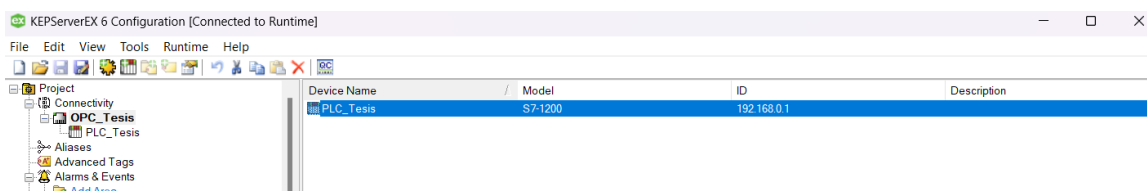
## Comunicación del PLC con InTouch.

Para poder establecer comunicación del PLC e InTocuh, se utilizó la comunicación OPC, la cual consta de una arquitectura basada en cliente/servidor, siendo el software de KepServer el servidor, y los clientes son el controlador lógico programable PLC y el software InTouch.

Para instaurar el servidor dentro de KepServer, se crea un nuevo canal y dentro de este, se creará un nuevo dispositivo, donde especificaremos el PLC a usar, el modelo y la dirección IP que tiene dicho elemento. Esto se puede observar en la Figura 69.

**Figura 69**

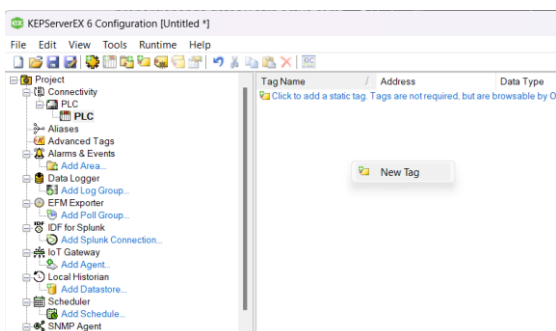
*Creación del servidor OPC.*



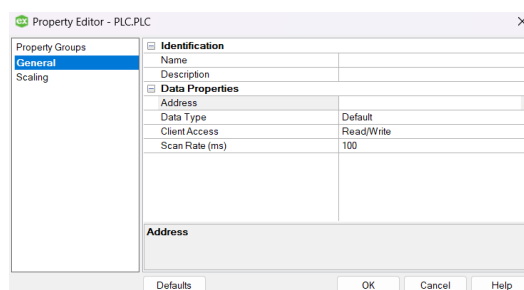
Los tags necesarios para comunicar el PLC con InTocuh se crean dentro del dispositivo antes mencionado; para crear los tags dentro del dispositivo se da click derecho y se busca la opción New Tag (Figura 70), después aparecerá una ventana emergente, la cual permite dar el nombre al tag y la dirección del mismo, a la que vamos a acceder desde InTouch.

**Figura 70**

*Creación de tags*



a) Creación de un nuevo tag

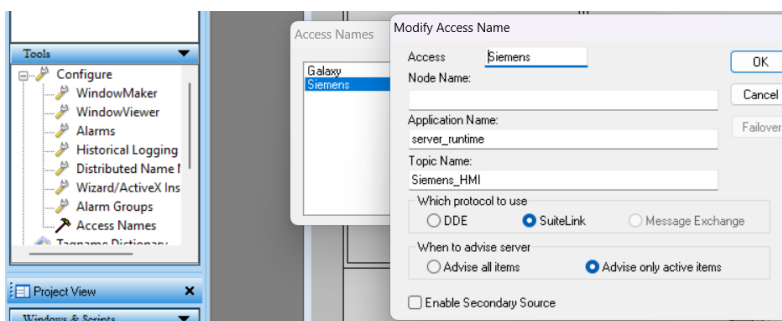


b) Dar nombre y la dirección del tag

Para acceder a todas las tags que se crearon en KepServer desde InTouch se requiere un nuevo Access Name el cual debe contener las opciones de Access, Application Name y Topic Name, como se muestra en la Figura 71.

