



Diseño e implementación de la estructura metálica, los equipos e instrumentos para el funcionamiento en modo manual del sistema de transporte y organización de envases para bebidas

Cunalata Orellana, Diego Andrés y Toapanta Toapanta, Kelvin Fabricio

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

4 de marzo del 2024

Latacunga



Plagiarism and AI Content Detection Report

Trabajo de integración curricular Die...

Scan details

Scan time:
March 4th, 2024 at 1:25 UTC

Total Pages:
50

Total Words:
12297

Plagiarism Detection



Types of plagiarism		Words
Identical	1.6%	200
Minor Changes	0.5%	65
Paraphrased	0.8%	104
Omitted Words	0%	0

AI Content Detection



Text coverage		Words
AI text	0%	0
Human text	100%	12297

[Learn more](#)

Firma:

Ing. Pruna Panchi, Edwin Patricio, Mgs.

C. C.: 0502651003



Departamento de Eléctrica, Electrónica y Telecomunicaciones

Carrera de Electrónica y Automatización

Certificación

Certifico que el trabajo de unidad de integración curricular: **“Diseño e implementación de la estructura metálica, los equipos e instrumentos para el funcionamiento en modo manual del sistema de transporte y organización de envases para bebidas”** fue realizado por los señores **Cunalata Orellana, Diego Andrés y Toapanta Toapanta, Kelvin Fabricio**, 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, 5 de marzo de 2024

Firma:

Ing. Pruna Panchi, Edwin Patricio, Mgs.

C. C.: 0502651003



Departamento de Eléctrica, Electrónica y Telecomunicaciones

Carrera de Electrónica y Automatización

Responsabilidad de Autoría

Nosotros, **Cunalata Orellana, Diego Andrés**, con cédula de ciudadanía n° 1804428850 y **Toapanta Toapanta, Kelvin Fabricio**, con cédula de ciudadanía n° 1725923476, declaramos que el contenido, ideas y criterios del trabajo de unidad de integración curricular: **“Diseño e implementación de la estructura metálica, los equipos e instrumentos para el funcionamiento en modo manual del sistema de 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, 5 de marzo de 2024

Cunalata Orellana, Diego Andrés

C. C.: 1804428850

Toapanta Toapanta, Kelvin Fabricio

C. C.: 1725923476



Departamento de Eléctrica, Electrónica y Telecomunicaciones

Carrera de Electrónica y Automatización

Autorización de Publicación

Nosotros, **Cunalata Orellana, Diego Andrés**, con cédula de ciudadanía n° 1804428850 y **Toapanta Toapanta, Kelvin Fabricio**, con cédula de ciudadanía n° 1725923476, autorizamos a la Universidad de las Fuerzas Armadas ESPE publicar el trabajo de unidad de integración curricular: **“Diseño e implementación de la estructura metálica, los equipos e instrumentos para el funcionamiento en modo manual del sistema de transporte y organización de envases para bebidas.”** en el Repositorio Institucional, cuyo contenido, ideas y criterios son de nuestra responsabilidad

Latacunga, 5 de marzo de 2024

Cunalata Orellana, Diego Andrés

C. C.: 1804428850

Toapanta Toapanta, Kelvin Fabricio

C. C.: 1725923476

Dedicatoria

Para mis queridos padres Teresa Orellana y Mario Cunalata, con amor y gratitud les dedico este trabajo de titulación que marca el fin de una etapa importante en mi vida académica. Con su apoyo incondicional, consejos y sacrificios, este logro no habría sido posible. Gracias por ser mis pilares y por creer en mi desde el principio.

A mis hermanos y hermana Juan, Mario, Maritza quienes siempre han sido un apoyo incondicional en mi vida, a ustedes mis confidentes incondicionales, les dedico este logro con todo mi cariño.

A Pao, aunque nuestros caminos hayan tomado rumbos diferentes, no puedo negar el papel fundamental que desempeñaste en este logro. Siempre recordaré con gratitud el tiempo que compartimos y como contribuiste a mi crecimiento personal y académico.

Diego Cunalata

Agradecimiento

A toda mi familia en especial a mi madre Teresa, quien siempre ha sido mi mayor motivación y a pesar de mis malas rachas siempre creíste en mí. Por tu amor incondicional, tu constante apoyo y tu inquebrantable fe en mí. Eres mi inspiración, mi guía y mi mayor ejemplo de fortaleza y perseverancia.

A mi tutor, el Ing. Edwin Pruna por su invaluable orientación, dedicación y enseñanzas durante el desarrollo de este trabajo de titulación, por su calidad como docente y persona, que gracias a su sabiduría, paciencia y apoyo han sido fundamental en mi crecimiento académico y profesional.

También agradezco a todos mis docentes que me han acompañado durante toda mi carrera profesional, gracias a cada uno de ellos por su amistad, enseñanza, consejos y correcciones que ha sido fundamental hacia la culminación de mi carrera.

Diego Cunalata

Dedicatoria

A mis amados padres, Delia Toapanta y Rodolfo Toapanta por el enorme sacrificio que juntos cargaron en sus hombros para hacer posible esto, por el cariño y consejos que nunca faltaron, este logro les pertenece a ustedes.

A mi amor Mercedes y mi pequeño Mael quienes son mi refugio y mi paz, porque se convirtieron en mi inspiración para nunca rendirme y siempre luchar por nuestro futuro brillante.

A mis hermanos Danilo, Karen y Jhostin por el apoyo que desinteresadamente me brindaron en los momentos difíciles del transcurso de esta etapa de mi vida, este logro fue gracias a todos ustedes.

A mis sobrinos Emily y Jared que siempre los llevo dentro de mi corazón.

Kelvin Fabricio

Agradecimiento

Agradezco a Dios por la vida y el bienestar de todos mis seres amados. Un agradecimiento especial a mi hermosa madre, por siempre darme un amor tan puro y sincero, por esas madrugadas en las que nunca me permitió salir sin desayunar, madre querida te agradezco porque siempre creíste en mí. A mi honorable padre, por las enseñanzas y el apoyo incondicional, por la espera paciente en tu automóvil en aquella gasolinera cuando regresaba a casa por las noches. A ustedes, mi eterna gratitud porque fueron mi pilar y mi ejemplo a seguir, ruego que la vida me alcance para poder darles todo lo que un día soñé.

A mi querida Mercedes, por el amor dulce y sincero que me brinda día a día el cual ha sido mi fortaleza para luchar en los momentos difíciles, gracias por ser mi compañera de vida y madre mi retoño. A mi hijo Maelito, gracias por convertirme en mejor persona y enseñarme esta etapa de padre, con sus pequeñas manitos y sonrisa ingenua me ha impulsado a seguir luchando por conseguir un futuro mejor, eres un milagro para mí, contigo me siento tan feliz.

A mi hermano Danilo, por actuar como un segundo padre, por sus consejos y por el apoyo en los momentos más decisivos de mi vida académica y personal. A mi hermana Karen, por ser mi amiga incondicional y un apoyo en momentos difíciles, por aquellos dulces que escondías en mi maleta. A mi hermano Jhostin, por su cariño, alegría y admiración, por esas bonitas experiencias de la infancia, por ese gusto musical que tenemos en común. También agradezco a todos mis amigos que han estado presentes en esta etapa.

Finalmente, agradezco a todos los docentes de la Universidad de las Fuerzas Armadas que pusieron un granito de conocimiento en mí, en especial a nuestro tutor de tesis Ing Edwin Pruna Panchi, por su guía y orientación en este proceso importante, gracias por compartir su conocimiento y experiencia.

Kelvin Fabricio

ÍNDICE DE CONTENIDOS

Carátula.....	1
Reporte de verificación de contenido	2
Certificación	3
Responsabilidad de Autoría	4
Autorización de Publicación	5
Dedicatoria.....	6
Agradecimiento.....	7
Dedicatoria.....	8
Agradecimiento.....	9
Índice de contenidos	10
Índice de figuras	13
Índice de tablas	17
Resumen	18
Abstract.....	19
Capítulo I: Introducción	20
Antecedentes.....	20
Planteamiento del problema y justificación.....	21
Hipótesis	23
Objetivos	23
<i>Objetivo general</i>	23
<i>Objetivos específicos</i>	23
Capítulo II: Fundamento Teórico.....	24
Lewansoul Brazo 6DOF	24
Robot cartesiano	26

Banda transportadora.....	27
Arduino Mega	28
Shield Ethernet W5100 Arduino	29
Controlador lógico programable (PLC)	30
CPU 1214C DC/DC/DC	31
Photo sensor PE-R05D.....	32
Módulo controlador de motor paso a paso TB6560 3A.....	33
Motor NEMA 17	34
Capítulo III: Desarrollo del sistema	36
Construcción de la estructura metálica para la estación de automatización.	36
Diseño de tablero de control de la planta	38
Diseño de tablero para estantería móvil	39
Montaje de robots y bandas en la estructura metálica mesa	40
Selección e instalación de los sensores.....	41
Conexión eléctrica de motores paso a paso Nema 17 y Driver TB6560	44
Implementación de un robot para el transporte de jabas	45
Conexión eléctrica de LEWANSOUL brazo robótico 6 DOF y placa Arduino Mega.....	46
Diseño de la etapa control	47
<i>Diseño de Diagrama Grafcet</i>	<i>47</i>
<i>Configuración de propiedades Ethernet del ordenador.....</i>	<i>49</i>
<i>Creación de un nuevo proyecto en TIA PORTAL V16.....</i>	<i>52</i>
<i>Creación y configuración de objetos tecnológicos</i>	<i>56</i>
Programación del funcionamiento en modo manual en el bloque Main.....	66
<i>Energizado del sistema.</i>	<i>70</i>
<i>Encendido de motores 1, 2, 3 y 4.</i>	<i>70</i>
<i>Posicionamiento Home de los motores 1, 2, 3 y 4.</i>	<i>71</i>

<i>Funcionamiento JOG de motores 1, 2, 3 y 4.</i>	71
<i>Modo Relative.</i>	72
<i>Modo Absolute.</i>	73
<i>Etapa 1.</i>	75
<i>Etapa 2.</i>	75
<i>Etapa 3.</i>	77
<i>Etapa 4.</i>	77
<i>Retorno a posiciones de referencia de los motores 1, 2, 3 y 4.</i>	78
<i>Comunicación Modbus PLC-Arduino.</i>	79
Capítulo IV: Pruebas y análisis de resultados	87
Conexión de la estación didáctica a la red de 110VAC	87
Botonera de Inicio, Stop y Paro de Emergencia de la estación didáctica.	88
Respuesta de Etapa 1	89
Respuesta de Etapa 2	92
Respuesta de Etapa 3	97
Respuesta de Etapa 4	99
Comprobación de hipótesis	102
Alcances	103
Limitaciones	103
Capítulo V: Conclusiones y recomendaciones	104
Conclusiones	104
Recomendaciones	105
Bibliografía	106
Anexos	109

ÍNDICE DE FIGURAS

Figura 1 <i>Brazo robótico Lewansoul Brazo 6DOF</i>	25
Figura 2 <i>Robot Cartesiano</i>	27
Figura 3 <i>Banda transportadora</i>	27
Figura 4 <i>Arduino Mega</i>	29
Figura 5 <i>Shield Ethernet W5100 Arduino</i>	30
Figura 6 <i>PLC S7 – 1200</i>	31
Figura 7 <i>Características generales de la CPU 1214C DC/DC/DC</i>	31
Figura 8 <i>Características de entradas y salidas de la CPU 1214C DC/DC/DC</i>	32
Figura 9 <i>Sensor Óptico PE-R05D</i>	33
Figura 10 <i>Driver TB6560</i>	34
Figura 11 <i>Motor NEMA 17</i>	35
Figura 12 <i>Diseño CAD de estructura metálica</i>	36
Figura 13 <i>Etapa de soldadura</i>	37
Figura 14 <i>Producto final de las estructuras metálicas</i>	38
Figura 15 <i>Diseño de tablero de control de la planta</i>	39
Figura 16 <i>Diseño de tablero para estantería móvil</i>	40
Figura 17 <i>Montaje de robos y bandas en estructura metálica</i>	41
Figura 18 <i>Instalación del sensor PE-R05D</i>	42
Figura 19 <i>Diagrama del circuito de salida del sensor foto eléctrico PE-R05D</i>	42
Figura 20 <i>Conexión de sensores, pulsadores e indicadores al PLC</i>	43
Figura 21 <i>Diagrama de conexiones entre Motor Nema 17, Driver Tb6560 y PLC</i>	44
Figura 22 <i>Implementación de un robot para el movimiento de las jabas</i>	45
Figura 23 <i>Conexión eléctrica de brazo robótico 6 DOF y placa Arduino Mega</i>	46
Figura 24 <i>Diseño simulado de la planta</i>	47
Figura 25 <i>Diagrama Grafcet</i>	49

Figura 26 <i>Búsqueda de la opción "Ver conexiones de red"</i>	50
Figura 27 <i>Selección de tarjeta de red Ethernet de la PC</i>	50
Figura 28 <i>Ventana Propiedades de Ethernet</i>	51
Figura 29 <i>Ping entre PLC y ordenador</i>	52
Figura 30 <i>Creación de nuevo proyecto en TIA PORTAL</i>	53
Figura 31 <i>Configurar un dispositivo en TIA PORTAL</i>	54
Figura 32 <i>Determinar la configuración del dispositivo conectado</i>	54
Figura 33 <i>Detección de hardware para PLC</i>	55
Figura 34 <i>Dispositivo detectado PLC S7-1200 CPU 1214C DC/DC/DC</i>	56
Figura 35 <i>Agregar objeto tecnológico Motion Control</i>	57
Figura 36 <i>Configuración de Parámetros Básicos – General</i>	58
Figura 37 <i>Configuración de Parámetros Básicos – Accionamiento</i>	60
Figura 38 <i>Configuración de micro pasos y corriente máxima del motor Nema 17</i>	61
Figura 39 <i>Cálculo del movimiento de la carga por vuelta del motor</i>	62
Figura 40 <i>Configuración de parámetros avanzados</i>	63
Figura 41 <i>Configuración de la Dinámica del objeto tecnológico</i>	64
Figura 42 <i>Configuración de para de emergencia del objeto tecnológico</i>	65
Figura 43 <i>Visualización de objetos tecnológicos creados</i>	65
Figura 44 <i>Ubicación de la herramienta Motion Control</i>	67
Figura 45 <i>Bloque MC_POWER</i>	68
Figura 46 <i>Bloque MC_Home</i>	68
Figura 47 <i>Bloque MC_Halt</i>	68
Figura 48 <i>Bloque MC_MoveRelative</i>	69
Figura 49 <i>Bloque MC_MoveAbsolute</i>	69
Figura 50 <i>Segmento Energizado del sistema</i>	70
Figura 51 <i>Segmento Encendido de motores 1, 2, 3 y 4</i>	70

Figura 52 Segmento de Posicionamiento Home de los motores 1, 2, 3 y 4	71
Figura 53 Segmento de Funcionamiento JOG de motores 1, 2, 3 y 4	72
Figura 54 Segmento de Modo Relative	73
Figura 55 Segmento Modo absoluto	74
Figura 56 Segmento Paro de emergencia de motores	74
Figura 57 Segmento Etapa 1	75
Figura 58 Segmento Etapa 2	76
Figura 59 Segmento Etapa 3	77
Figura 60 Segmento Etapa 4	78
Figura 61 Segmento Retorno a posición home	78
Figura 62 Herramienta Modbus TCP	79
Figura 63 Bloque MB_CLIENT	80
Figura 64 Creación de un Bloque Base de Datos	81
Figura 65 Variables creadas en Base de datos MODBUS_CONFIGURATION_DB	81
Figura 66 Configuración de variable LEER_CONNECT	83
Figura 67 Configuración de variable CONNECT_ESCRIBIR	83
Figura 68 Identificador HW del submódulo de interfaz IE	84
Figura 69 Creación de base de datos MODBUS_DATA_DB	85
Figura 70 Creación de Array Recibidos y Enviados	85
Figura 71 Segmento Comunicaciones MODBUS TPC/IP entre PLC- Arduino	86
Figura 72 Selector en Modo Manual	87
Figura 73 Conexión de la estación didáctica a la RED de 110VAC	88
Figura 74 Accionamiento de botón INICIO	88
Figura 75 Accionamiento de botón STOP	89
Figura 76 Accionamiento de botón PARO DE EMERGENCIA	89
Figura 77 Indicador de ETAPA 1	90

Figura 78 <i>Envase seleccionado para pruebas de funcionamiento</i>	90
Figura 79 <i>Sensado de envase en Etapa 1</i>	91
Figura 80 <i>Gráfica de pastel de resultados obtenidos de Etapa 1</i>	92
Figura 81 <i>Indicador de ETAPA 2</i>	93
Figura 82 <i>Posiciones de la jaba</i>	93
Figura 83 <i>Agarre perfecto del envase</i>	96
Figura 84 <i>Grafica de barras agrupadas de resultados obtenidos de Etapa 2</i>	96
Figura 85 <i>Indicador de ETAPA 3</i>	97
Figura 86 <i>Sensado de jaba por sensor PE-R05D</i>	98
Figura 87 <i>Grafica de pastel de resultados obtenidos de Etapa 3</i>	98
Figura 88 <i>Indicador ETAPA 4</i>	99
Figura 89 <i>Comprobación de comunicación mediante MODBUS_DATA_DB</i>	100
Figura 90 <i>Levantamiento de jaba con robot manipulador</i>	100
Figura 91 <i>Gráfica de barras de Resultados obtenidos de Etapa 4</i>	102

ÍNDICE DE TABLAS

Tabla 1 <i>Parámetros del Bus servo</i>	25
Tabla 2 <i>Especificaciones del brazo robótico</i>	26
Tabla 3 <i>Características del sensor Foto eléctrico</i>	33
Tabla 4 <i>Valores para configuración de Parámetros Básicos sección Accionamiento</i>	58
Tabla 5 <i>Valores para la configuración de parámetros avanzados sección Mecánica</i>	60
Tabla 6 <i>Descripción MB_CLIENT</i>	82
Tabla 7 <i>Resultados de funcionamiento de Etapa 1</i>	91
Tabla 8 <i>Resultados de funcionamiento de Etapa 2</i>	94
Tabla 9 <i>Resultados de funcionamiento de Etapa 4</i>	101

Resumen

La presente tesina tiene como objetivo el diseño e implementación de la estructura metálica, los equipos e instrumentos para el funcionamiento en modo manual del sistema de transporte y organización de envases para bebidas, esto se lo va a implementar en un controlador S7-1200 CPU 1214C DC/DC/DC de la marca Siemens, por su robustez, flexibilidad, capacidad de controlar diferentes procesos industriales y sobre todo por permitir el control de motores paso a paso Nema mediante sus salidas transistorizadas que trabajan a una frecuencia de 20khz. Se implementará el modo manual de la etapa 1 y 3, transporte de envases y jabs mediante el control de la velocidad y sentido del motor nema, en la etapa 2, organización de envases, se realizará el control del robot cartesiano para el posicionamiento preciso del gripper y posterior traslado de envases y finalmente en la etapa 4, el accionamiento del robot manipulador Lewansoul mediante comunicación MODBUS TCP/IP que usará el PLC y la tarjeta Arduino. Los sensores y actuadores a utilizar son de fácil adquisición en el mercado y son compatibles para ser usados como entradas y salidas del controlador. Por último, se prioriza la seguridad en general mediante el paro de emergencia y el stop para que el estudiante tenga la facilidad de realizar prácticas de una manera segura y fortalecer sus conocimientos en el aprendizaje de procesos batch.

Palabras clave: modo manual, motor paso a paso, salidas transistorizadas, procesos batch.

Abstract

The objective of this thesis is the design and implementation of the metal structure, equipment and instruments for the manual operation of the transportation and organization system for beverage containers. This will be implemented on an S7-1200 CPU 1214C controller. . DC/DC/DC from the Siemens brand, for its robustness, flexibility, ability to control different industrial processes and above all for allowing the control of Nema stepper motors through its transistorized outputs that work at a frequency of 20khz. The manual mode of stage 1 and 3, transport of containers and bags, will be implemented by controlling the speed and direction of the nema motor, in stage 2, organization of containers, the control of the Cartesian robot for precise positioning will be carried out. of the gripper and subsequent transfer of containers and finally in stage 4, the operation of the Lewansoul manipulator robot through MODBUS TCP/IP communication that will use the PLC and the Arduino card. The sensors and actuators to be used are easily available on the market and are compatible to be used as inputs and outputs of the controller. Finally, safety in general is prioritized through stops and emergency stops so that the student can easily carry out the practices safely and strengthen their knowledge in learning batch processes.

Keywords: manual mode, stepper motor, transistorized outputs, batch processes.