**Figura 71**

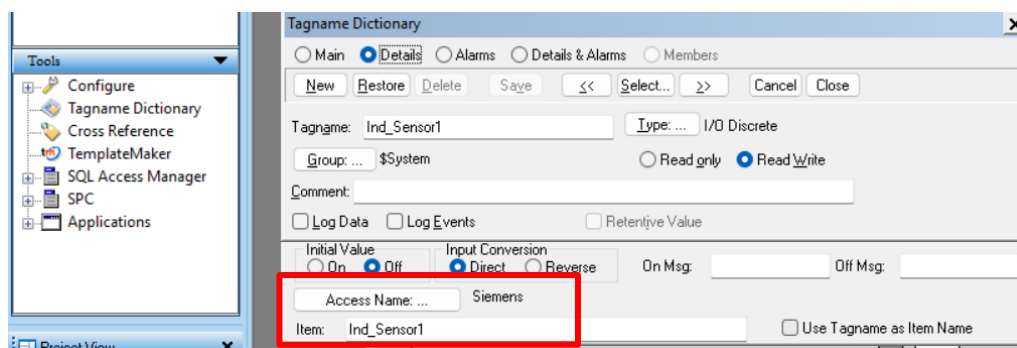
*Creación de un Access Name*



Después de crear el Access Name, se establece los tags que se van a ocupar dentro del HMI, seleccionando el tipo de tag, Acces Name antes mencionado y estableciendo el mismo nombre que se encuentra en el servidor OPC; lo anteriormente mencionado se puede observar en la Figura 72.

**Figura 72**

*Creación y enlace de los tags*





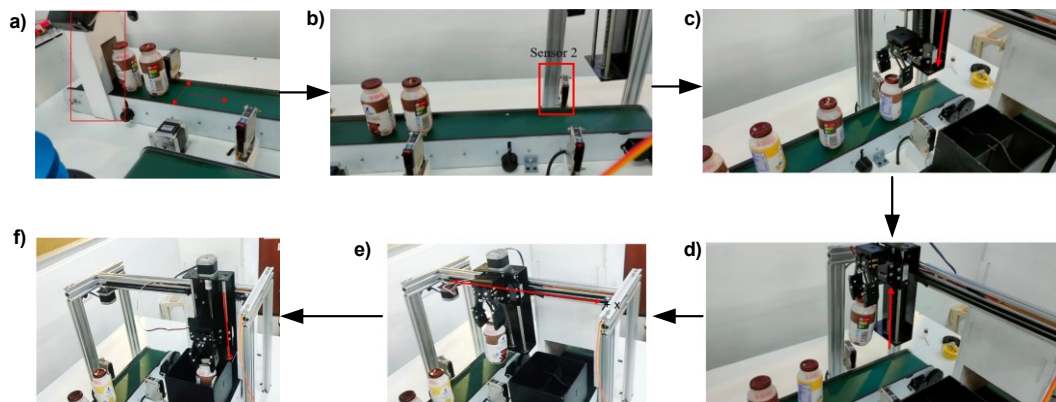
## Capítulo IV

### Análisis de resultados

Ejecutar pruebas de funcionamiento de la estación empleando el modo automático.

**Figura 73**

*Funcionamiento del proceso de forma automática*



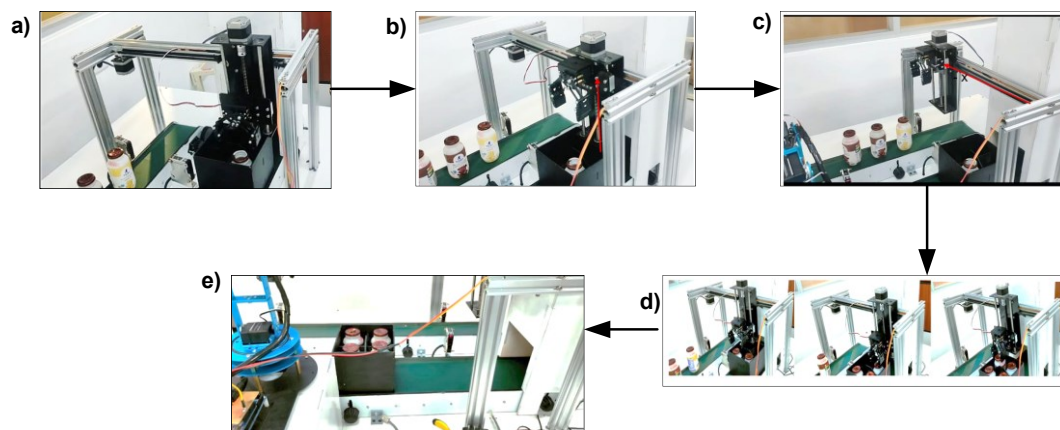
En el literal a) de la Figura 73 se muestra la etapa 1 donde la estructura de madera se encarga de ubicar al envase en la posición correcta sobre la banda transportadora de forma que quede alineada al gripper de brazo cartesiano.

En el literal b) se presenta el transporte de los envases hacia la posición donde se encuentra el sensor 2, cuando el envase llega al rango de medición los contactos pasan a estar normalmente cerrado; en el literal c) se muestra la etapa 2 donde se activa el eje z desplazando hacia abajo hasta la altura del envase.

En el literal d) presenta el agarre del envase y desplazamiento hacia arriba del eje z; en el literal e) se presenta el traslado del envase desde la banda transportadora 1 hacia la posición donde está ubicado el palet. En el literal f) se presenta la colocación del envase 1 en la posición uno del palet de 4 envases.

## Figura 74

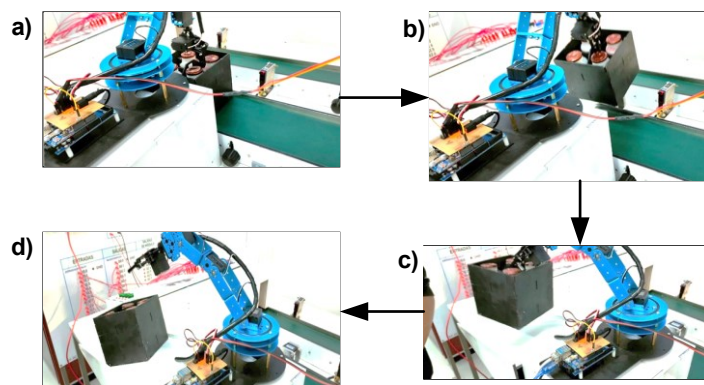
### Proceso de llenado el del palet



En el literal a) de la Figura 74 se presenta la colocación del envase en el palet una vez que el eje z llegue a la posición deseada sobre el palet, el gripper se abre dejando caer al envase por completo. En el literal b) se presenta como el eje z sube con el gripper abierto hasta llegar a la posición en la parte superior. En el literal c) se presenta el movimiento del robot cartesiano en el eje x, regresando a la posición inicial, y repitiendo el ciclo hasta cumplir con el número de envases; en el literal d) de la se presenta el traslado del palet lleno hacia la ubicación del sensor 4 donde se detendrá la banda transportadora, ahí el PLC envía en un dato hacia el Arduino para que entre en la subrutina de robot palet.