Capítulo I

Introducción

Antecedentes

Hoy la automatización de procesos industriales es muy importante, las tareas tediosas y peligrosas las realizan 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.

Prácticamente todas las industrias alrededor del mundo tienen un pequeño sistema automático. Estos sistemas son esenciales para agilizar sus operaciones y mantenerse competitivas en un entorno empresarial cada vez más globalizado y exigente. Desde la fabricación de productos hasta la gestión de la cadena de suministro, la automatización desempeña un papel fundamental. Los robots, sensores y software especializados permiten a las empresas optimizar procesos, reducir costos y mejorar la calidad de sus productos y servicios. Además, la automatización también contribuye a mejorar las condiciones de trabajo al asumir tareas peligrosas o repetitivas, liberando a los empleados para concentrarse en actividades más estratégicas e innovadoras (Jaramillo, 2015).

La automatización del embotellado en Ecuador es un proceso que ha experimentado un avance significativo en los últimos años (Núñez Solano, 2017). A día de hoy, el transporte y la organización de envases para bebidas en Ecuador se lleva a cabo principalmente de manera manual o con la ayuda de equipos convencionales, lo que conlleva largos periodos de entrega, pérdida eventual de producto y un alto costo operativo (Cadena, 2016). La automatización de este proceso podría ofrecer numerosas ventajas, como la reducción de errores humanos, la optimización de la capacidad de carga y la mejora de la velocidad del motor en la cinta transportadora (Naranjo, 2019). Este cambio hacia la automatización no solo mejoraría la

eficiencia del proceso, sino que también tendría un impacto positivo en la productividad y la rentabilidad de la industria del embotellado en Ecuador.

La provincia de Cotopaxi ha adoptado tecnologías de automatización en industrias agrícolas y de manufactura (Caisabanda & Choloquina, 2022), mediante la implementación de maquinaria actualizada lo cual permite una mayor eficiencia en el diseño, producción y distribución de sus productos y soluciones tecnológicas (Tufiño et al., 2023). Pero pese al esfuerzo realizado por las empresas, es importante enmarcar que la provincia aun presenta un gran desafío como falta de programas de capacitación especializada y la falta de implementación de procesos autónomos en las etapas de fabricación (Iza Laica & Tapia Velásquez, 2023).

Los laboratorios de la carrera de Electrónica y Automatización de la Universidad de las Fuerzas Armadas ESPE Sede Latacunga cuentan con una estación didáctica diseñada para la automatización del transporte y la organización de envases de granulados o de cereales. Este sistema servirá como punto de referencia fundamental en el desarrollo de una estación didáctica destinada a la automatización del proceso de envasado de bebidas. En este contexto, se propone la planificación y ejecución del diseño e implementación de dicha estación, con el objetivo de proporcionar a los estudiantes de la carrera una herramienta esencial para la realización de prácticas en el campo de la automatización. Esta iniciativa se considera de gran relevancia para el desarrollo de habilidades y competencias necesarias en la formación profesional de los estudiantes.

Planteamiento del problema y justificación

La carrera de Ingeniería en Electrónica y Automatización de la Universidad de las Fuerzas Armadas ESPE Sede Latacunga se enfrenta actualmente a un desafío importante: la falta de equipamiento adecuado para la enseñanza de las nuevas asignaturas relacionadas con la Automatización Industrial. Esta carencia dificulta el proceso de enseñanza-aprendizaje

práctico, ya que los estudiantes no tienen acceso a las herramientas necesarias para adquirir experiencia en el campo de la automatización de procesos industriales. Para superar esta limitación, es imprescindible que la universidad invierta en la adquisición de estaciones didácticas que permitan a los estudiantes poner en práctica los conocimientos teóricos y desarrollar habilidades técnicas relevantes para su futura carrera profesional.

La implementación de estaciones didácticas para la automatización de procesos industriales sería de gran beneficio para los estudiantes de la carrera de Ingeniería en Electrónica y Automatización. Estas estaciones proporcionarían un entorno de aprendizaje práctico donde los estudiantes podrían familiarizarse con los diferentes componentes y sistemas utilizados en la automatización industrial, como sensores, actuadores, controladores y sistemas de comunicación. Además, les permitiría diseñar, simular y poner en funcionamiento sistemas automatizados, lo que fortalecería su comprensión de los conceptos teóricos y les brindaría una experiencia valiosa para su futura inserción laboral en el campo de la automatización industrial. En resumen, el diseño de estaciones didácticas para la automatización de procesos industriales es esencial para mejorar la calidad de la enseñanza en la carrera de Ingeniería en Electrónica y Automatización.

Los estudiantes de la carrera de Electrónica y Automatización requieren una sólida formación práctica en este campo, sin embargo, la ausencia de un entorno de aprendizaje especializado les limita la oportunidad de adquirir las habilidades necesarias para abordar los desafíos técnicos y operativos inherentes a la automatización de envasado de bebidas. El diseño y la implementación de la estación proporcionarán a los estudiantes un entorno de aprendizaje práctico y realista donde puedan aplicar conceptos teóricos, desarrollar habilidades de programación y control, y comprender los aspectos técnicos y operativos de la automatización en este contexto.

Hipótesis

El diseño e implementación de una estación didáctica de transporte y organización de envases, permitirá el control del sistema.

Objetivos

Objetivo general

- Diseñar e implementar la estructura metálica, los equipos e instrumentos para el funcionamiento en modo manual del sistema de transporte y organización de envases para bebidas.

Objetivos específicos

- Construir la estructura metálica para la estación de automatización.
- Seleccionar e instalar sensores en las etapas del sistema de transporte y organización de envases para bebidas.
- Implementar el sistema de transporte de envases (banda transportadora más sistema de control).
- Implementar el sistema de control de movimiento para la organización de los envases para bebidas en las jabs.
- Implementar el sistema de transporte de jabs (banda transportadora más sistema de control).
- Implementar un robot para el movimiento de las jabs hacia el área de despacho.
- Diseñar los esquemas eléctricos del sistema de automatización del transporte y organización de envases para bebidas.
- Realizar pruebas de funcionamiento en modo manual del sistema de automatización del transporte y organización de envases para bebidas.

Capítulo II

Fundamento Teórico

En este capítulo se presenta información que se ha obtenido de libros, investigaciones, publicaciones y diferentes bases digitales enfocadas a la educación e investigación. Todo ello con la finalidad de ayudar en la comprensión de la tesina a desarrollar.

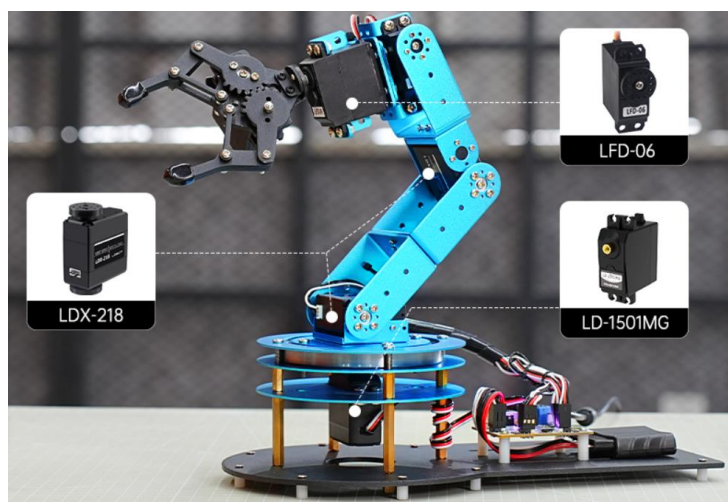
Lewansoul Brazo 6DOF

Los brazos robóticos, que tienen su origen en los brazos humanos, se pueden utilizar para mover objetos, tienen una amplia gama de tamaños y propósitos. Pueden realizar trabajos de mantenimiento y están equipados con herramientas que comúnmente son ventosas y grippers. (ESA Education, 2020)

El Lewansoul Brazo 6DOF emplea un motor de bus serie con retroalimentación de ángulo, temperatura y voltaje, así como protección de bloqueo del eje del motor. El servomotor serie LX-15D cuenta con una vida útil prolongada y puede operar de forma ininterrumpida durante al menos un mes. Fabricado con piezas de excelente calidad en todos los aspectos, fabricación completamente metálica, diseño exclusivo de la biela y superficie tratada con oxidación mediante chorro de arena, contribuyen a que su apariencia sea más atractiva y refinada como se lo observa en la Figura 1.

Figura 1

Brazo robótico Lewansoul Brazo 6DOF.



Nota. Tomado de (LewanSoul, 2023)

Ofrece diversos métodos de programación, incluyendo la programación de computadoras, la programación de teléfonos móviles y la programación manual de posiciones. (LewanSoul, 2023) El brazo robótico Lewansoul Brazo 6DO, cuenta con tres tipos de servo, los cuales han sido distribuido como se lo muestra en la Figura 1. Cada servo cuenta con características y especificaciones diferentes, esto se lo puede observar en la Tabla 1.

Tabla 1

Parámetros del Bus servo

PARÁMETROS DE SERVO BUS				
Nombre de servo	Velocidad de rotación	Par torque	Rango de rotación	Características
LDX-218	0.16 seg/60° 7.4V	15kg.cm 6V 17kg.cm 7.4V	0°-180°	-Engranaje metálica. -Doble eje
LD-1501MG	0.16 seg/60° 7.4V	15kg.cm 6.5V 17kg.cm 7.4V	0°-180°	-Engranaje metálica. -Doble eje
LFD-06	0.25 seg/60° 7.4V	6kg.cm 7.4V	0°-180°	-Engranaje metálica. -Doble eje

Nota. Tomado de (LewanSoul, 2023)

Las especificaciones del brazo robótico se presentan en la Tabla 2.

Tabla 2

Especificaciones del brazo robótico

DETALLE	ESPECIFICACIONES
Dimensión de producto	285*120*465 mm
Peso del Producto	1.24 kg
Material	Aleación de aluminio
DOF	6DOF (5DOF + Gripper)
Fuente de alimentación	7.5V 3A DC Adaptador de corriente
Sistema controlador	6 canales Servo Controlador
Servo	LDX-218, LD-1501MG, LFD-06
Métodos de controls	PC, APP, Wireless Handle
Comunicación	Bluetooth, USB
Tamaño del paquete	335*290*85mm
Peso del paquete	2kg

Nota. Tomado de (LewanSoul, 2023)

Robot cartesiano

Un robot cartesiano, también conocido como robot lineal, es una máquina que se mueve a lo largo de tres ejes cartesianos (X, Y, Z). Esto significa que pueden moverse de arriba a abajo, de afuera hacia adentro y de lado a lado. Estos robots se utilizan a menudo para automatizar procesos industriales como el montaje, la soldadura y la manipulación de materiales. A diferencia de otros tipos de robots, como los robots escénicos, los robots cartesianos son modulares y constan de una serie de varillas y piezas conectadas como se muestra en la Figura 2. Esto los hace fáciles de montar y mantener, lo que los convierte en una opción popular para muchas empresas. (Muñoz, 2023)

Figura 2*Robot Cartesiano*

Nota. Tomado de (Muñoz, 2023)

Banda transportadora

La cinta transportadora consiste en una banda continua que se desliza sobre rodillos giratorios que están soportados por un bastidor resistente. El accionamiento se realiza por fricción entre los tambores (accionados estos por un motor eléctrico normalmente) y la banda transportadora. Las bandas transportadoras son equipos que utilizan el principio de transporte continuo y resuelven problemas básicos del transporte de elementos. (Andrea, 1980)

Las cintas transportadoras se usan principalmente para transportar materiales granulados, agrícolas e industriales, tales como cereales, carbón, minerales, etcétera.

Figura 3*Banda transportadora*

Nota. Tomado de (Turbosquid, 2017)

Arduino Mega

Uno de los miembros más potentes de la familia Arduino es el Arduino Mega, caracterizado por su robustez y una cantidad considerable de pines de entrada y salida, lo que lo hace ideal para proyectos más complejos que requieren múltiples sensores y actuadores. El Mega es compatible con los microcontroladores ATmega2560 y ATmega1280, lo que le proporciona capacidades ampliadas en comparación con otros modelos como el Arduino Uno (Dolinay et al., 2021)

El Arduino Mega 2560 es una placa controladora basada en el microcontrolador ATmega2560. Tiene un total de 54 pines digitales para entradas y salidas (15 de las cuales pueden generar pulsos PWM), 16 entradas analógicas, 4 puertos UART por hardware, un conector USB para programación y alimentación, una entrada de jack para alimentación externa, pines ICSP y un botón de Reset. (González, 2024)

Características de Arduino MEGA 2560

- Microcontrolador: ATmega2560
- Tensión de alimentación: 5V
- Tensión de entrada recomendada: 7-12V
- Límite de entrada: 6-20V
- Pines digitales: 54 (14 con PWM)
- Entradas analógicas: 16
- Corriente máxima por pin: 40 mA
- Corriente máxima para el pin 3.3V: 50 mA
- Memoria flash: 256 KB
- SRAM: 8 KB
- EEPROM: 4 KB
- Velocidad de reloj: 16 MHz

Figura 4

Arduino Mega

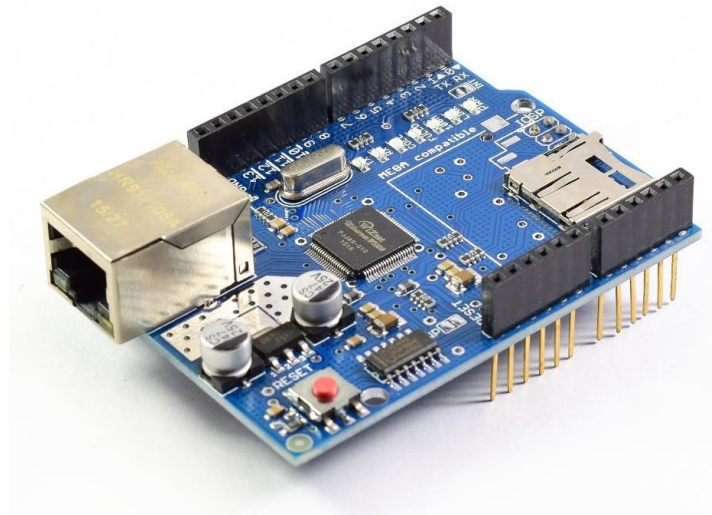


Nota. Tomado de (González, 2024)

Shield Ethernet W5100 Arduino

El Shield Ethernet W5100 se caracteriza por su chip Wiznet W5100 que facilita el acceso a la red con velocidades de conexión de 10/100Mb y la capacidad de manejar hasta cuatro conexiones de socket simultáneas (Gogineni et al., 2015). Este módulo se monta directamente sobre la placa Arduino, y la comunicación entre ambos se realiza mediante un cable de red, lo que permite el monitoreo y control de dispositivos de forma remota (Santos et al., 2021).

Uno de los aspectos más destacados del Shield Ethernet W5100 es su versatilidad y facilidad de uso. Se ha documentado que la configuración inicial para conectar el módulo y la placa Arduino a la red es un proceso relativamente sencillo, que se puede realizar en pocos minutos con la ayuda de bibliotecas disponibles en el entorno de desarrollo de Arduino (Ayuningtyas & Ilman, 2021). Este aspecto es particularmente importante para aquellos que están comenzando en el campo de la automatización y desean implementar soluciones IoT (Internet de las Cosas) sin tener que invertir en un conocimiento técnico avanzado.

Figura 5*Shield Ethernet W5100 Arduino*

Nota. Tomado de (Naylamp, 2024)

Controlador lógico programable (PLC)

Según (Cera-Martínez et al., 2018) Los Automatas Programables Lógicos (PLC, por sus siglas en inglés) son una pieza central en este avance tecnológico, ofreciendo robustez, flexibilidad y eficiencia en una amplia gama de aplicaciones industriales.

Según (Abad et al., 2023) La automatización mediante PLC implica la utilización de sistemas que funcionan sin la intervención directa del operador, excepto para modificar parámetros del proceso cuando sea necesario. Estos sistemas no solo continúan operando de manera autónoma, sino que también recopilan información vital sobre el desarrollo del proceso, lo que posteriormente facilita la implementación de mejoras, la detección de fallas y una intervención rápida y precisa.

Figura 6

PLC S7 – 1200



Nota. Tomado de (SIEMENS, 2023)

CPU 1214C DC/DC/DC

El PLC S7-1200 con la CPU 1214c DC/DC/DC posee características generales que se presenta en la Figura 7. (SIEMENS, 2023)

Figura 7

Características generales de la CPU 1214C DC/DC/DC

Información general	
Designación del tipo de producto	CPU 1214C DC/DC/DC
Versión de firmware	V4.6
Ingeniería con	
<ul style="list-style-type: none"> Paquete de programación 	STEP 7 V18 o superior
Tensión de alimentación	
Valor nominal (DC)	
<ul style="list-style-type: none"> 24 V DC 	Sí
Rango admisible, límite inferior (DC)	20,4 V
Rango admisible, límite superior (DC)	28,8 V
Protección contra inversión de polaridad	Sí
Tensión de carga L+	
<ul style="list-style-type: none"> Valor nominal (DC) Rango admisible, límite inferior (DC) Rango admisible, límite superior (DC) 	24 V 20,4 V 28,8 V

Nota. Tomado de (SIEMENS, 2023)

En cuanto a las características que presentan las entradas y salidas se detallan en la Figura 8.

Figura 8**Características de entradas y salidas de la CPU 1214C DC/DC/DC**

Entradas digitales	
Nº de entradas digitales	14; integrado
• De ellas, entradas usable para funciones tecnológicas	6; HSC (High Speed Counting)
Fuente/sumidero (M/P)	Sí
Número de entradas atacables simultáneamente	
Todas las posiciones de montaje	
— hasta 40 °C, máx.	14
Tensión de entrada	
• Valor nominal (DC)	24 V
• para señal "0"	5 V DC, con 1 mA
• para señal "1"	15 V DC at 2,5 mA
Retardo a la entrada (a tensión nominal de entrada)	
para entradas estándar	
— parametrizable	0,2 ms, 0,4 ms, 0,8 ms, 1,6 ms, 3,2 ms, 6,4 ms y 12,8 ms, elegible en grupos de 4
— en transición "0" a "1", máx.	0,2 ms
— en transición "0" a "1", máx.	12,8 ms
para entradas de alarmas	
— parametrizable	Sí
para funciones tecnológicas	
— parametrizable	Monofásica: 3 @ 100 kHz y 3 @ 30 kHz, Diferencial: 3 @ 80 kHz y 3 @ 30 kHz
Salidas digitales	
Número de salidas	10
• de ellas, salidas rápidas	4; Salida de tren de impulsos 100 kHz
Limitación de la sobretensión inductiva de corte a	L+ (-48 V)
Poder de corte de las salidas	
• con carga resistiva, máx.	0,5 A
• con carga tipo lámpara, máx.	5 W
Tensión de salida	
• para señal "0", máx.	0,1 V; con carga de 10 kOhm
• para señal "1", mín.	20 V
Intensidad de salida	

Nota. Tomado de (SIEMENS, 2023)

Photo sensor PE-R05D

Los avances en la tecnología de sensores ópticos han permitido el desarrollo de dispositivos cada vez más precisos y versátiles, adecuados para una amplia gama de aplicaciones, desde la detección ambiental hasta la robótica móvil. La comunicación por luz visible (VLC) es un área que ha experimentado avances significativos, estos sensores detectan la intensidad de la luz entrante (Yamazato & Haruyama, 2014).

Figura 9*Sensor Óptico PE-R05D*

Nota. Tomado de (HANYOUNG NUX, 2012)

Las características del sensor se presentan en la Tabla 3.

Tabla 3*Características del sensor Foto eléctrico*

SENSOR FOTO ELÉCTRICO						
MARCA	MODELO	DETECCIÓN	RANGO	ALIMENTACIÓN	MODO DE OPERACIÓN	POTENCIA
HANYOUNG	PE-R05D	REFLEXIÓN	50 cm	AC/DC 24V,	Light: ON	2 W
NUX		DIFUSA		50/60 Hz		

Nota. Tomado de (HANYOUNG NUX, 2012)

Módulo controlador de motor paso a paso TB6560 3A

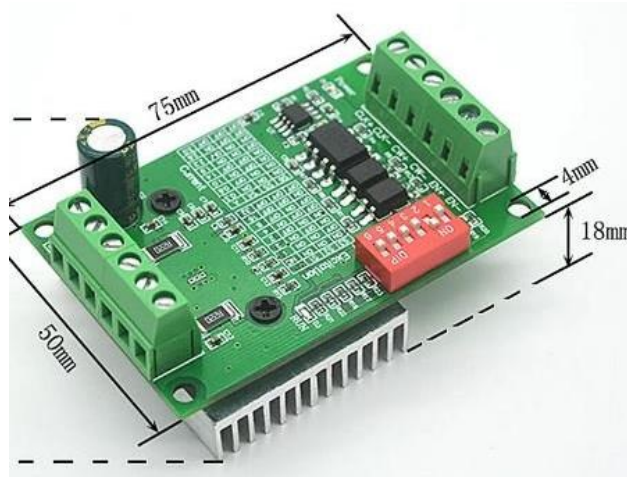
El controlador de motor TB6560 es un componente que funciona para comunicar el controlador con el actuador y fortalecer la señal de salida del controlador para que el actuador pueda leerla.