## Figura 75

### Funcionamiento del Robot manipulador organización de palet lleno

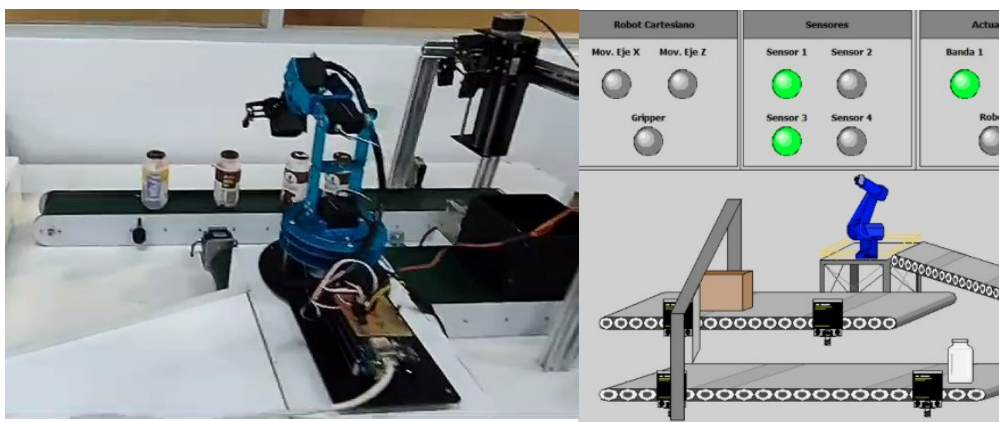


En la Figura 75 se muestra como el robot realiza la tarea de paletizado tomando el palet con el gripper revise el literal a), el robot alza el palet, luego de esto el robot gira 90 grados para depositar el palet en la rampa de salida del producto final.

## Resultados obtenidos del HMI

### Figura 76

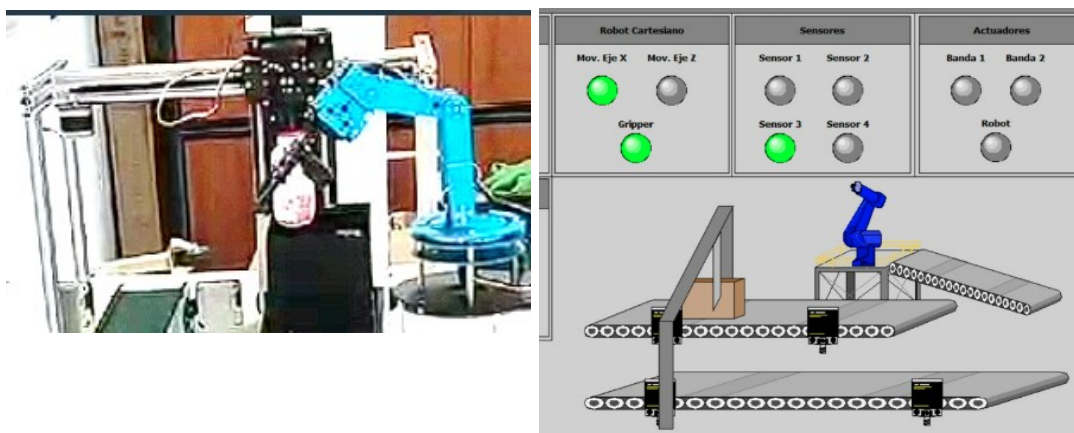
*Luces indicadoras encendidas del sensor 1 y sensor 3*



En la Figura 76 se puede observar en el HMI las luces indicadoras del sensor 1, sensor 3 están activas, indicando que en cada sensor ubicado en el proceso existe una botella y una caja respectivamente, también indica que la banda 1 está activa.