Driver TB6560 Controlador Motor A Pasos, permite controlar la corriente que se le puede suministrar al motor a través de interruptores y ajustar su número de pasos, es de uso sencillo

y usa un acoplamiento de alta velocidad para garantizar que no haya pérdidas en el control de los pasos que el motor proporciona. Este módulo es ideal para ser acoplado con un microcontrolador o tarjeta de desarrollo, ya que incluye distintas entradas para poder habilitar el motor y controlar la dirección del giro y pasos, estas entradas están aisladas por optoacoplador y se pueden activar con estados lógicos altos y bajos. (Unit Electronics, 2024)

Figura 10

Driver TB6560



Nota. Tomado de (Unit Electronics, 2024)

Motor NEMA 17

NEMA 17 es un motor paso a paso híbrido con un ángulo de paso de $1,8^\circ$ (200 pasos/revolución). Cada fase consume 1,2 A a 4V, lo que permite un par de retención de 3,2 kg-cm. El motor paso a paso NEMA 17 se utiliza generalmente en impresoras, máquinas CNC y cortadoras láser. (PBC Linear, 2020)

Figura 11*Motor NEMA 17*

Nota. Tomado de (PBC Linear, 2020)

Este motor tiene seis cables conectados a dos devanados divididos. Los cables negro, amarillo y verde forman parte del primer devanado, mientras que el rojo, el blanco y el azul forman parte del segundo devanado. (Changzhou, 2020)

Especificaciones técnicas del motor paso a paso NEMA 17

- Tensión nominal: 12 V CC.
- Corriente: 1.2A a 4V
- Ángulo de paso: 1,8 grados.
- No. de Fases: 4
- Longitud del motor: 1,54 pulgadas
- Cable de 4 hilos y 8 pulgadas
- 200 pasos por revolución, 1,8 grados
- Temperatura de funcionamiento: -10 a 40 °C
- Torque de sujeción unipolar: 22,2 oz-pulg.

Capítulo III

Desarrollo del sistema

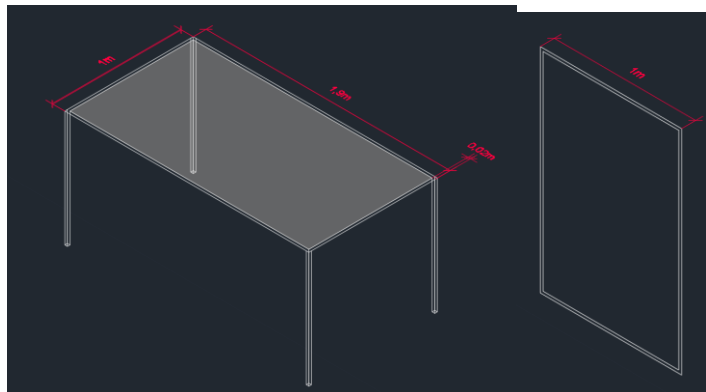
En este capítulo se presenta el desarrollo práctico de la estación, se considera el ensamble de la estructura metálica, la selección e instalación de sensores, instalación del robot cartesiano y el brazo robótico, a la vez se presenta la implementación de la automatización para el funcionamiento de cada etapa en forma independiente.

Construcción de la estructura metálica para la estación de automatización.

Para iniciar, se procede con el desarrollo de un diseño técnico preciso de la estructura metálica, teniendo en cuenta las dimensiones, especificaciones de carga y ubicación de componentes. Para lo cual, se realiza el diseño y simulación en el software AUTOCAD. Se contempla dos piezas en la estructura metálica, la primera es la mesa, la cual servirá como soporte del robot cartesiano, brazo robótico 6DOF, bandas transportadoras, sensores, y tablero de control. La segunda pieza es una estantería móvil en la cual irán colocados el PLC S7-1200 con el módulo de expansión de salidas, la fuente DC, además se incluirá un tablero con conectores banana hembra con la finalidad de dar una funcionalidad Plug-and-Play o PnP. La estructura metálica diseñada en el Software AUTOCAD se lo puede visualizar en la Figura 12.

Figura 12

Diseño CAD de estructura metálica



Una vez finalizado con el diseño de la estructura metálica en un software CAD, se prepara los materiales, se utiliza un tubo cuadrado de 2cm x 2cm para la estructura metálica diseñada, se realiza los respectivos cortes de ancho, profundidad y altura para la mesa, que será base para el posicionamiento del robot cartesiano, las bandas transportadoras, el tablero de control y los elementos utilizados para este proyecto.

Se realiza los cortes para el diseño de la mesa siendo 1 metro de ancho por 1.90 de profundidad y 0.90 metros de altura, una vez realizado los cortes se procede a soldar dando forma a la mesa con el respectivo diseño. A su vez se realiza los respectivos cortes para la estantería metálica donde se implementará el tablero de control.

Seguidamente en la etapa de ensamble se utiliza el proceso de soldadura SMAW, en la cual el metal se funde por el calor generado por un arco formado entre la punta del electrodo y el metal base a unir, para este caso se utiliza el electrodo EW60111/8-1K.

Figura 13

Etapa de soldadura



a. Soldadura Swam

b. electrodo 6011

A continuación, se lijan las estructuras metálicas y se corrigen imperfecciones, para el proceso de pintura se limpia el metal para quitar las impurezas como grasa y restos de suciedad para que la capa de pintura se adhiera al metal, se aplica la pintura especial para proteger y embellecer el metal. Finalmente, como se puede observar en la Figura 14 se tiene las estructuras

metálicas listas para continuar con el proceso de posicionamiento y sujeción de los elementos que conforman el proyecto.

Figura 14

Producto final de las estructuras metálicas

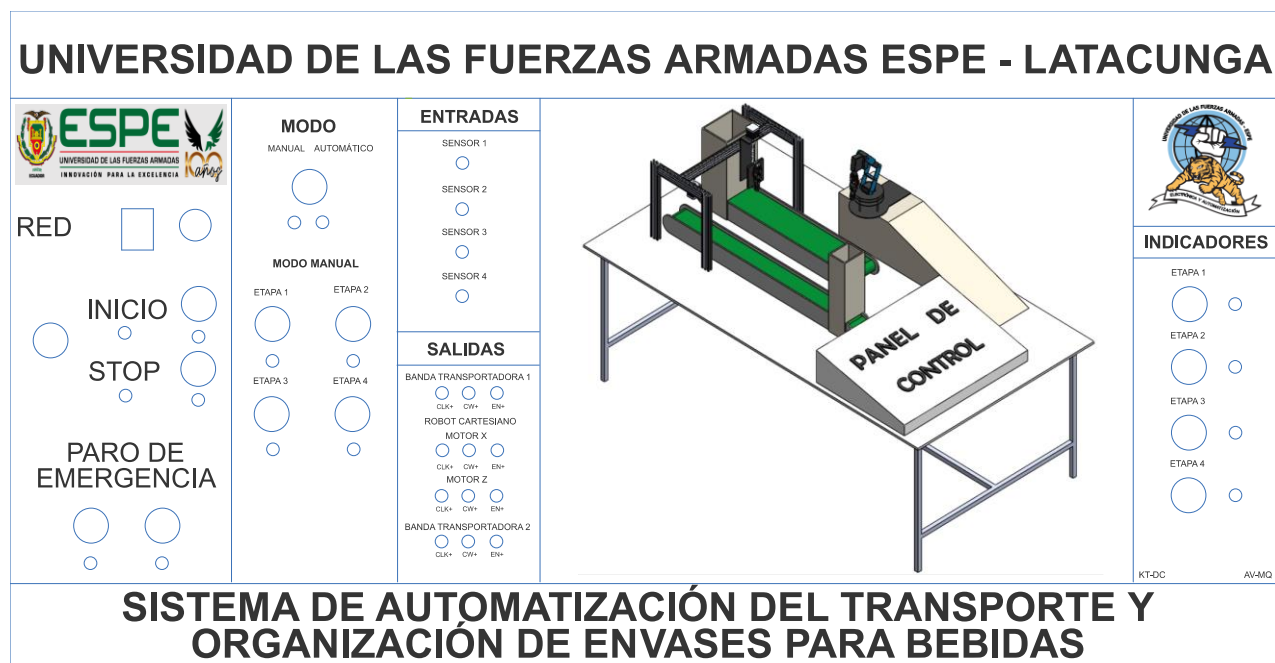


Diseño de tablero de control de la planta

El módulo didáctico contara con la funcionalidad Plug-and-Play, ya que está destinado a ser un módulo de aprendizaje. Es por ello, que se ha considerado que todos los elementos que están ubicados en la estructura metálica (mesa), como sensores, actuadores, pulsadores e indicadores serán conectados a terminales banana hembra de 8mm, tal como se muestra en la Figura 15.

Figura 15

Diseño de tablero de control de la planta



Diseño de tablero para estantería móvil

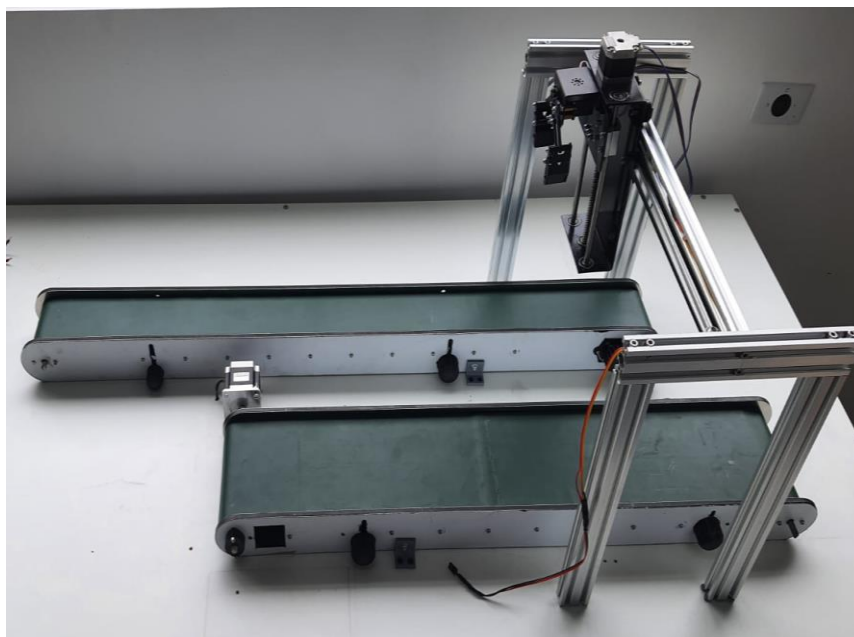
En la estantería móvil se ha considerado ubicar el elemento de control, por lo cual se ve necesario conectar todos los terminales de entradas y salidas del PLC a conectores banana hembra de 8mm para seguir con la funcionalidad Plug-and-Play, tal como se lo puede visualizar en la Figura 16.

Figura 16*Diseño de tablero para estantería móvil***Montaje de robots y bandas en la estructura metálica mesa**

Se realiza perforaciones para ajustar las bases del robot cartesiano y del robot manipulador, las dos bandas transportadoras, la rampa de las jabas y el tablero de control. Se usa pegamento de madera para dar un mayor agarre. El sistema implementado se muestra en la Figura 17.

Figura 17

Montaje de robots y bandas en estructura metálica

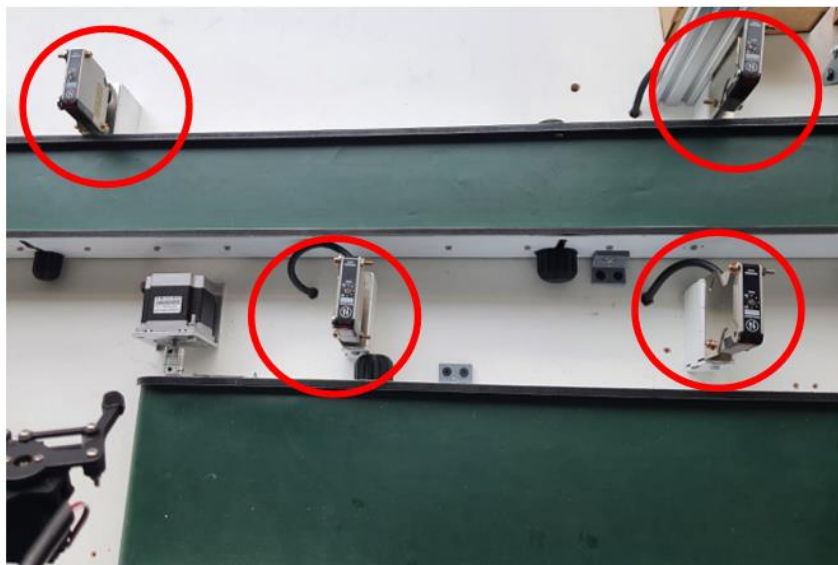


Selección e instalación de los sensores

En este apartado se selecciona el sensor foto eléctrico PE-R05D, el cual va instalado en los extremos de las bandas transportadoras, tienen como función detectar la llegada de un envase nuevo o jaba. Se escoge el sensor PE-R05D por sus buenas características como lo es el rango de sensado de 500 mm, este rango es regulable de acuerdo a las necesidades del entorno, otra razón por la cual se selecciona este sensor es por su bajo consumo de energía de un máximo de 2 W, sobre todo se escoge este sensor por su tipo de alimentación el cual es de 24 VDC, siendo éste, un voltaje común con el que trabajan todos los demás elementos del sistema. Además, cuenta con protección IP54 según la IEC (International Electrotechnical Commission), que indica que es resistente al polvo y a la humedad o contacto al agua.

Figura 18

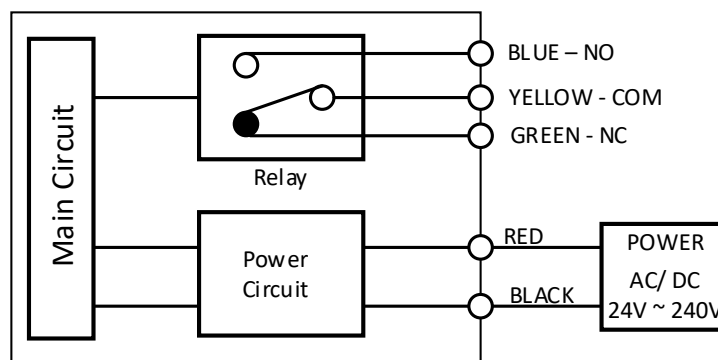
Instalación del sensor PE-R05D



El sensor foto eléctrico PE-R05D cuenta con 5 terminales, los cuales se muestran en la Figura 19.

Figura 19

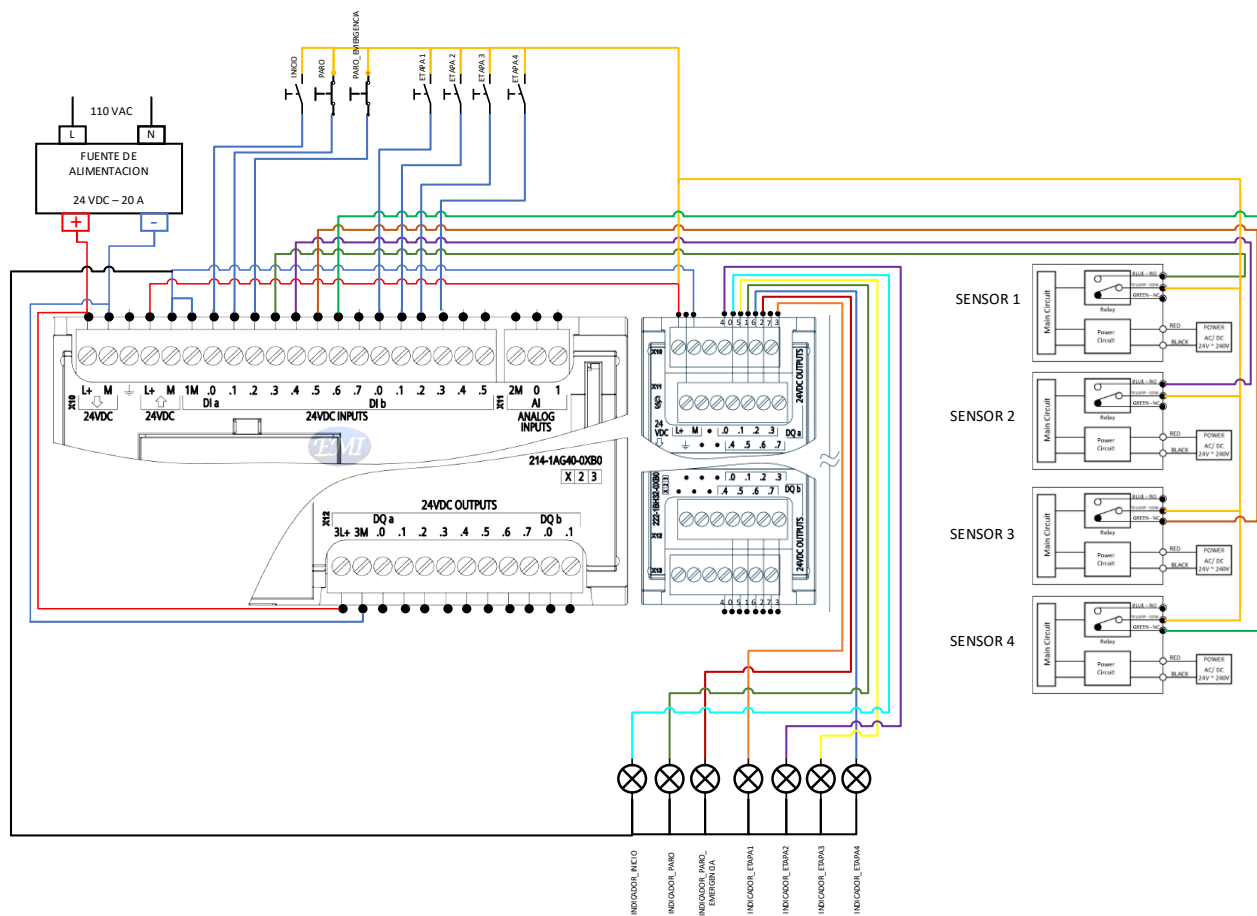
Diagrama del circuito de salida del sensor foto eléctrico PE-R05D



La manera en la que se deben conectar los cuatro sensores al PLC se lo puede ver en la Figura 20, a la vez se visualiza como están conectados los pulsadores de inicio, paro, paro de emergencia y los cuatro botones de las etapas, también se puede visualizar la conexión de las luces indicadoras.

Figura 20

Conexión de sensores, pulsadores e indicadores al PLC

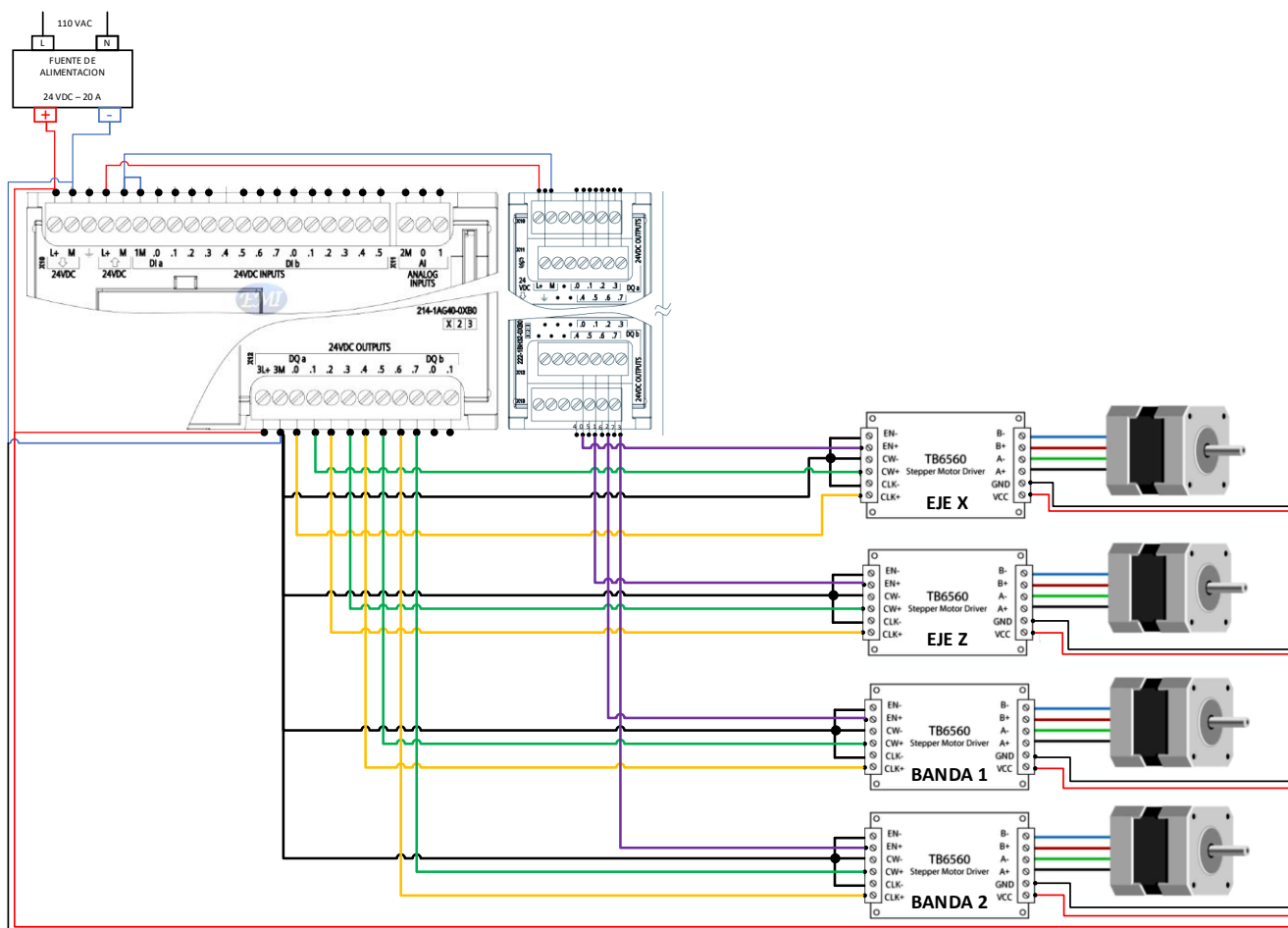


Conexión eléctrica de motores paso a paso Nema 17 y Driver TB6560

Los actuadores que dan movimiento al Eje X, Eje Z del robot, Banda 1 y Banda 2, son los motores paso a paso Nema 17, dicho motor consume 1,2 A. Para la interconexión entre el motor y el PLC se ha usado el Driver TB6560, el cual sirve como interfaz, tomando señales de control del PLC y traduciéndolas en señales de potencia necesarias para controlar el motor paso a paso, además hay que mencionar que el Driver cuenta con protección contra cortocircuitos, protección contra sobrecalentamiento. El diagrama de conexión que se emplea para cada driver se lo puede visualizar en la Figura 21.

Figura 21

Diagrama de conexiones entre Motor Nema 17, Driver Tb6560 y PLC



Implementación de un robot para el movimiento de las jabas hacia el área de despacho.

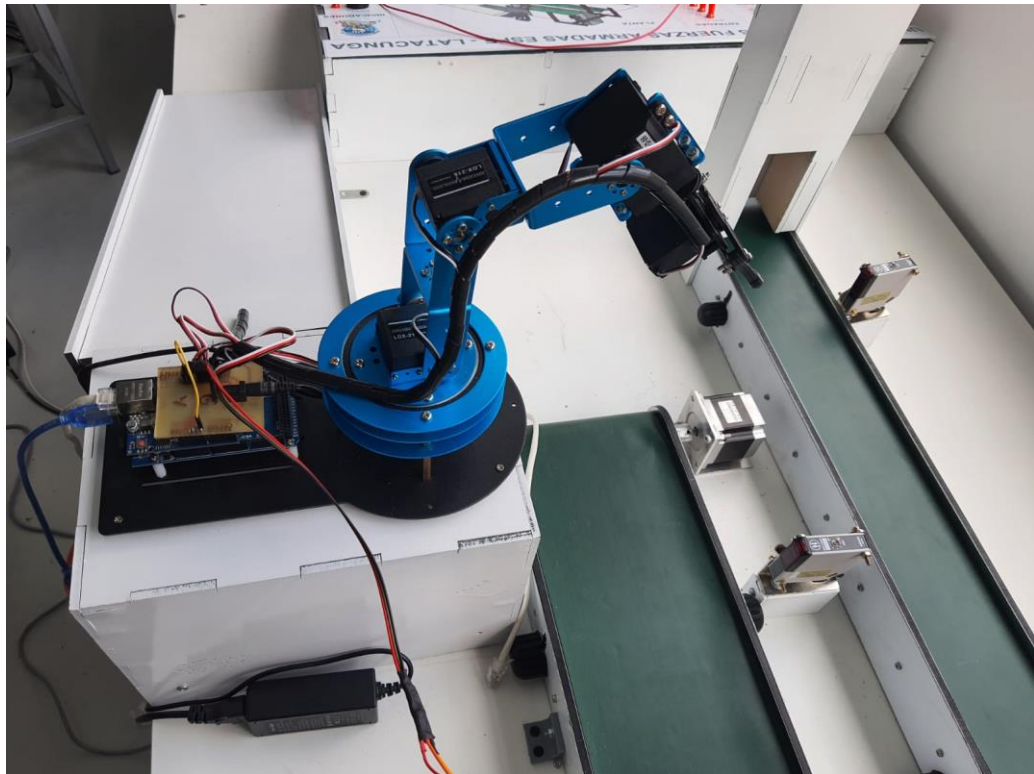
El encargado de mover las jabas hasta el área de despacho, es el robot manipulador Lewansould, que dispone 6 servomotores distribuidos uno en cada articulación, su estructura es de aluminio, el extremo operativo que este robot dispone es un gripper.

Para el control y la programación de la tarea se hace uso de la tarjeta Arduino mega, la tarea que se le va a programar tiene como propósito que el robot levante las jabas llenas de envases desde la banda 2 hasta la rampa de despacho.

El robot cuenta con una base de aluminio la cual posee orificios los cuales fueron de gran ayuda al momento de fijar el robot en su área de trabajo. El resultado final de la implementación se lo puede visualizar en la Figura 22.

Figura 22

Implementación de un robot para el movimiento de las jabas

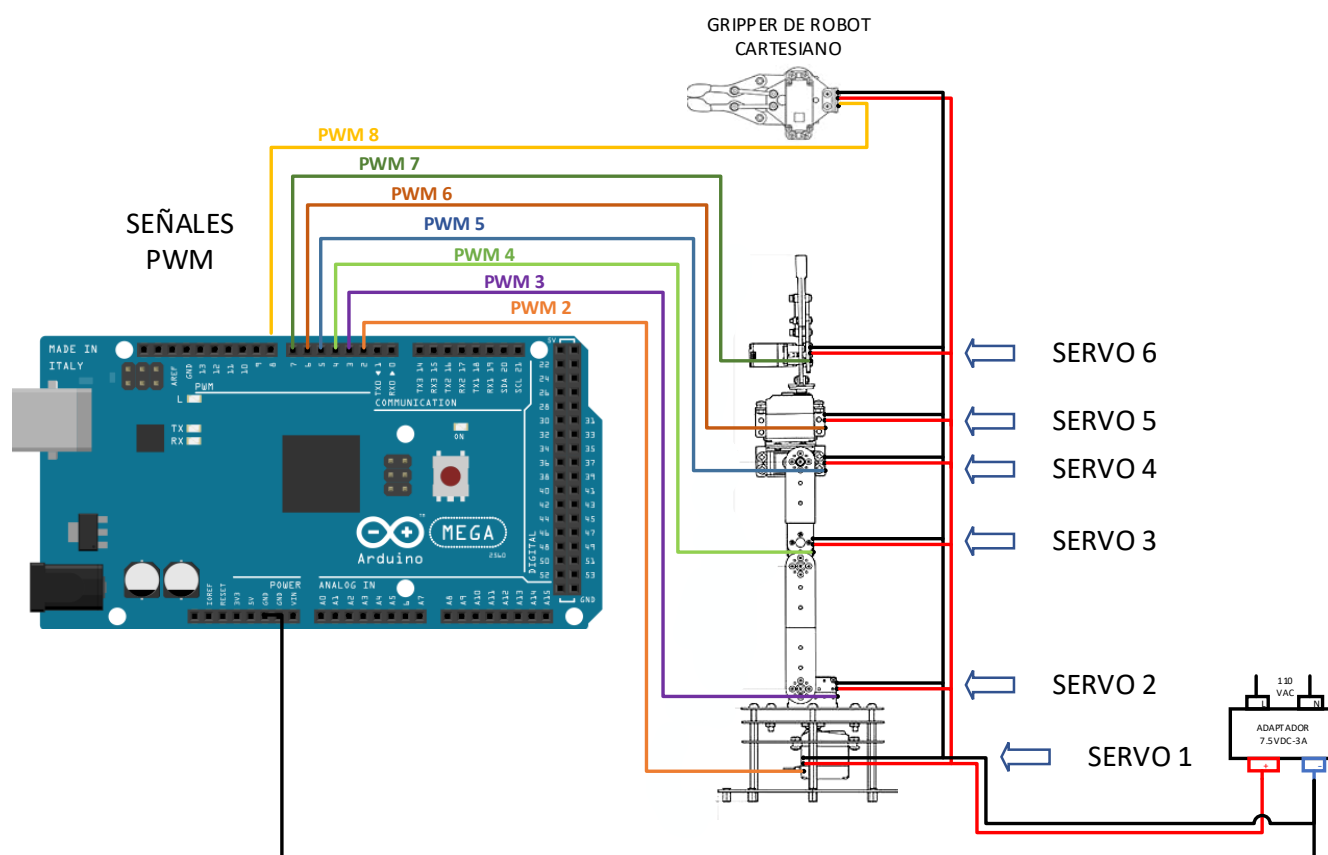


Conexión eléctrica de LEWANSOUL brazo robótico 6 DOF y placa Arduino Mega

El brazo Lewansoul 6DOF posee 6 servomotores que funcionan con una alimentación de 7.5 VDC, el control de este robot recae en la placa Arduino mega, que a la vez realizara el control del gripper del robot cartesiano, se utiliza el adaptador propio del robot para suministrar el voltaje requerido. El diagrama de conexiones se presenta en la Figura 23.

Figura 23

Conexión eléctrica de LEWANSOUL brazo robótico 6 DOF y placa Arduino Mega



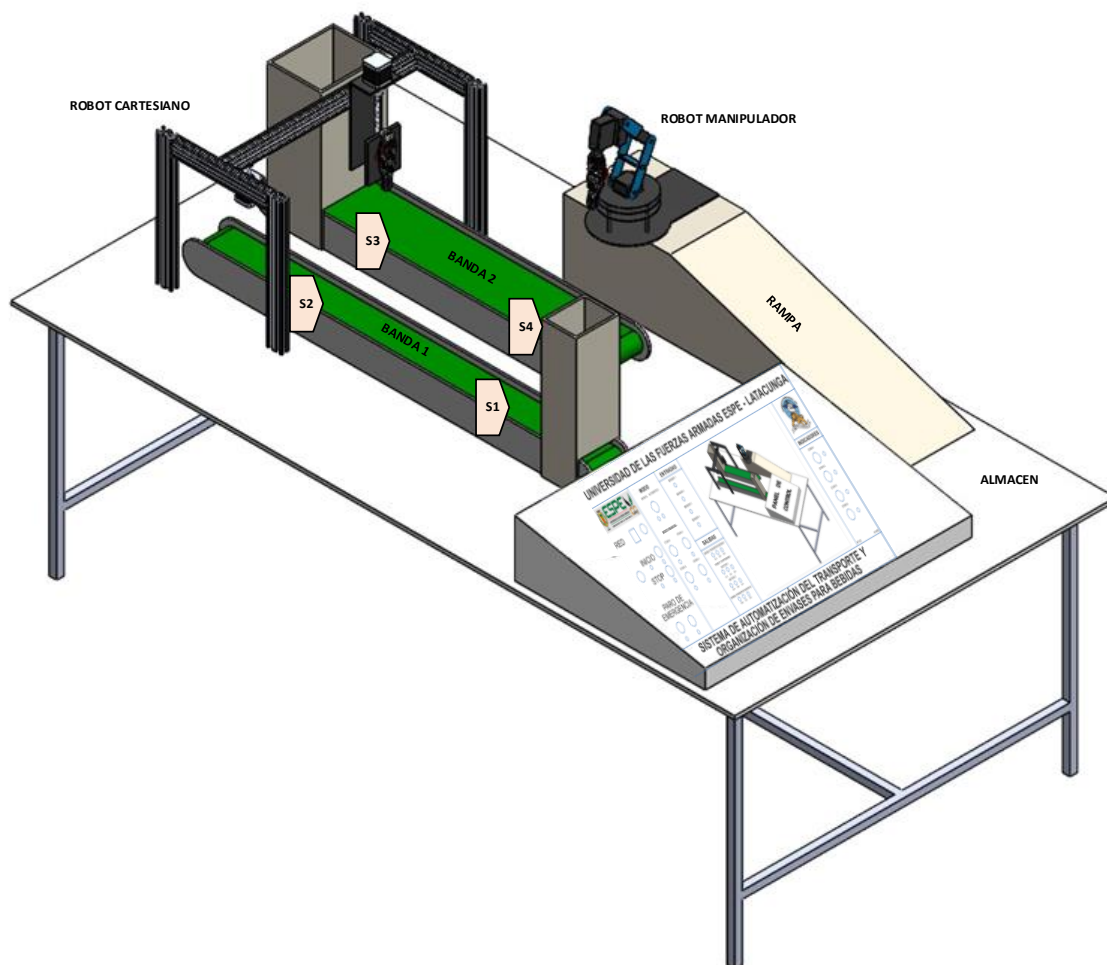
Diseño de la etapa control

Diseño de Diagrama Grafcet

Para desarrollar el diagrama, es primordial conocer el proceso, es por ello que en la Figura 24, se muestra el diseño estructural de la planta a ser automatizada.

Figura 24

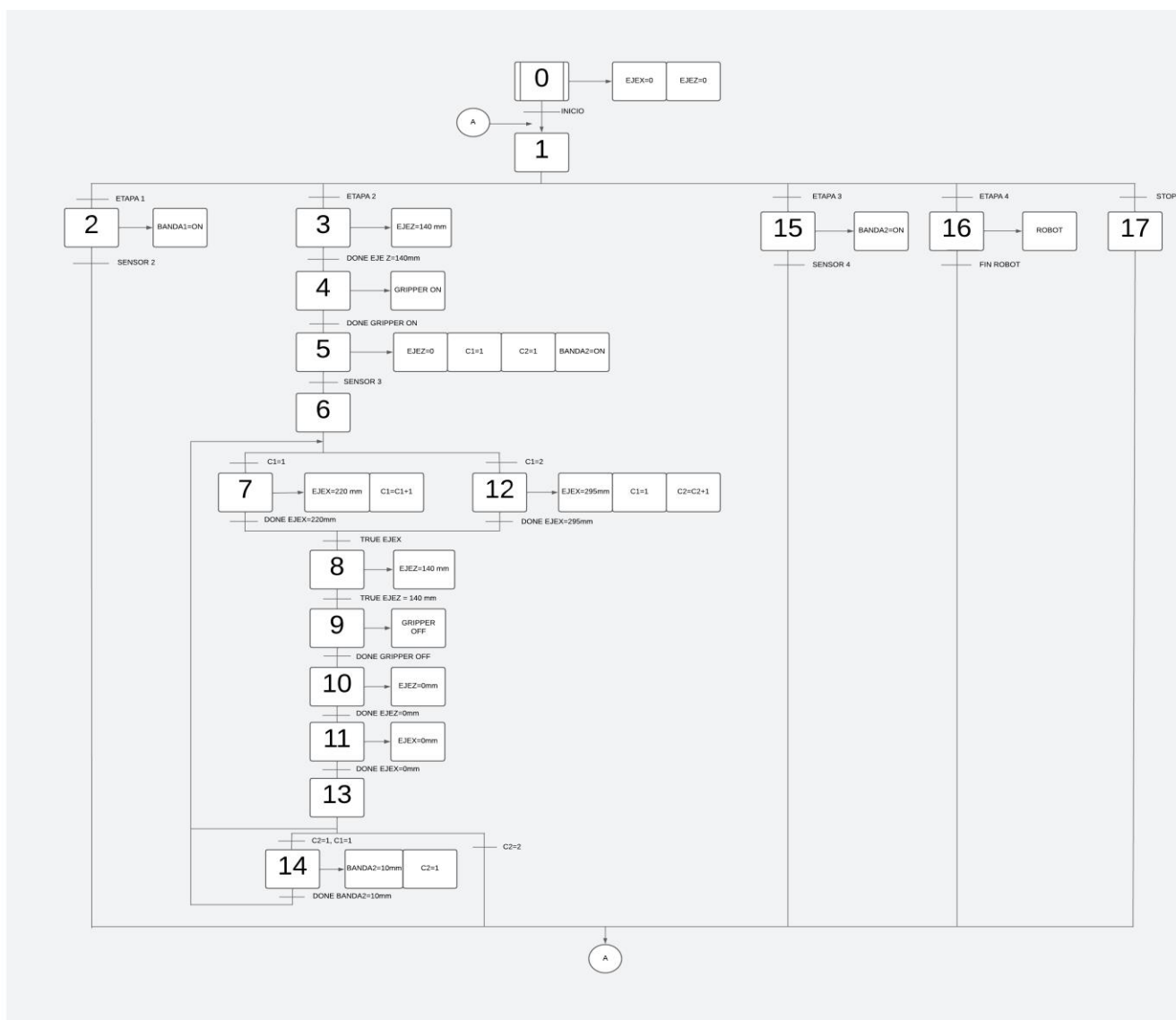
Diseño simulado de la planta



Se utiliza el diagrama Grafcet como una representación de la lógica secuencial del sistema, en la cual se incluyen las cuatro etapas del proceso desarrollado como se muestra en

la Figura 25, esta herramienta se emplea mucho en la industria de manufactura del Ecuador y todo el mundo. La primera etapa es el transporte de los envases por la Banda 1 hasta ser detectado por el sensor S2, esta etapa se activa cuando se presiona el botón Etapa 1 del tablero de control. La segunda etapa activa al robot cartesiano, desplazando el Eje z hasta llegar a una altura apropiada, cerrara el gripper y subirá hasta la posición Home, seguidamente desplazara el Eje X hasta posicionarse por encima de la Banda 2, y procederá a bajar a una posición programada y abrirá el gripper dejando caer el envase, toda esta secuencia se repetirá cuatro veces hasta llenar las 4 posiciones de la jaba, esta etapa iniciara cuando se presione el botón Etapa 2 en el tablero de control, y el sensor S2 detecte la existencia de producto, caso contrario esta etapa no podrá arrancar. La tercera etapa consiste en el transporte de la jaba que contiene los envases hasta el final de la Banda 2, exactamente hasta ser detectado por el sensor S4, esta etapa se activa cuando se presiona el botón Etapa 3 en el tablero de control. Finalmente, la etapa 4 es la activación del robot manipulador, el cual bajara hasta una altura adecuada, cerrara el gripper engancho un extremo de la jaba y la llevara hasta la rampa que conduce al almacén de jabas.

Figura 25
Diagrama Grafset

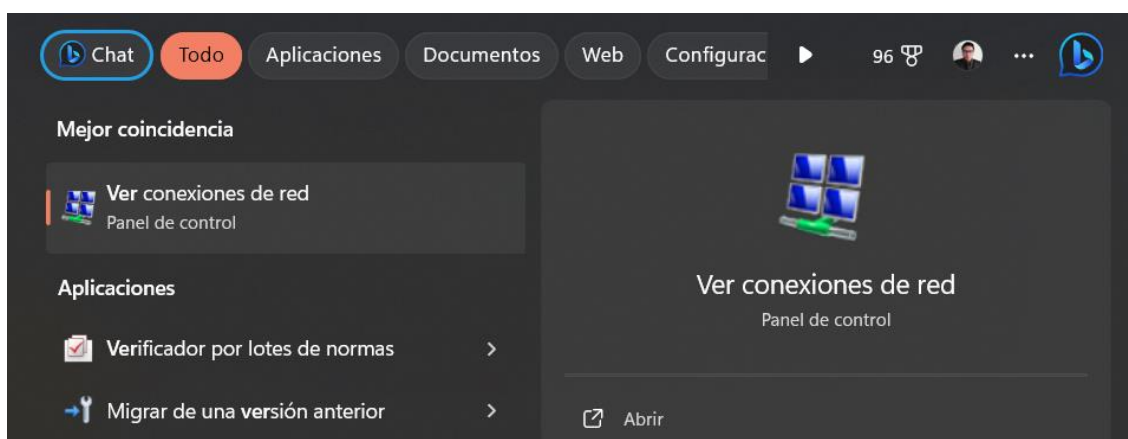


Configuración de propiedades Ethernet del ordenador

Para establecer una conexión entre el PLC y el ordenador es necesario que todos los dispositivos se encuentren en la misma subred, para ello se debe buscar la opción “Ver propiedades de red” tal como se muestra en la Figura 26.