### Figura 77

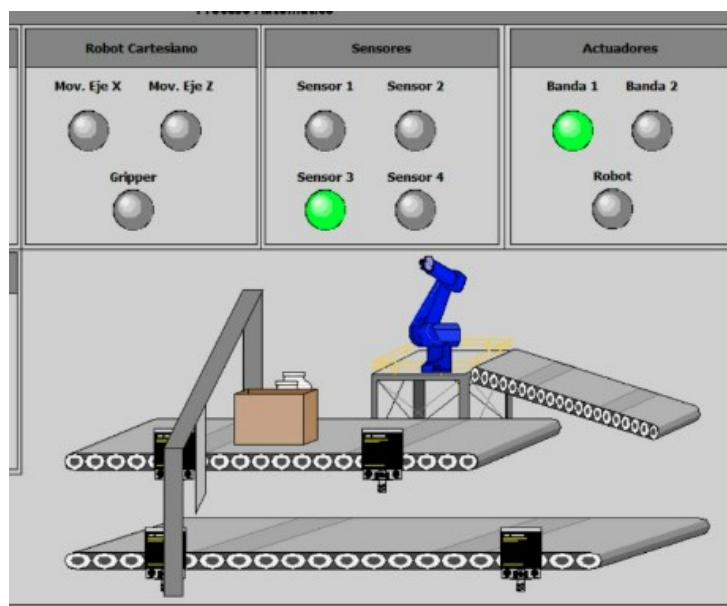
*Movimiento del Robot Cartesiano en el HMI*



En la Figura 77 se puede observar en el HMI el desplazamiento del robot cartesiano en el eje x, comparado con el proceso, en el HMI se indica que se está ejecutando el movimiento en dicho eje, adicionalmente, que el gripper del robot cartesiano está cerrado

### Figura 78

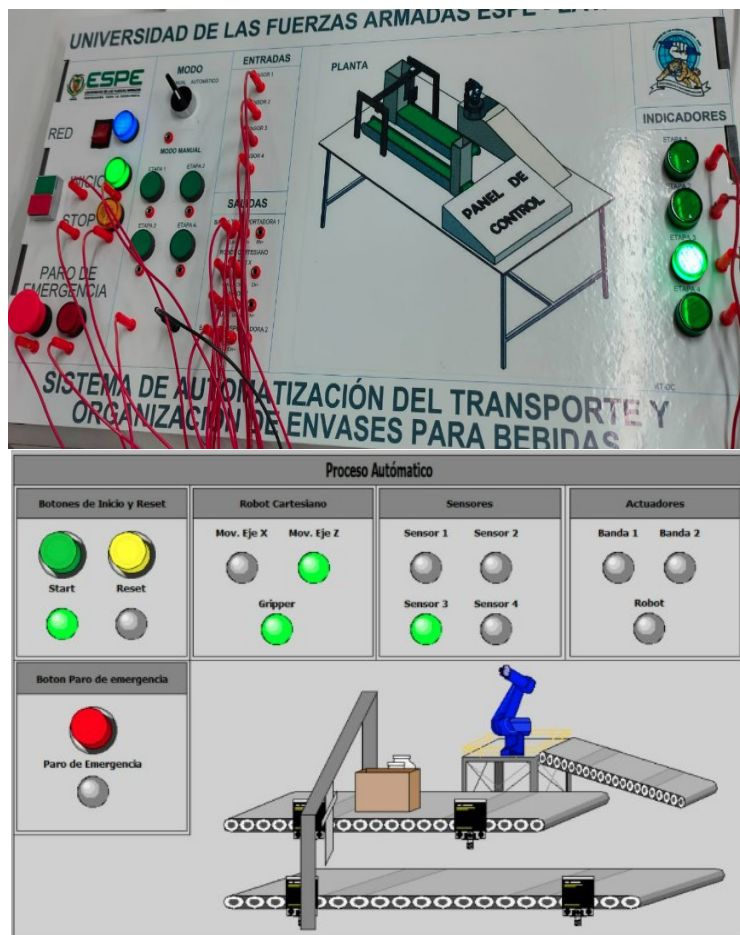
*Dos botellas paletizadas*



En la Figura 78 se puede observar dentro del HMI que en la banda transportadora 2 ya están dos botellas paletizadas y que existe una botella que se está transportando hacia el robot cartesiano.

**Figura 79**

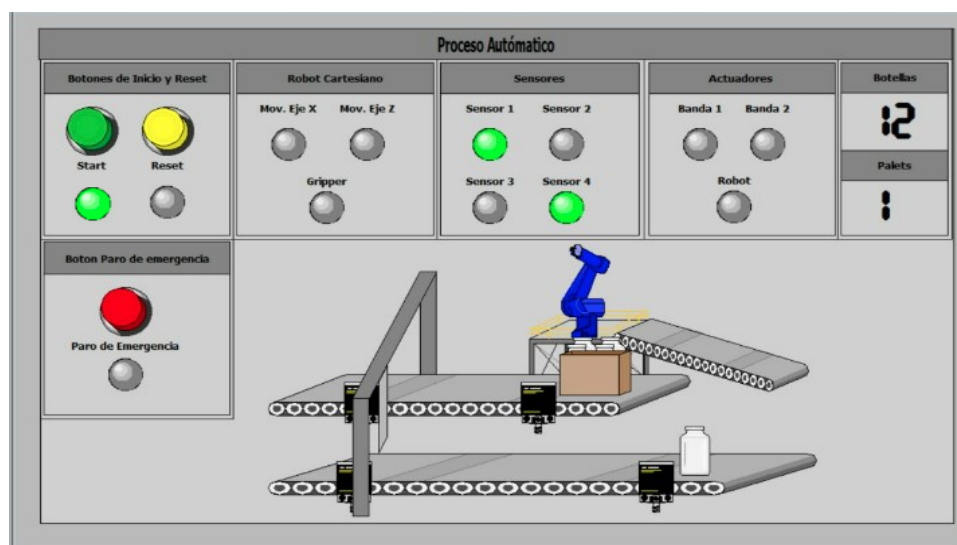
*Comparación del Panel de Control con el HMI*



En la Figura 79 se puede observar que en el panel de control del proceso se encuentran encendidas dos luces indicadoras, una de ellas representa que está activo el proceso y la segunda luz indica que existe un palet detectado por el sensor 3 y de igual manera se puede visualizar en el HMI.

**Figura 80**

*Supervisión de entrada y salidas de Productos*



En la Figura 80 se puede observar que ya existe un palet en la rampa de salida, también que existe un palet entrando hacia el robot antropomorfo para que lo coloque en la rampa de salida y existen otras tres botellas que están siendo transportadas hacia el robot cartesiano.

### **Resultados obtenidos de pruebas**

En el Anexo 1 (Tabla 4) se presenta los datos obtenidos en modo automático donde se lleva el número de envases que ingresan a la etapa 1, el conteo de los envases que llegan correctamente a la etapa 2, adicionalmente, se hizo el conteo de envases sujetados por el gripper correctamente y el número de envases paletizados correctamente colocados en la jaba palet.

Con los resultados obtenidos en el conteo de envases en las etapas de traslación y colocación dentro de palet se obtiene de un total 65 envases que ingresaron, 2 se descartaron en la etapa 2 debido a una mala detección provocada por el sensor. En la sujeción del gripper se descartó tres envases por su mala ubicación en la banda transportadora, y, por último, en la etapa de paletizado se perdieron 4 envases por caída del gripper en el proceso de traslado y por mala ubicación del palet (java).

**Tabla 2**

*Número de envases perdidos - número de envases paletizados*

<b>Número de Prueba</b>	<b>Número de envases</b>	<b>Número de envases perdidos</b>	<b>Numero de envases paletizados</b>
1	65	4	61
2	65	5	60
3	65	3	62
4	65	4	61
5	65	2	63
<b>Promedio</b>	<b>65</b>	<b>3,6</b>	<b>61,4</b>
<b>Porcentaje</b>		<b>5,5%</b>	<b>94,5%</b>

Por lo tanto, se realizó 5 pruebas en diferentes días y diferentes condiciones con el mismo número de envases por prueba con el fin de evaluar la cual es el porcentaje de error que presenta la automatización de este proceso. En la Tabla 2 se muestra que en número de pruebas realizadas se obtiene un 5.5% perdidas y el 94.5% se paletizo con normalidad y correctamente.