Figura 26

Búsqueda de la opción "Ver conexiones de red"

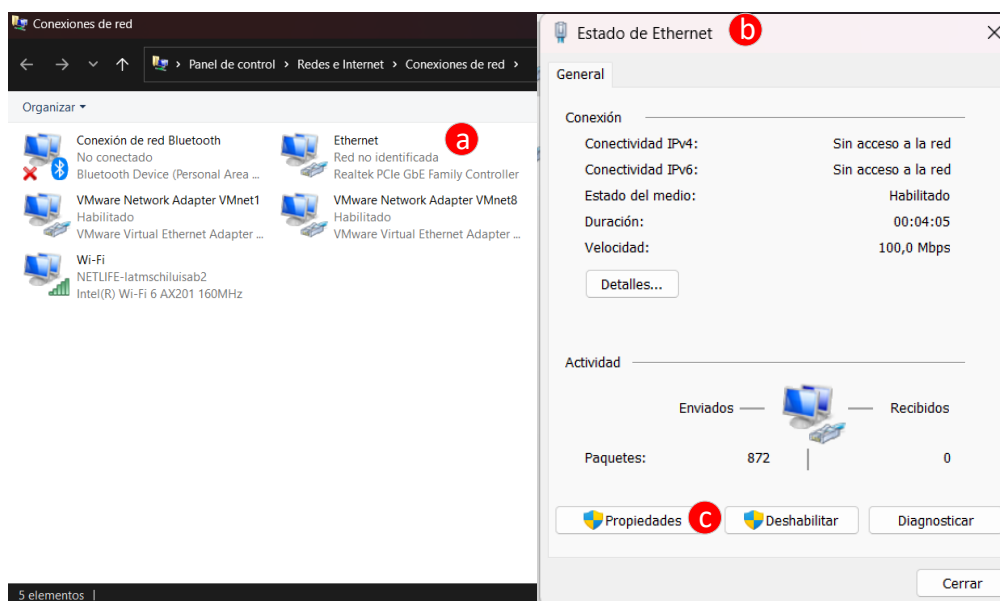


Una vez dentro de Conexiones de Red, se debe:

- a. Buscar la tarjeta de red Ethernet de la PC física, que en este caso es la Realtek PCIe GbE Family Controller
- b. Dar doble clic y desplegar la pestaña Estado Ethernet
- c. Clic en Propiedades, tal como se muestra en la Figura 27.

Figura 27

Selección de tarjeta de red Ethernet de la PC

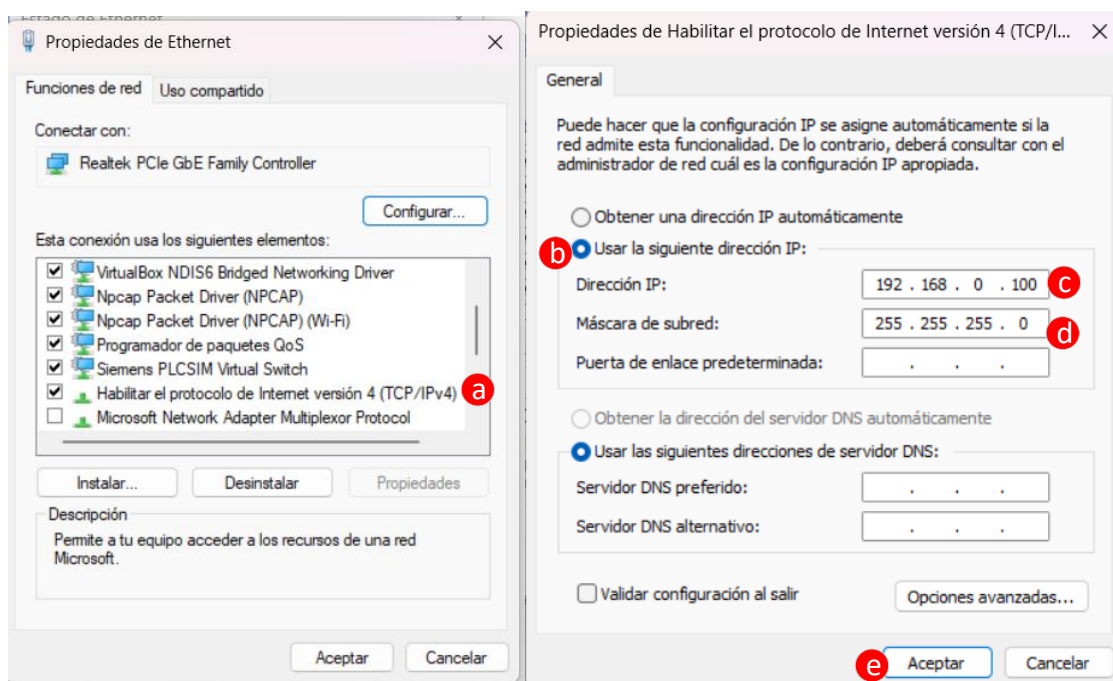


Por consiguiente, se despliega la venta “Propiedades de Ethernet”, en la cual se debe ingresar los valores que se muestran en la Figura 28.

- Dar doble clic en “Habilitar el protocolo de Internet version 4(TCP/IPv4)”
- Habilitar la opcion “Usar la siguiente direccion IP”
- En el campo “Direccion IP” se ingresa la IP 192.168.0.100, esta IP no debe ser igual a la del PLC.
- El apartado de “Mascara de subred” se genera por defecto con 255.255.255.0
- Clic en Aceptar.

Figura 28

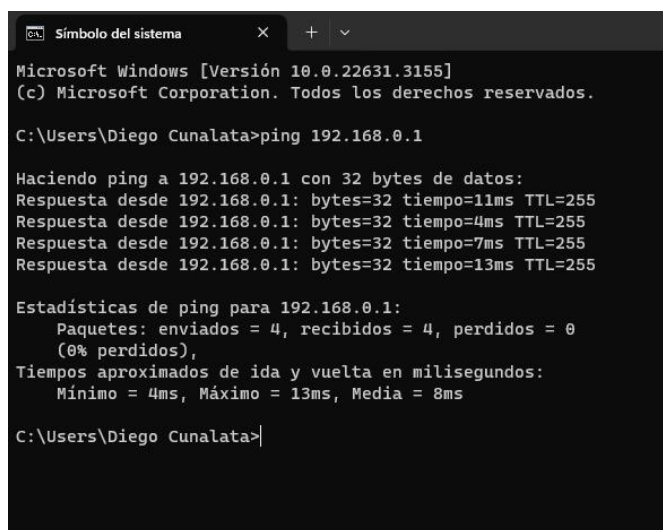
Ventana Propiedades de Ethernet



Finalmente, para comprobar que la asignación de las direcciones IP ha sido exitosa se recomienda realizar un ping entre el PLC y el ordenador, para ello se utiliza el símbolo del sistema tal como se muestra en la Figura 29.

Figura 29

Ping entre PLC y ordenador



```
Símbolo del sistema
Microsoft Windows [Versión 10.0.22631.3155]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\Diego Cunalata>ping 192.168.0.1

Haciendo ping a 192.168.0.1 con 32 bytes de datos:
Respuesta desde 192.168.0.1: bytes=32 tiempo=11ms TTL=255
Respuesta desde 192.168.0.1: bytes=32 tiempo=4ms TTL=255
Respuesta desde 192.168.0.1: bytes=32 tiempo=7ms TTL=255
Respuesta desde 192.168.0.1: bytes=32 tiempo=13ms TTL=255

Estadísticas de ping para 192.168.0.1:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
              (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 4ms, Máximo = 13ms, Media = 8ms

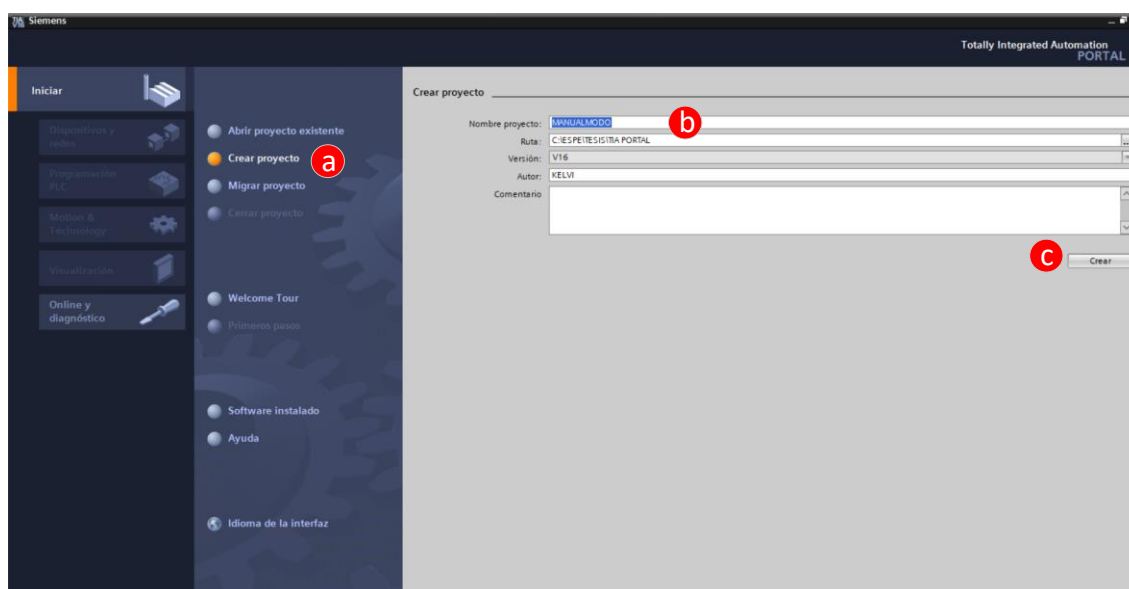
C:\Users\Diego Cunalata>
```

Creación de un nuevo proyecto en TIA PORTAL V16

Para la programación del PLC de la marca Siemens se va a usar el software TIA PORTAL V16, y a continuación se muestran los pasos necesarios para creación de un nuevo proyecto.

En primera instancia se ingresa al menú principal de TIA PORTAL, tal como se visualiza en la Figura 30, y los pasos a seguir son los siguientes:

- a. Clic en “Crear Proyecto”
- b. En el apartado de “Nombre” y “Ruta”, se llena de acuerdo a requerimientos del usuario.
- c. Clic en Crear.

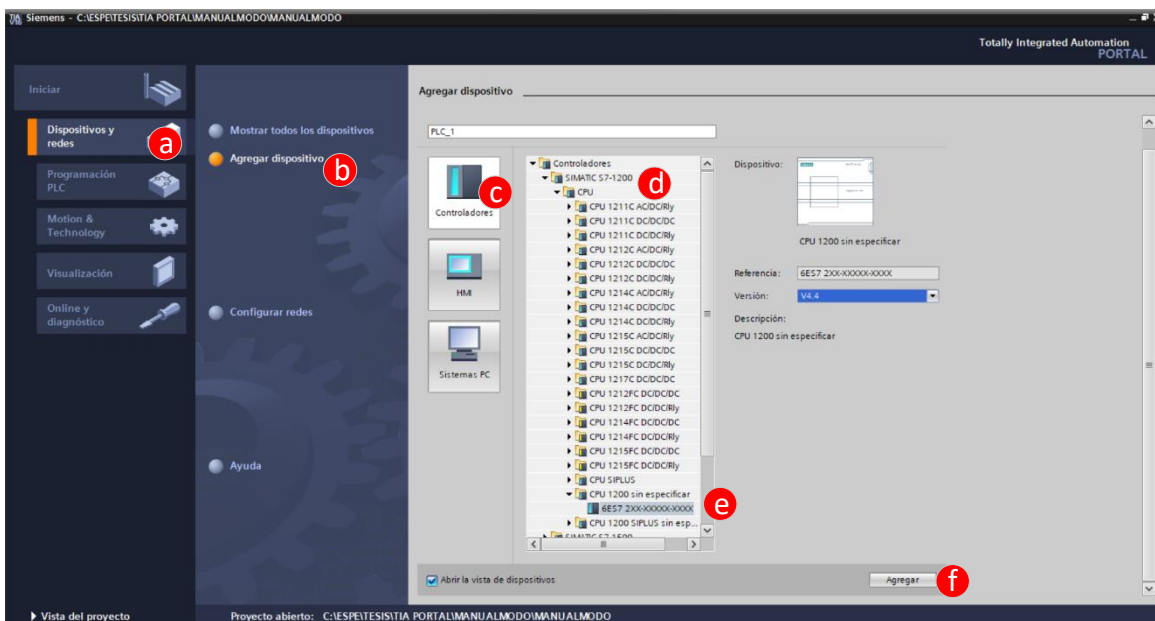
Figura 30*Creación de nuevo proyecto en TIA PORTAL*

En la Figura 31, se muestra la configuración del dispositivo con el que se va a trabajar, para ello, se deben seguir con los siguientes pasos:

- a. Clic en Dispositivos y Redes
- b. Clic en Agregar Dispositivo
- c. Seleccionar el apartado Controladores
- d. Desplegar las opciones de SIMATIC S7-1200 y escoger CPU
- e. Buscar la opción “CPU 1200 sin especificar”, esto ayuda a detectar con precisión el modelo de nuestro PLC
- f. Dar clic en Agregar.

Figura 31

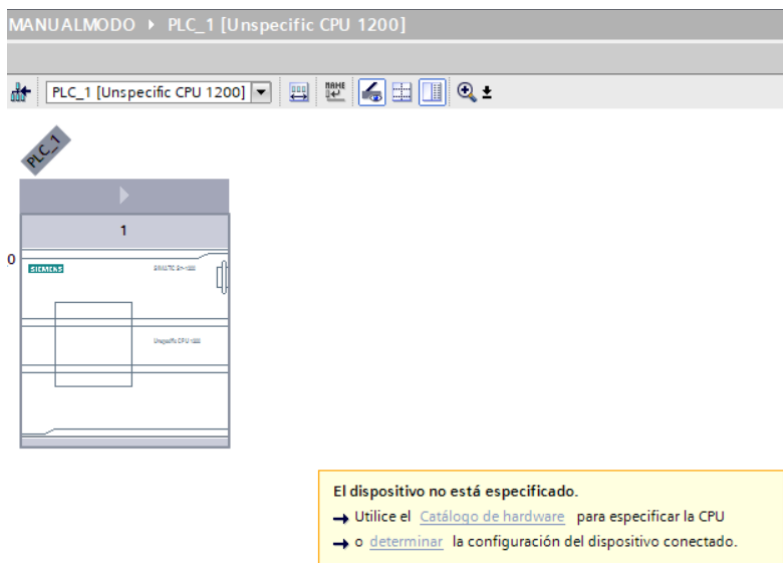
Configurar un dispositivo en TIA PORTAL



Una vez agregado el dispositivo, se abre la pantalla de programación del TIA PORTAL, en la cual se muestra un recuadro de color amarillo con un mensaje, en el cual se debe seleccionar la opción “determinar” tal como se muestra en la Figura 32.

Figura 32

Determinar la configuración del dispositivo conectado

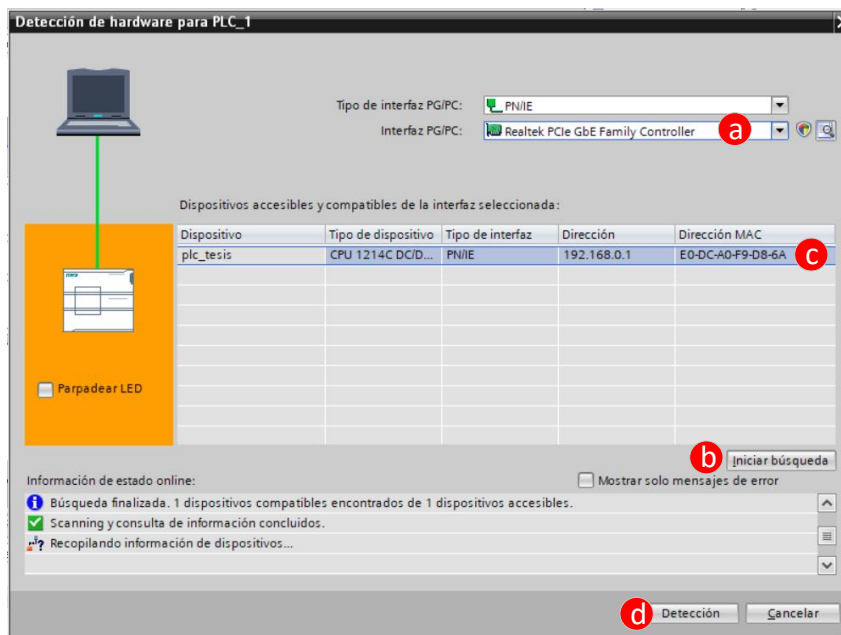


En consecuencia, al presionar “determinar” se despliega una ventana como se visualiza en Figura 33, en la cual se debe seguir los pasos que se detallan a continuación:

- a. En Interfaz PG/PC se debe escoger la tarjeta de red del ordenador.
- b. Dar clic en iniciar búsqueda
- c. Seleccionar el PLC con la IP que aparece en el recuadro de “Dispositivos accesibles y compatibles”
- d. Clic en detección.

Figura 33

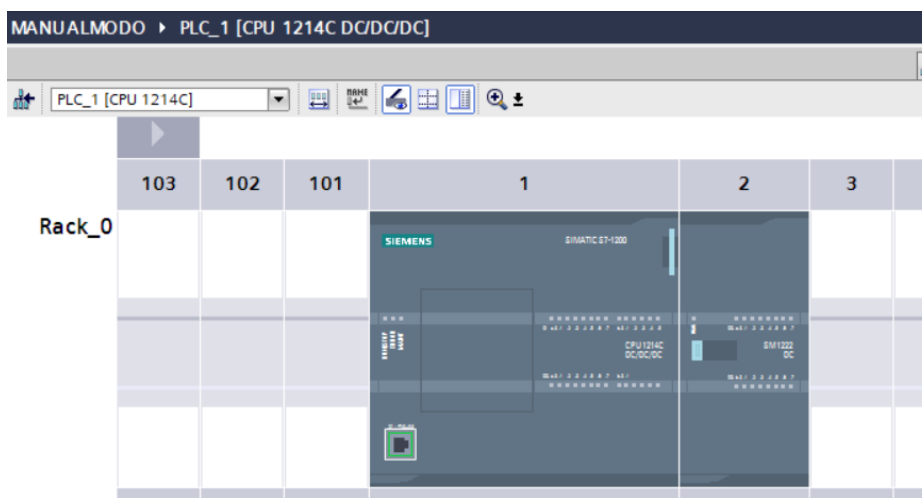
Detección de hardware para PLC



Finalmente se muestra que el dispositivo detectado es el PLC S7-1200 CPU 1214C DC/DC/DC y el módulo de expansión de salidas DIGITALES SM1222 DC tal como se observa en la Figura 34.

Figura 34

Dispositivo detectado PLC S7-1200 CPU 1214C DC/DC/DC



Creación y configuración de objetos tecnológicos

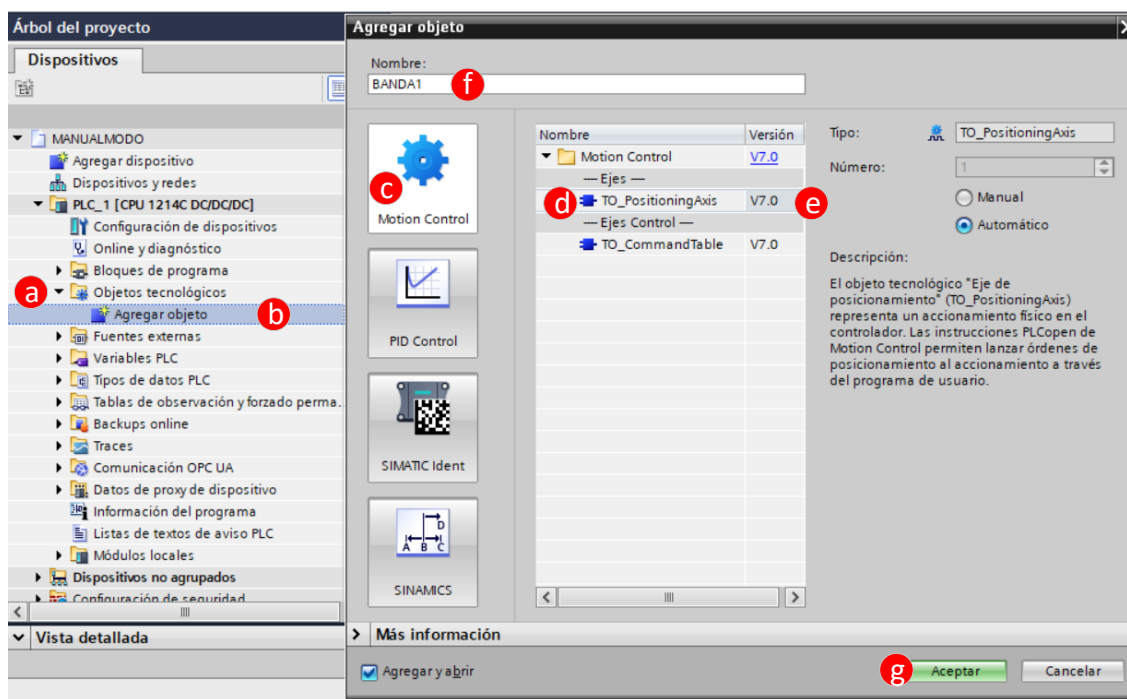
Para controlar el movimiento de los motores paso a paso es necesario la creación y configuración del objeto tecnológico Motion Control, se debe crear un objeto tecnológico para cada motor, los pasos necesarios se enumeran a continuación.