Las pruebas en la etapa de paletizado y traslación por la banda número 2 dan como resultado el número de botellas que llegaron a esta etapa, las cuales dan un total 61 envases, con 4 por cada palet; revise la Tabla 5 del Anexo 1, exceptuando los envases que se perdieron durante el proceso por los inconvenientes antes mencionados.

**Tabla 3**

*Número de palets completos - Numero de palets incompletos*

<b>Número de Prueba</b>	<b>Número de palets</b>	<b>Número de palets incompletos</b>	<b>Número de palets completos</b>
1	16	2	14
2	16	3	13
3	16	2	14
4	16	2	14
5	16	1	15
<b>Promedio</b>	16	<b>2,0</b>	<b>14,00</b>
<b>Porcentaje</b>		<b>12,5%</b>	<b>87,5%</b>

Con los datos obtenidos en la Tabla 3 se muestra los resultados obtenidos se tiene que existe un 12.5 % de palets incompletos y el 87.5 % de palets completos, tomar en cuenta que un palet se considera incompleto sea por falta de un envase o hasta 4, es decir que estos datos dependen de la eficacia de las etapas anteriormente mencionadas.



## Capítulo V

### Conclusiones y Recomendaciones

#### Conclusiones.

- Los diagramas eléctricos de conexión permiten la identificación de los sensores, actuadores, controladores, etc. en el proceso, para verificación del funcionamiento correcto, así como también para realizar los diferentes tipos de mantenimiento industrial.
- El uso de diagramas grafcet en la solución de problemas de automatización facilitan la programación en lenguaje Ladder, dividiendo el proceso en varias etapas y transiciones, en las cuales se activan los actuadores dependiendo de la necesidad de la etapa y de las transiciones que se ejecuten.
- Los sensores y actuadores del sistema son compatibles con el controlador lógico programable PLC Siemens S7-1200, debido a las señales estándares de control que generan y reciben todos los instrumentos industriales de la estación.
- La implementación de una Interface Humano Máquina en un sistema de automatización permite la interacción entre el operador del sistema y el proceso, esto para ingresar las variables de interés, tales como, cantidad de envases, cantidad de palets, también otras acciones como el paro de emergencia o la puesta en marcha desde dicha interface.
- Los resultados obtenidos en la etapa de pruebas permiten afirmar que la automatización de este proceso es eficaz, debido a que existe un 5.5% de pérdidas de envases durante la tarea de paletizado y organización. Dichas pérdidas, son generadas por factores externos a la automatización del proceso, tales como el diseño de la planta.

### **Recomendaciones.**

- La utilización del OPC KEPServer DEMO tiene limitación de tiempo de uso de 2 horas por no contar con una licencia, esa siendo la limitación para el monitoreo en tiempo real del HMI realizado en Wonderware Intouch. Por lo cual tener en cuenta este tiempo para volver a iniciar el OPC cada que se termine el runtime. Es recomendable adquirir una licencia empresarial para evitar estas limitaciones.
- La utilización de tarjetas Arduino limita en cuanto a robustez de la planta ya que por factores físicos (leves movimientos) se puede desconectar la shield ethernet de comunicación entre el PLC y el Arduino MEGA, por lo cual se recomienda utilizar tarjetas que ya tenga incorporado esta función y el periférico para entrada RJ45 como tarjetas Raspberry o Jetson de la fábrica NVIDIA.
- La comunicación Modbus TCP /IP se realizó en este proyecto, pero se utilizó un switch TP-LINK el cual es de uso ofimático, que para una estación didáctica es aceptable, para darle robustez y mira a lo industrial se recomienda utilizar equipos de marcas como MikroTik y Cisco.