1. Agregar Objeto tecnológico.

- a. Dentro del Árbol de proyectos de TIA PORTAL, seleccionar la opción “Objetos Tecnológicos” como se muestra en Figura 35.
- b. Clic en “Agregar Objeto”
- c. Seleccionar “Motion Control”
- d. Escoger la opción “TO_PositioningAxis”
- e. En cuanto a la versión escoger la V7.0
- f. Escribir un nombre.
- g. Clic en “Agregar”

Figura 35

Agregar objeto tecnológico Motion Control



2. Configuración de Parámetros Básicos

Dentro de este apartado se debe configurar la pestaña General tal como se visualiza en la Figura 36, y la pestaña Accionamiento como se presenta en Figura 37.

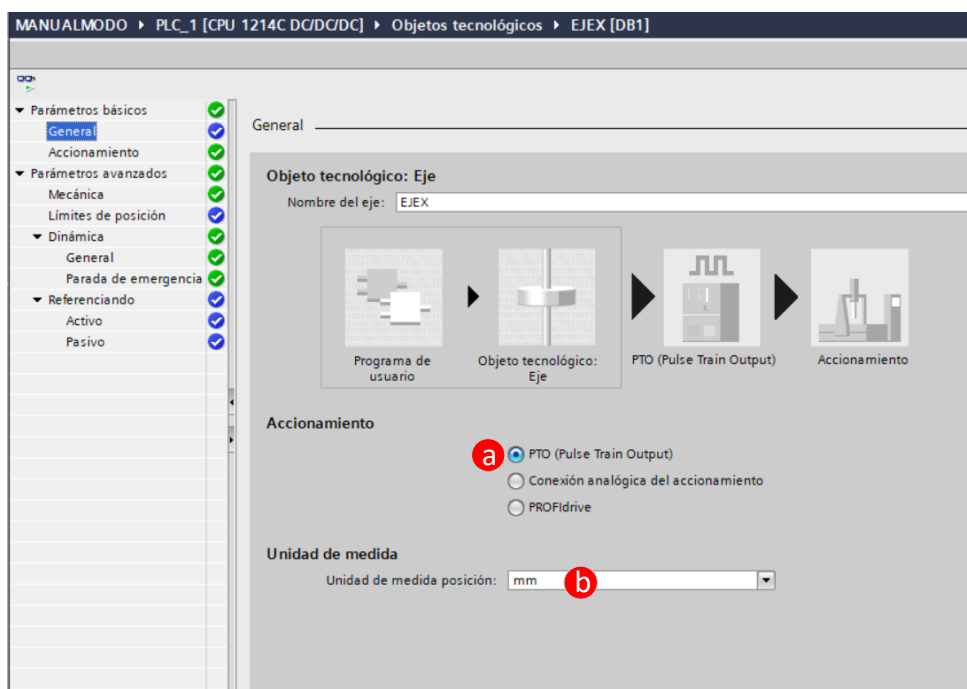
2.1. General

a. Seleccionar PTO (Pulse Train Output)

b. Unidad de medida posición, se ha escogido milímetros.

Figura 36

Configuración de Parámetros Básicos - General



2.2. Accionamiento

La configuración en este apartado ira cambiando de acuerdo al motor que se vaya a configurar, es por ello que en la Tabla 4, se presentan los valores que corresponden a cada motor.

Tabla 4

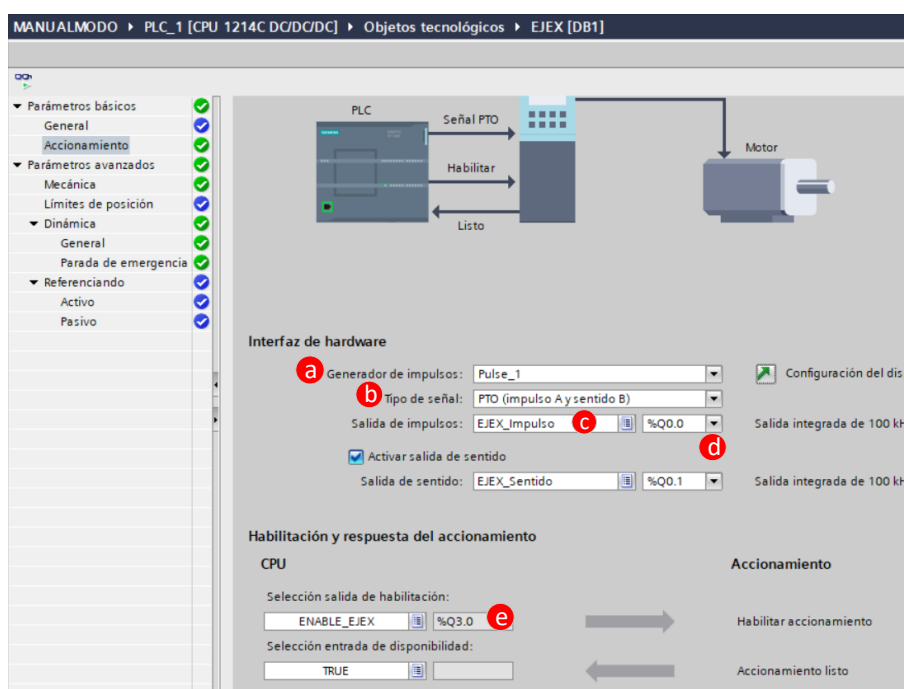
Valores para configuración de Parámetros Básicos sección Accionamiento

Motor no.	Descripción	Generador de impulsos	Salida de impulsos	Salida de sentido	Selección salida de habilitación
1	EJE X	Pulse_1	Q0.0	Q0.1	Q3.0
2	EJE Z	Pulse_2	Q0.2	Q0.3	Q3.1
3	BANDA 1	Pulse_3	Q0.4	Q0.5	Q3.2
4	BANDA 2	Pulse_4	Q0.6	Q0.7	Q3.3

- a. En el apartado “Generador de impulsos” seleccionar el pulso correspondiente a cada motor, según la Tabla 4, es importante mencionar que el hardware nos permite utilizar un máximo de hasta cuatro pulsos, es por ello que, si se desea agregar un motor nuevo, es necesario la adquisición de un nuevo hardware.
- b. En “Tipo de señal” seleccionar la opción PTO (impulso A y sentido B)
- c. En “Salida de impulsos” el nombre se generará automáticamente, de acuerdo con el nombre del objeto tecnológico que se esté creando, en cuanto a las direcciones de salidas digitales del PLC se debe ir cambiando según Tabla 4.
- d. En “Salida de sentido” el nombre se genera automáticamente como se menciona en el literal b, y la dirección de salida del PLC se debe ir cambiando según la Tabla 4.
- e. Finalmente, en el apartado “Sección salida de habilitación”, se escogen las salidas digitales del módulo, tal como indica la Tabla 4. Esta configuración en especial sirve para habilitar el accionamiento y respuesta del motor. Para mayor comprensión se recomienda visualizar Figura 21.

Figura 37

Configuración de Parámetros Básicos – Accionamiento



3. Parámetros avanzados

Los parámetros que se deben configurar van variando de acuerdo al motor, su utilidad, su ubicación física y el tipo de carga que va a mover, es por ello que en la Tabla 5, se describen los valores que han sido asignados para cada motor.

Tabla 5

Valores para la configuración de parámetros avanzados sección Mecánica

Motor no.	Descripción	Impulsos por vuelta de motor	Movimiento de la carga por vuelta del motor	Sentido de giro permitido
1	EJE X	1600	40 mm	Ambos sentidos
2	EJE Z	1600	8 mm	Ambos sentidos
3	BANDA 1	1600	180 mm	Ambos sentidos
4	BANDA 2	1600	150 mm	Ambos sentidos

Para el cálculo de impulsos por vuelta de motor se debe considerar el valor de ángulo de paso que nos entrega el fabricante del motor Nema 17.

$$\text{Pasos por revolución} = \frac{360^\circ}{\text{ángulo de paso}} \quad \text{EC.1}$$

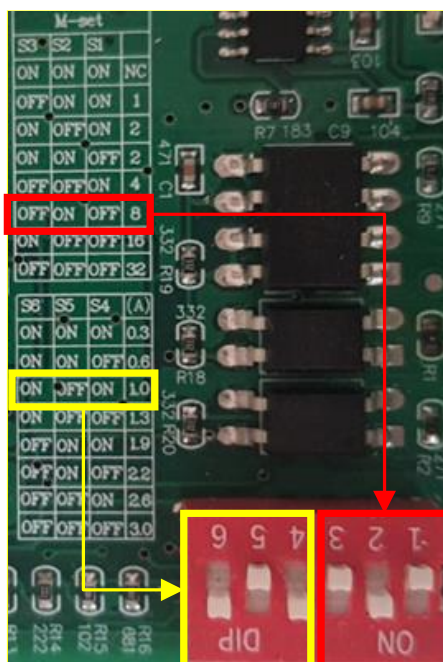
$$\text{Pasos por revolución} = \frac{360^\circ}{1.8^\circ} \quad \text{EC.2}$$

$$\text{Pasos por revolución} = \mathbf{200} \quad \text{EC.3}$$

Otro dato necesario es la cantidad de micro pasos que da el motor, para ello es necesario mirar la configuración del Driver TB6560, ya que dicho Driver permite la opción de dar un paso completo, 1/2 paso, 1/4 de paso, 1/8 de paso, 1/16 de paso y un máximo de 32 micro pasos. Todos los motores utilizados en la planta han sido configurados en 8 micro pasos tal como se muestra en la Figura 38.

Figura 38

Configuración de micro pasos y corriente máxima del motor Nema 17



Ahora que ya se disponen de todos los datos necesarios, es momento de calcular los impulsos por vuelta del motor, para ello se realiza el siguiente cálculo:

$$\text{Impulsos por vuelta del motor} = (\text{Pasos Por Revolucion}) * (\text{MicroPasos}) \quad \text{Ec.4}$$

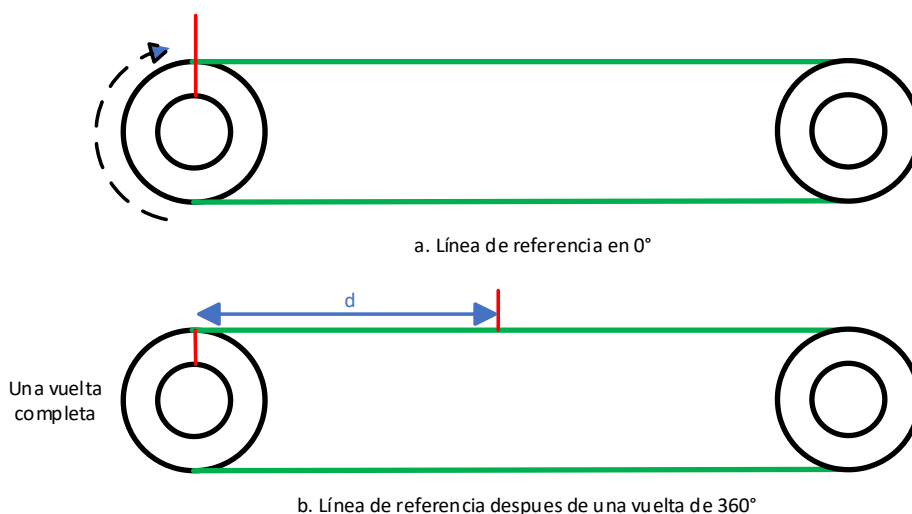
$$\text{Impulsos por vuelta del motor} = (200) * (8) \quad \text{Ec.5}$$

$$\text{Impulsos por vuelta del motor} = \mathbf{1600} \quad \text{Ec.6}$$

En cuanto al valor que se asigna en el apartado *Movimiento de la carga por vuelta del motor* va a depender de las características físicas de la estructura en la cual va a funcionar el motor y la carga que soportara dicho elemento. Para obtener el valor de cada motor, es necesario marcar un punto de referencia en su eje, y hacerlo girar 360° para luego medir la distancia “d” que ha logrado desplazar a su estructura física, esto se lo puede ver de mejor manera en la Figura 39, en la que se ha tomado como ejemplo a una banda transportadora.

Figura 39

Cálculo del movimiento de la carga por vuelta del motor



Una vez explicado los valores de la Tabla 5, se procede a llenar los datos en la pestaña Mecánica para los cuatro motores, tal como se visualiza en la Figura 40.

- Ingresar el valor de 1600 en el apartado “Impulso por vuelta del motor”, este valor es constante para todos los motores.
- En el “Aparatado de la carga por vuelta del motor” ingresar el valor que se indica en la Tabla 5, de acuerdo al motor que se esté configurando.
- Escoger la opción Ambos sentidos.

Figura 40

Configuración de parámetros avanzados



4. Configuración de la Dinámica

4.1. General

Dentro de la pestaña Dinámica se debe configurar los siguientes aspectos:

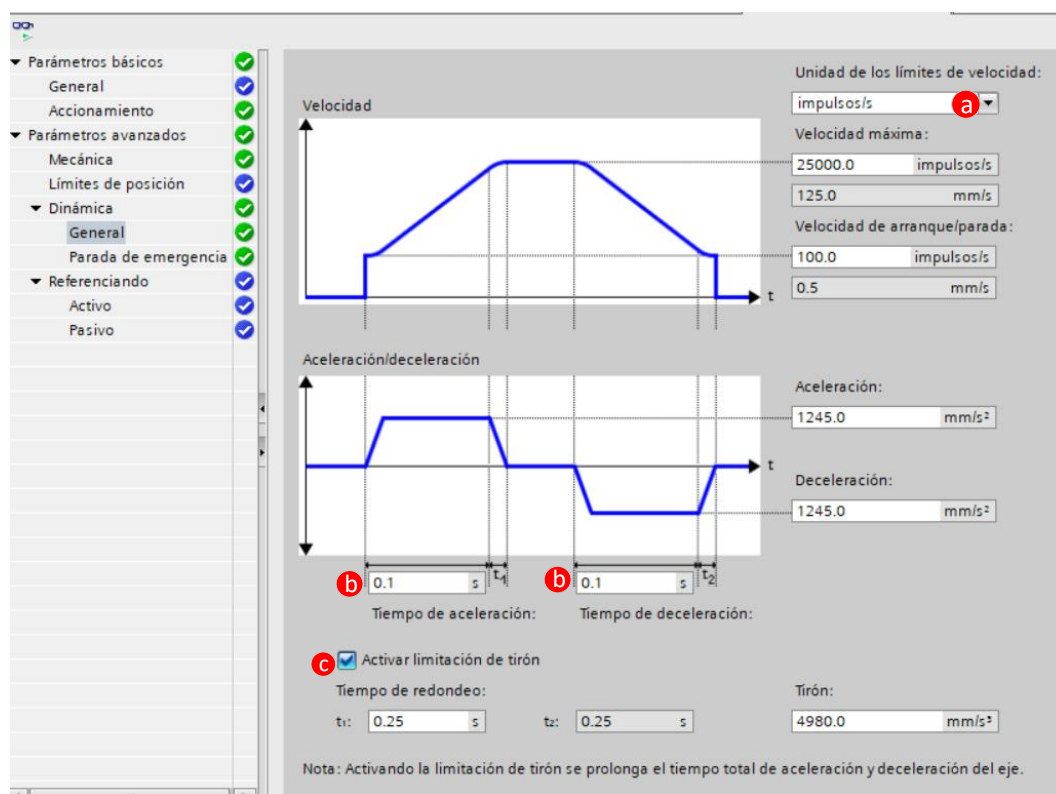
- Escoger la opción impulsos/s dentro del apartado “Unidad de límites de velocidad”.
- Ingresar el valor de 0.1 en “Tiempo de aceleración” y “Tiempo de deceleración”, este valor es porque se le condiciona mediante software que estos dos parámetros ocurran en un tiempo de 0.1 [s].

- c. Activar límite de tirón, esta opción es muy importante ya que prolonga el tiempo total de aceleración y deceleración del eje, dando como resultado un arranque más suave de los motores.

Los otros parámetros que se observan en la Figura 41, aparecen automáticamente con las configuraciones realizadas en los pasos anteriores.

Figura 41

Configuración de la Dinámica del objeto tecnológico

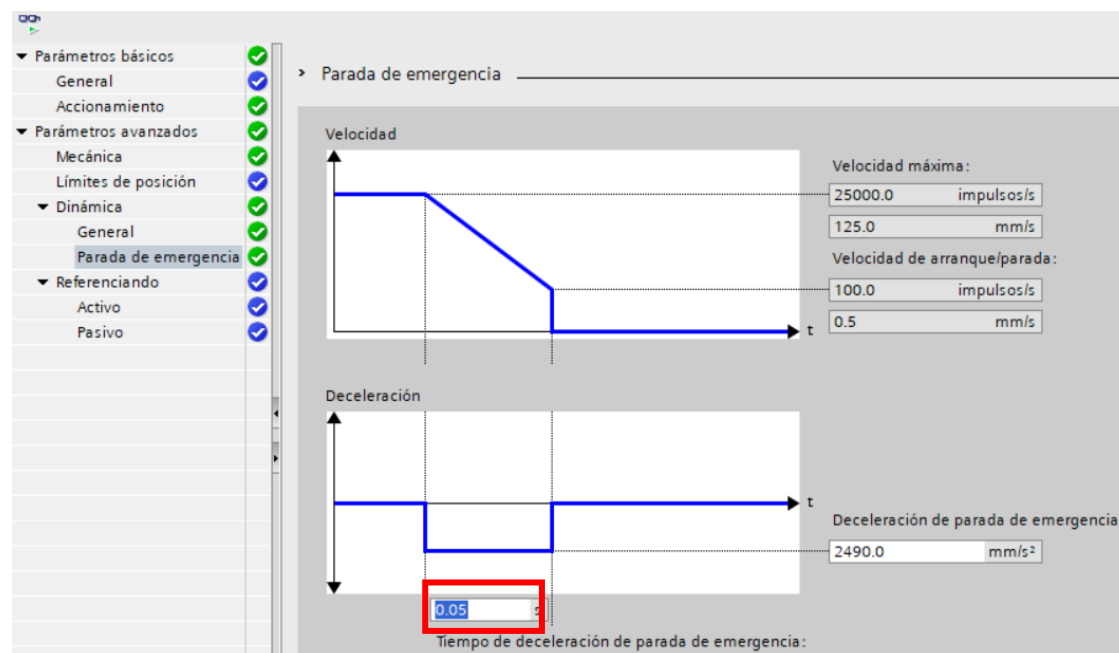


4.2. Parada de emergencia

Al tratarse de un paro de emergencia, lo que debe condicionar es que suceda de inmediato o en el menor tiempo posible, por tal motivo la deceleración debe ocurrir en un lapso de 0.05 [s], tal como se visualiza en la Figura 42.

Figura 42

Configuración de para de emergencia del objeto tecnológico



Al finalizar la configuración de los cuatro objetos tecnológicos, de los cuatro motores, siguiendo los pasos mostrados anteriormente se puede visualizar en el Árbol de proyectos los nombres de dichos objetos, tal como se visualiza en la Figura 43.

Figura 43

Visualización de objetos tecnológicos creados



Programación del funcionamiento en modo manual en el bloque Main

El bloque de programación Main, se ha dividido en trece segmentos, los cuales son:

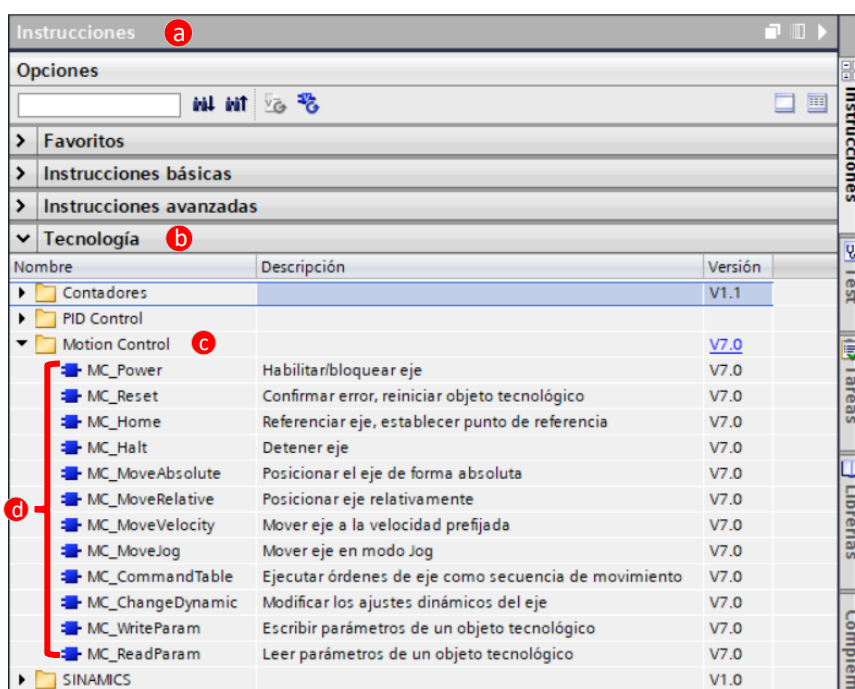
1. Energizado del sistema.
2. Encendido de motores 1, 2, 3 y 4.
3. Posicionamiento Home de los motores 1, 2, 3 y 4.
4. Funcionamiento JOG de motores 1, 2, 3 y 4.
5. Modo Relative.
6. Modo Absolute.
7. Paro de emergencia de motores 1, 2, 3 y 4.
8. Etapa 1.
9. Etapa 2.
10. Etapa 3.
11. Etapa 4.
12. Retorno a posiciones de referencia de los motores 1, 2, 3 y 4.
13. Comunicación Modbus PLC-Arduino.

Antes de iniciar con la programación es necesario la explicación de los bloques que se va a utilizar, los cuales están dentro de la herramienta *Motion Control*. Para acceder a dicha herramienta es necesario seguir los siguientes pasos:

- a. Buscar la pestaña “Instrucciones”, por defecto se encuentra en el lado derecho de la pantalla de TIA PORTAL, como se visualiza en Figura 44.
- b. Clic en “Tecnología”
- c. Desplegar la carpeta “Motion Control”
- d. Existen 12 bloques asociados a la herramienta Motion Control, su descripción breve se presenta en Figura 44.

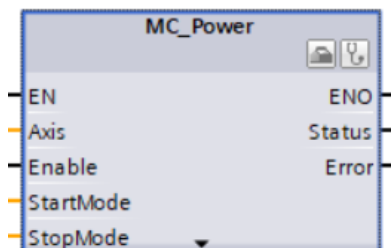
Figura 44

Ubicación de la herramienta Motion Control

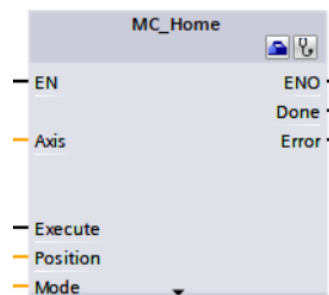


Los bloques que se ha utilizado para la automatización son: MC_Power, MC_Home, MC_Halt, MC_MoveAbsolute y MC_MoveRelative. Para llevar estos bloques al Main, basta con arrastrarlos hasta el segmento en el que se los vaya a utilizar.

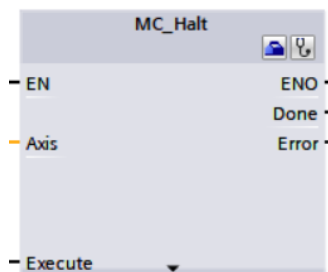
MC_Power: su función es habilitar o energizar el eje, y se caracteriza porque no puede ser cancelado por ninguna orden de Motion Control. Su aspecto se lo visualiza en la Figura 45, para conocer el tipo de dato que ingresa en cada PIN de cualquier elemento de TIA PORTAL basta con señalarlo y presionar el botón **F1** del teclado.

Figura 45*Bloque MC_POWER*

MC_Home: su función es la referenciación del eje, es decir que este bloque almacena la posición actual del eje como el punto cero. Su aspecto se lo puede visualizar en Figura 46.

Figura 46*Bloque MC_Home*

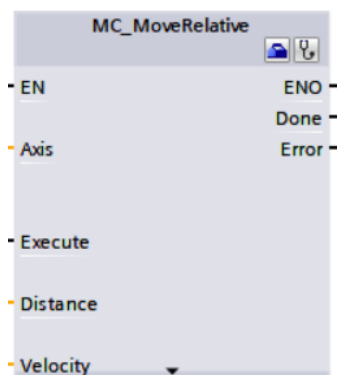
MC_Halt: la función principal de este bloque es detener al eje en cualquier circunstancia, con frecuencia es usado para un paro de emergencia, su aspecto se lo puede visualizar en la Figura 47.

Figura 47*Bloque MC_Halt*

MC_MoveAbsolute: la función de este bloque es posicionar al eje de forma relativa, es decir que su punto de referencia inicial se mueve a medida que el eje se desplace, su aspecto se lo puede visualizar en la Figura 48.

Figura 48

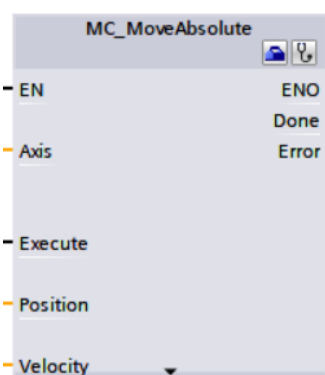
Bloque MC_MoveRelative



MC_MoveRelative: la función de este bloque es posicionar al eje de forma absoluta, es decir que su punto de referencia inicial permanece constante, su aspecto se lo puede visualizar en la Figura 49.

Figura 49

Bloque MC_MoveAbsolute



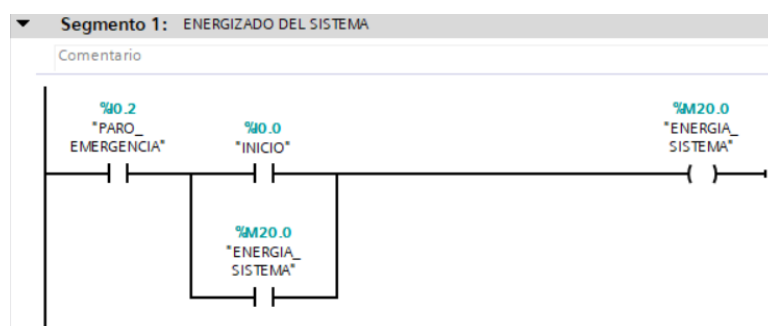
A continuación, se presenta la descripción detallada de cada segmento antes mencionado.

Energizado del sistema.

En este segmento se enclava la memoria ENERGIA_SISTEMA, la que se encargara de la activación de los 12 segmentos restantes y es accionada por el botón INICIO, podrá ser interrumpida únicamente por el botón PARO_EMERGENCIA, tal como se visualiza en la Figura 50.

Figura 50

Segmento Energizado del sistema

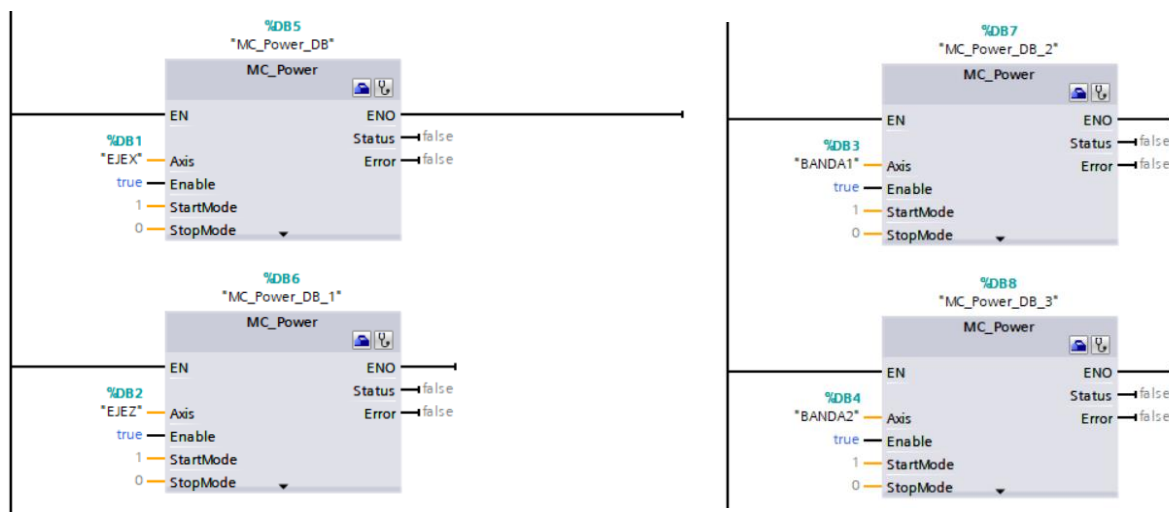


Encendido de motores 1, 2, 3 y 4.

Se utiliza los bloques MC_Power para encender los motores, se configuran los axis EJEX, EJEZ, BANDA1 y BANDA2 como se observa en la Figura 51.

Figura 51

Segmento Encendido de motores 1, 2, 3 y 4

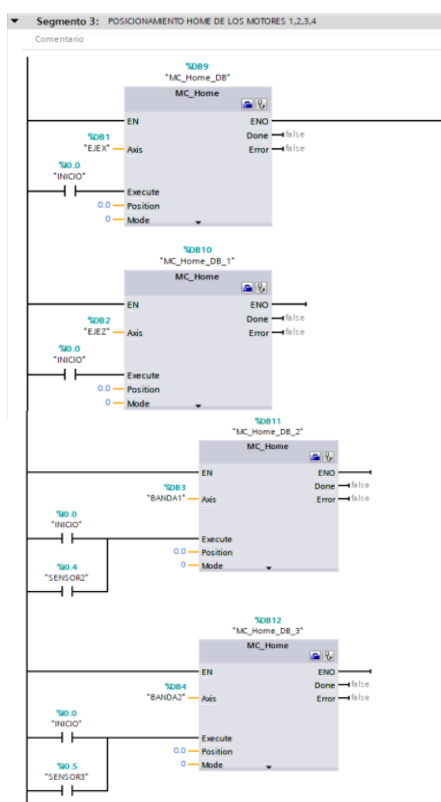


Posicionamiento Home de los motores 1, 2, 3 y 4.

Se emplea el bloque MC_Home para referenciar los cuatro ejes, el botón INICIO es el que activa este bloque para indicar que la posición cero va a ser la actual, y desde ahí se realizaran los diferentes movimientos, esto se puede observar en la Figura 52.

Figura 52

Segmento de Posicionamiento Home de los motores 1, 2, 3 y 4

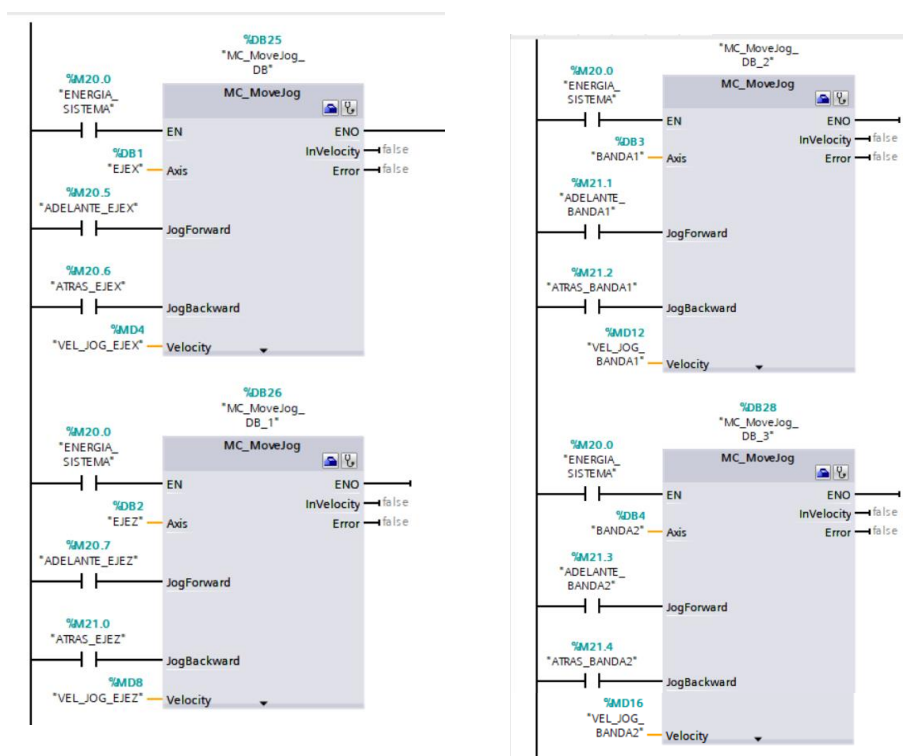


Funcionamiento JOG de motores 1, 2, 3 y 4.

El funcionamiento en modo JOG se encarga de mover los diferentes ejes de una forma constante siempre y cuando se tenga activado el contacto que ejecuta el bloque, se a empleado este modo de operación para realizar test de los ejes y posicionar los mismos en caso de ser necesario. La implementación JOG se la puede visualizar en la Figura 53.

Figura 53

Segmento de Funcionamiento JOG de motores 1, 2, 3 y 4

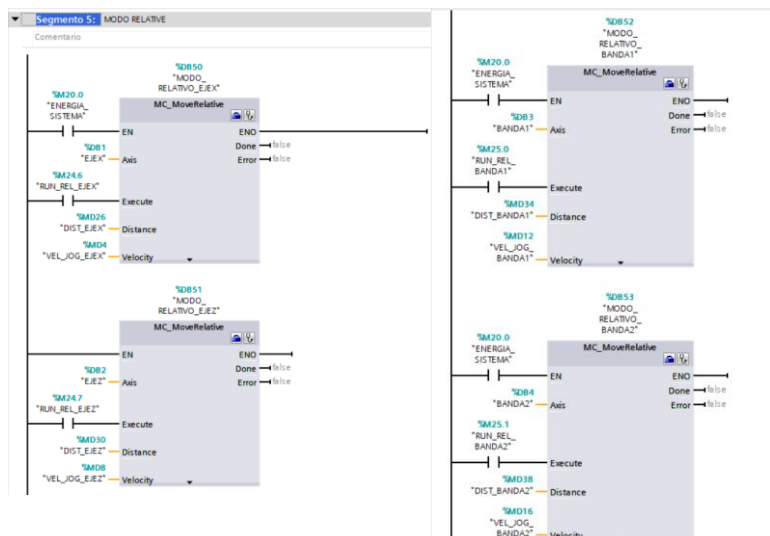


Modo Relative.

Dentro del programa se considerado pertinente la implementación del funcionamiento en modo relativo, con la finalidad de realizar pruebas independientes para cada eje de forma relativa, es por ello que en la Figura 54 se muestra la configuración realizada para este modo de operación.

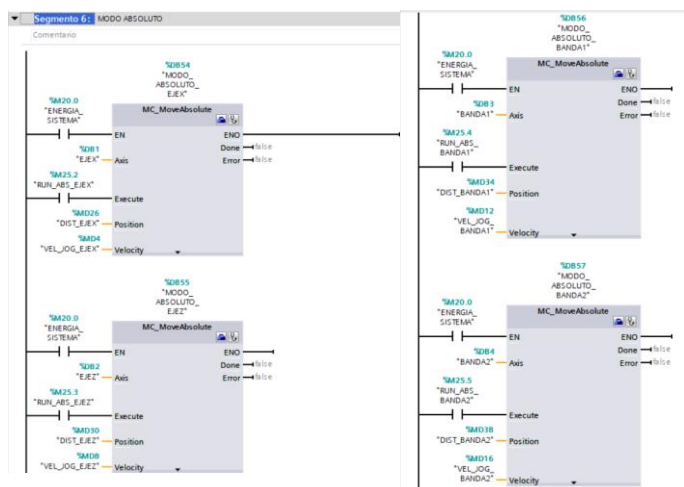
Figura 54

Segmento de Modo Relative

**Modo Absolute.**

El modo absoluto va como complemento del Modo relativo, se ha decidido la implementación de este modo para realizar pruebas independientes de cada eje en modo absoluto, a la vez permite diferenciar el contraste que existe con el modo relativo. Su configuración se presenta en la Figura 55.

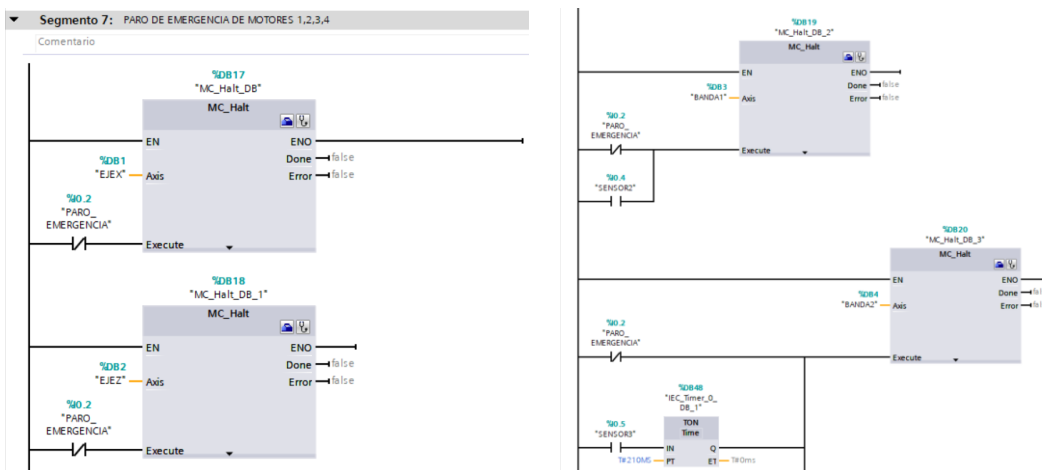
Figura 55
Segmento Modo absoluto



Paro de emergencia de motores 1, 2, 3 y 4.

Se ha implementado bloques MC_Halt para que paren los motores en cualquier instante como medida de precaución, ante la presencia de algún peligro hacia el operador o algún fallo que afecte la planta, dicha acción se lo realiza mediante el botón PARO_EMERGENCIA que se encuentra en la botonera del tablero de control, la implementación de este importante bloque se lo visualiza en la Figura 56.

Figura 56
Segmento Paro de emergencia de motores

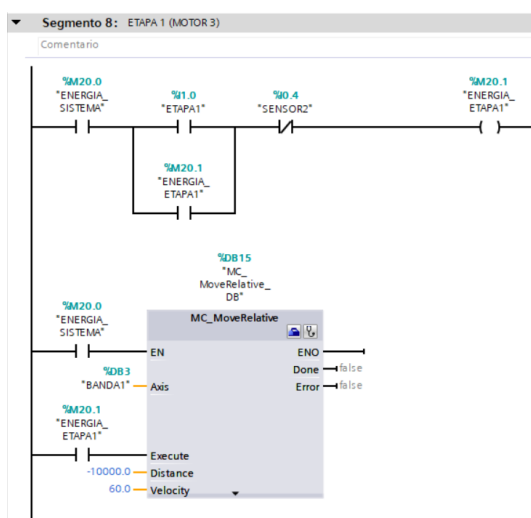


Etapa 1.

La etapa 1 sigue la lógica de programación del diagrama Grafcet de la Figura 25 se ha usado el bloque MC_MoveRelative para mover el eje BANDA1, cada vez que un envase sea detectado por el SENSOR 2 interrumpirá el movimiento de la banda. Esto se lo puede ver en la Figura 57.