### **Trabajos Futuros**

- Este proyecto está diseñado bajo la normativa ISA S 88 que permite cambiar y modificar las etapas del proceso, se podría implementar el internet de las cosas IIOT para llevar los datos de producción (paletizado) hacia la nube, e incluso mediante el uso de aplicaciones IOT para smartphone la creación de bots de alerta temprana cuando suceda una anomalía en el proceso.

## BIBLIOGRAFÍA

- QualiaTech Corp Company. (2021). *automation-networks*. Obtenido de PROTOCOLO MODBUS: <https://automation-networks.es/glossary/modbus-tcpip>
- Adsrobotics. (27 de Julio de 2020). *Adsrobotics*. Obtenido de Adsrobotics: <https://www.adsrobotics.com/blog/automatizacion-procesos-industriales/>
- Anderson, D. (17 de Julio de 2018). *slideshare*. Obtenido de MEMORIAS DE LOS PLC: <https://es.slideshare.net/DannyAnderson19/memorias-de-los-plc>
- Arduino. (24 de 01 de 2024). *Arduino Ethernet Shield*. Obtenido de Arduino Docs: <https://docs.arduino.cc/retired/shields/arduino-ethernet-shield-without-poe-module/>
- Arduino, S. (2024). *Arduino Store*. Obtenido de Arduino Store: <https://store.arduino.cc/products/arduino-mega-2560-rev3>
- Arrieta Arvilla, R. E., & Gonzales, K. F. (2003). TUTORIAL BASICO PARA PROGRAMACIÓN DE PLC. Cartagena, Colombia.
- Barrietos A., P. L. (1997). *Fundamentos de Robótica*. Madrid: McGRAW-HILL.
- DE LORENZO OF AMERICA CORP. S.A. (s.f.). *delorenzo.com.mx*. Obtenido de ROBOT MANIPULADOR CON 6 GRADOS DE LIBERTAD: <https://delorenzo.com.mx/descargas/ROBOTICA/DL%20ROBSIX.PDF>
- Gonzalez, O. (2024). *uía de modelos Arduino y sus características*. Obtenido de bricogeek: <https://lab.bricogeek.com/tutorial/guia-de-modelos-arduino-y-sus-caracteristicas/arduino-mega-2560>
- López J., A. M. (2007). *Robot Cartesiano: seguimientos de trayectorias irregulares arbitrarias mediante computadora*. México: Universidad Autonoma del estado de Hidalgo.
- Marcia Carrion, C. R. ( 2012). *DESARROLLO DE SOFTWARE HMI SCADA E IMPLEMENTACIÓN SOBRE UN MÓDULO DIDÁCTICO AUTÓNOMO PARA VENTAS DELA EMPRESA ECUAINSETEC CÍA. LTDA*. Quito: UPS.

Martínez, H. (2015). *MODULO DIDÁCTICO PARA PRÁCTICAS DE LABORATORIO CON CONTROLADORES LÓGICOS PROGRAMABLES*. SAN NICOLÁS DE LOS GARZA, NUEVO LEÓN: U.N.L.

Pablo, D. (2008). *PLC Automatización y Control Industrial* . Buenos Aires : Editorial Hispano Americana S.A. .

Plcynergy. (2023). *Plcynergy*. Obtenido de What Is Modbus TCP Protocol? Introduction To ModbusTCP: <https://plcynergy.com/modbus-tcp-protocol/>

Sainsmart. (2024). *www.sainsmart.com*. Obtenido de Mega2560+Ethernet Shield Kit for Arduino ATMEGA8U2 W5100: <https://www.sainsmart.com/products/sainsmart-mega2560-ethernet-shield-kit-for-arduino-atmega8u2-w5100>

Vallejo. (2018). *PLC Los Controladores Lógicos Programables*.

Vestertraining. (2024). *vestertraining.com*. Obtenido de vestertraining.com: <https://vestertraining.com/blog/diferencias-scada-hmi/>

**ANEXOS**