Figura 57

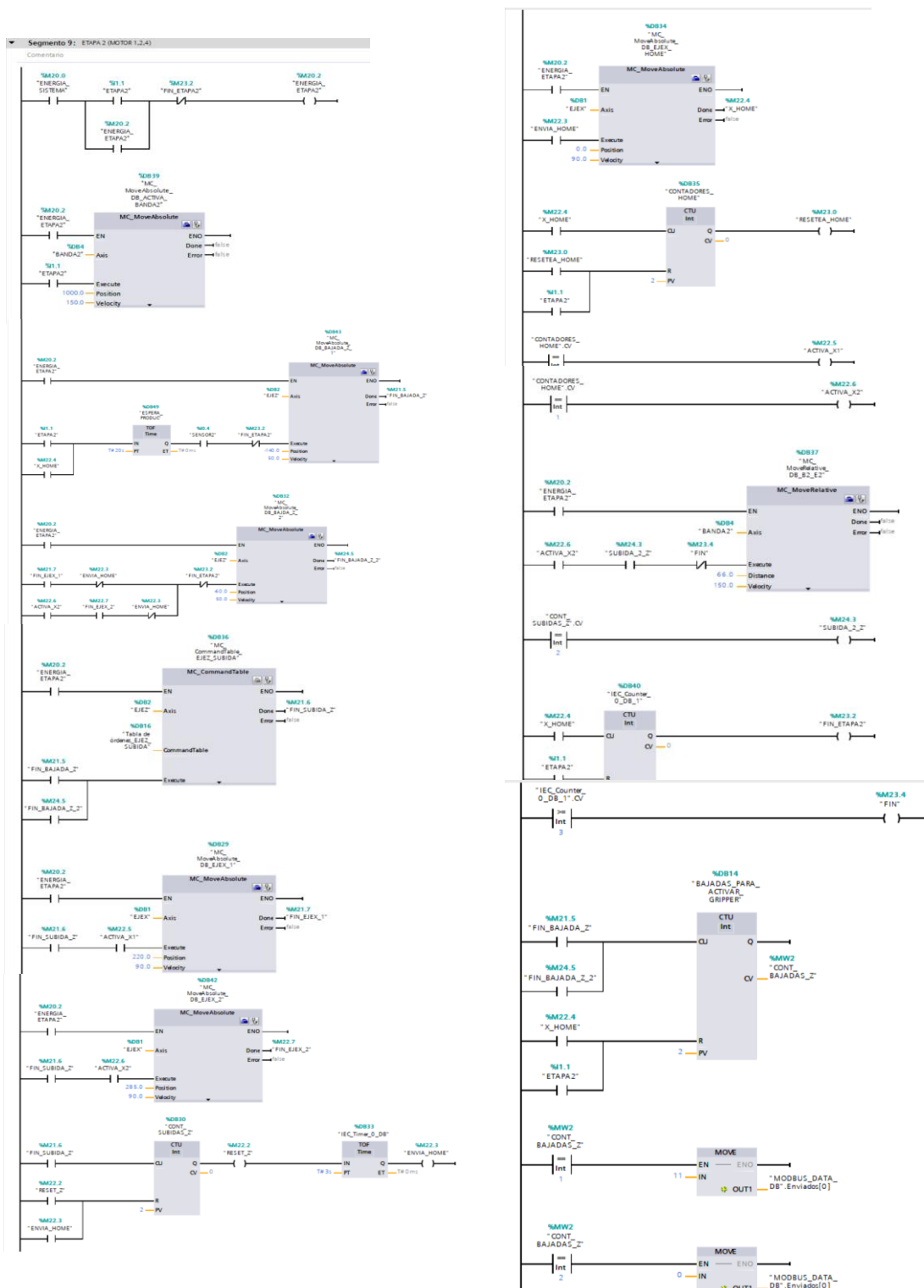
Segmento Etapa 1



Etapa 2.

La segunda etapa consiste en el traslado de los envases desde la Banda 1 hacia la jaba ubicada en la Banda 2, mediante el uso del Robot Cartesiano, su lógica se basa en el diagrama Grafcet que se muestra en la Figura 25. En esta etapa intervienen los bloques MC_MoveRelative, MC_MoveAbsolute, contadores y temporizadores, tal como se puede ver en la Figura 58.

Figura 58
Segmento Etapa 2

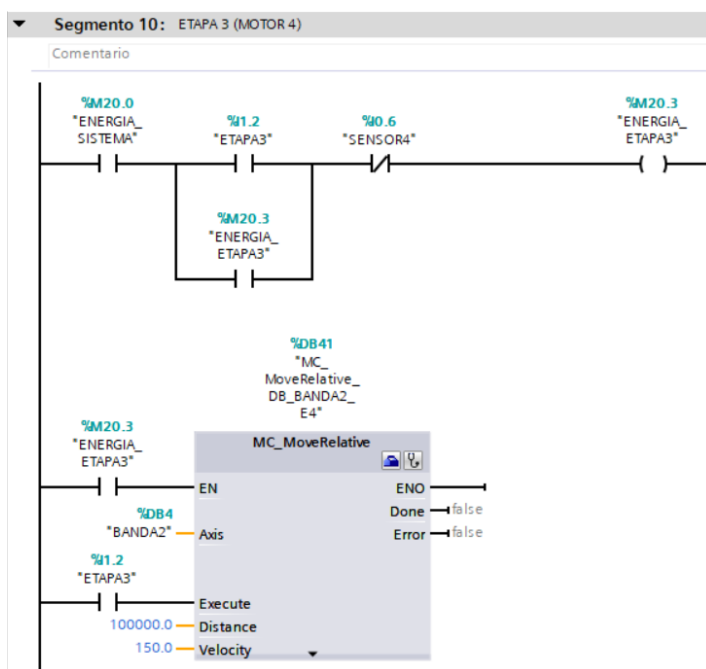


Etapa 3.

La tercera etapa consiste en el traslado de la jaba llena de envases desde su posición de carga hasta ser detectado por el SENSOR 4, y su implementación se lo puede visualizar en la Figura 59.

Figura 59

Segmento Etapa 3

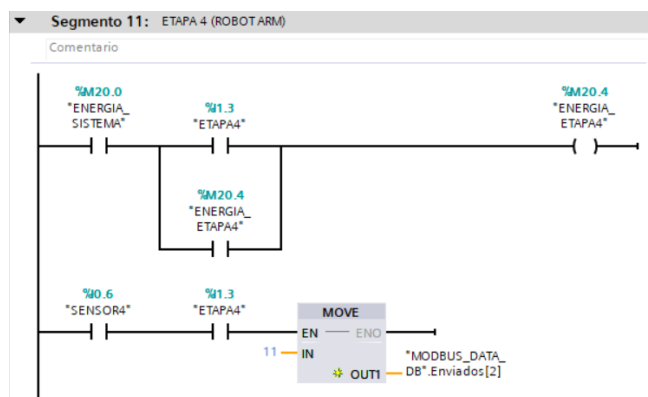


Etapa 4.

Finalmente, la etapa cuatro consiste en la activación del robot manipulador, el cual se encarga de llevar la jaba llena de envases hasta la rampa de almacenamiento. En esta etapa interviene la comunicación MODBUS TCP/IP entre el PLC y la tarjeta Arduino. Su implementación se lo puede visualizar en la Figura 60.

Figura 60

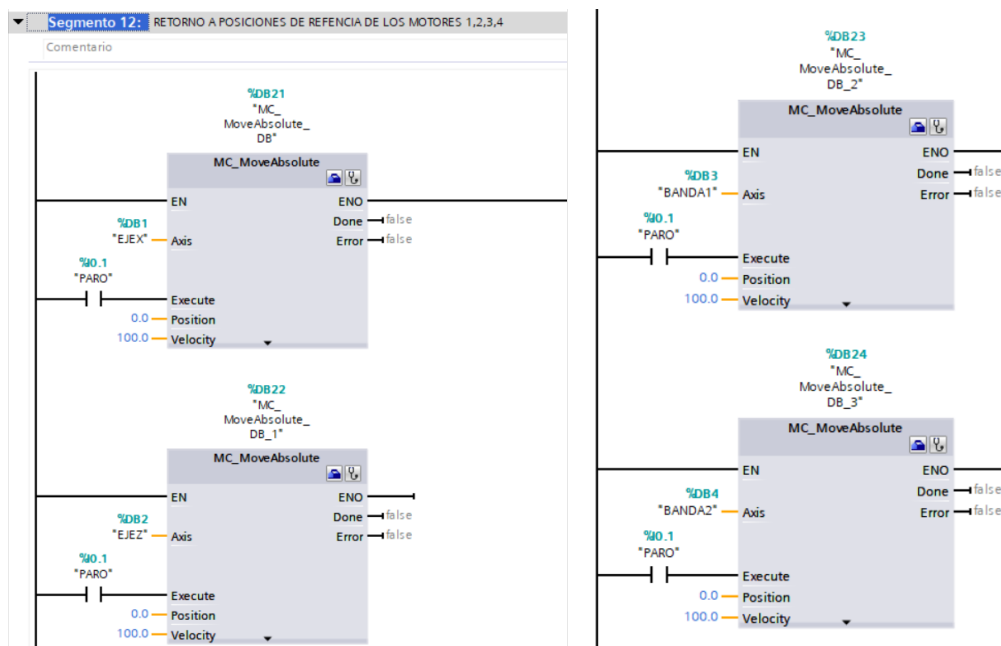
Segmento Etapa 4

**Retorno a posiciones de referencia de los motores 1, 2, 3 y 4.**

Se ha visto necesario la implementación de un segmento dedicado al retorno de los ejes a su punto de origen o posición home como se muestra en Figura 61, para aquellos casos en los que se vea estrictamente necesario interrumpir el proceso y llevar los ejes al origen. Esta instrucción se la realiza mediante el botón PARO que se ubica en el tablero de control.

Figura 61

Segmento Retorno a posición home



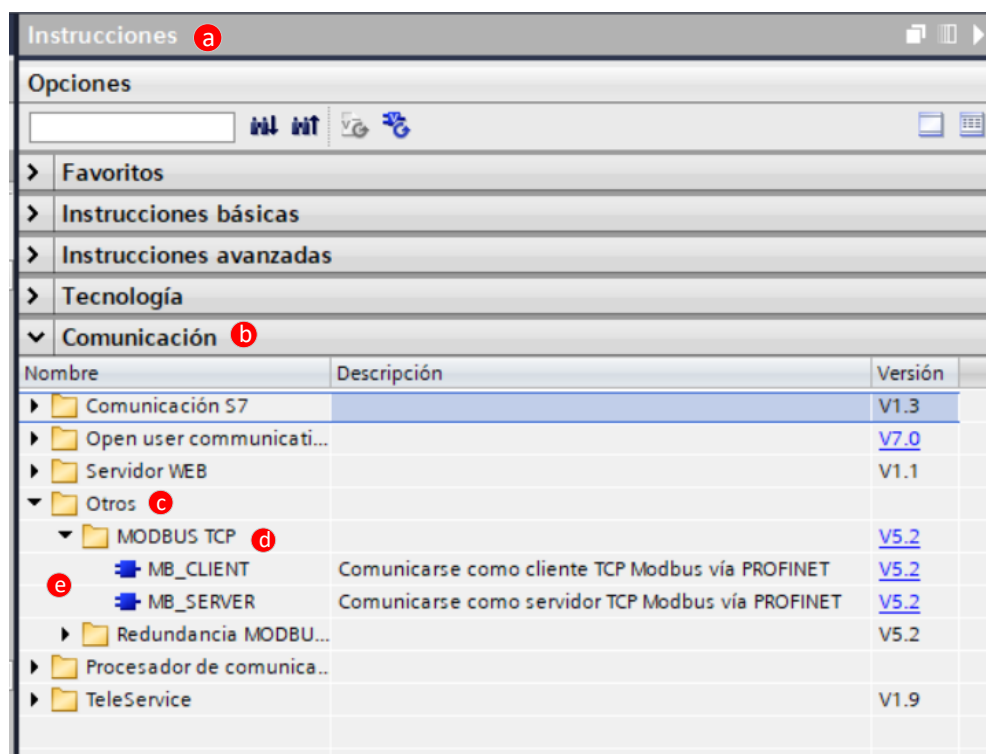
Comunicación Modbus PLC-Arduino.

Un aspecto importante de esta automatización es la comunicación MODBUS TCP/IP que existe entre el PLC S7-1200 y la placa Arduino, es por ello que se ha dedicado un segmento específicamente para el desarrollo de dicha comunicación. Los pasos que se requieren para esta configuración son los siguientes:

1. *Abrir la carpeta MODBUS TCP*
 - a. Para ello hay que dirigirse al derecho de la pantalla principal y encontrar la pestaña “Instrucciones” como se indica en Figura 62.
 - b. Clic en “Comunicación”
 - c. Desplegar la carpeta “Otros”
 - d. Seleccionar la carpeta “Modbus TCP”
 - e. Visualizar los bloques “MB_CLIENT” y “MB_SERVER”

Figura 62

Herramienta Modbus TCP

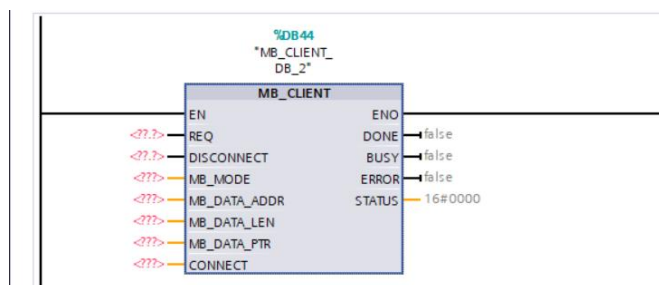


2. Seleccionar bloque MB_CLIENT

El protocolo MODBUS TCP/IP tiene arquitectura Cliente – Servidor, en esta ocasión el PLC S7-1200 va a funcionar como cliente y la placa Arduino tomara la función de servidor, es por ello que se debe hacer uso del bloque MB_CLIENT, tal como se indica en la Figura 63.

Figura 63

Bloque MB_CLIENT



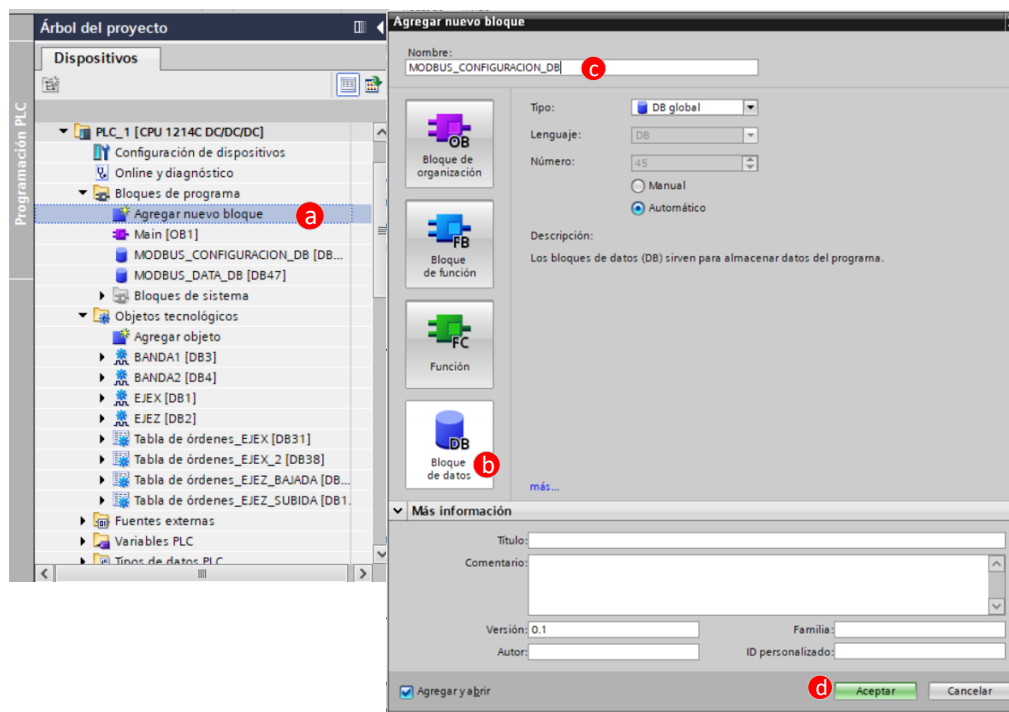
3. Creación de Base de datos MODBUS_CONFIGURATION_DB

Como se puede observar en Figura 63, el bloque requiere de varios parámetros, es por ello que se ha decidido crear una Base de datos como se muestra en Figura 64, en la cual se almacena los datos de configuración del bloque MB_CLIENT.

- a. Dentro de la carpeta “Bloques de programa”, escoger la opción “Agregar nuevo Bloque”
- b. Escoger la opción “Bloque de datos”
- c. Ingresar el nombre MODBUS_CONFIGURATION_DB para esta base de datos.
- d. Clic en Aceptar.

Figura 64

Creación de un Bloque Base de Datos



4. Ingreso de variables en la Base de datos MODBUS_CONFIGURACION_DB

Las variables que se necesitan crear son las que se muestran en la Figura 65. Se consideran variables de lectura y escritura ya que se va enviar información desde el PLC hasta el Arduino y viceversa.

Figura 65

Variables creadas en Base de datos MODBUS_CONFIGURACION_DB

MODBUS_CONFIGURACION_DB								
	Nombre	Tipo de datos	Valor de arranq...	Remanen...	Accesible d...	Escrib...	Visible en ..	Valor de a..
1	Static							
2	LEER_REQ	Bool	false	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
3	LEER_DISCONNECT	Bool	FALSE	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
4	LEER_MB_MODE	USInt	103	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
5	LEER_MB_DATA_ADDR	UDInt	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
6	LEER_MB_DATA_LEN	UInt	10	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
7	LEER_CONNECT	TCON_IP_v4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
8	ESCRIBIR_REQ	Bool	false	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
9	ESCRIBIR_DISCONNECT	Bool	FALSE	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
10	ESCRIBIR_MB_MODE	USInt	116	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
11	ESCRIBIR_MB_DATA_A...	UDInt	5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
12	ESCRIBIR_MB_DATA_L...	UInt	3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
13	CONNECT_ESCRIBIR	TCON_IP_v4		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Para conocer el tipo de dato que requiere el bloque MB_CLIENT en cada terminal, hay que dar clic sobre éste y presionar F1 del teclado, con lo cual se despliega una ventana que contiene una tabla de información detallada. De esa información se ha realizado un resumen que se presenta en la Tabla 6.

Tabla 6

Descripción MB_CLIENT

Parámetro	Declaración	Tipo de datos	Descripción
REQ	Input	BOOL	El parámetro REQ se controla por nivel. Así, mientras la entrada esté activada (REQ=true), la instrucción enviará órdenes de comunicación
DISCONNECT	Input	BOOL	0: Establecer conexión de comunicación con el interlocutor configurado en el parámetro CONNECT (ver parámetro CONNECT). 1: Deshacer la conexión. Durante la desconexión no se ejecuta ninguna otra función. Tras deshacer la conexión correctamente, el parámetro STATUS devuelve el valor 0003.
MB_MODE	Input	USINT	103: Leer de 1 a 125 registros de parada en la dirección remota de 0 a 65.535 106: Escribir 1 registro de retención en la dirección remota de 0 a 65.535
MB_DATA_ADDR	Input	UDINT	en función de MB_MODE
MB_DATA_LEN	Input	UINT	Longitud de datos: Número de bits o palabras para el acceso a los datos
CONNECT	IntOut	TCON_IP_v4	TCON_IP_v4: contiene todos los parámetros de direccionamiento necesarios para establecer una conexión programada. Si se utiliza TCON_IP_v4, la conexión se establece al llamar la instrucción "MB_CLIENT"

Dentro de las variables mostradas en Figura 65, LEER_CONNECT y CONNECT_ESCRIBIR requieren de una configuración extra, la cual se presenta en Figura 66 y

Figura 67 respectivamente. Para llenar los parámetros, se ha utilizado la información proporcionada en los comentarios. Siendo así:

- **ID** un valor que el usuario puede escoger arbitrariamente para identificar la conexión. En este caso 222 para leer y 91 para escribir.
- **ConnectionType** se debe ingresar el valor de 11 para trabajar en MODBUS TCP/IP
- **ActiveEstablished** se debe seleccionar TRUE para activarlo.
- **ADDR** es un arreglo en el cual se ingresa la dirección IP de la Shield Ethernet W5100 que en este caso es 192.168.0.30
- **RemotePort y LocalPort** es el número de puerto remoto y local respectivamente, en el cual se ingresa un valor de tipo UInt (números enteros de 32 bits sin signo), en este caso el valor es de 502.

Figura 66

Configuración de variable LEER_CONNECT

7	LEER_CONNECT	TCON_IP_v4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
8	Interfaceld	HW_ANY	64	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	HWIdentifier of IE-interface submodule
9	ID	CONN_OUC	222	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	connection reference / identifier
10	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 (17=TCP)
11	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
12	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)
13	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
14	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
15	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
16	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
17	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
18	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
19	LocalPort	UInt	502	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	local UDP/TCP port number

Figura 67

Configuración de variable CONNECT_ESCRIBIR

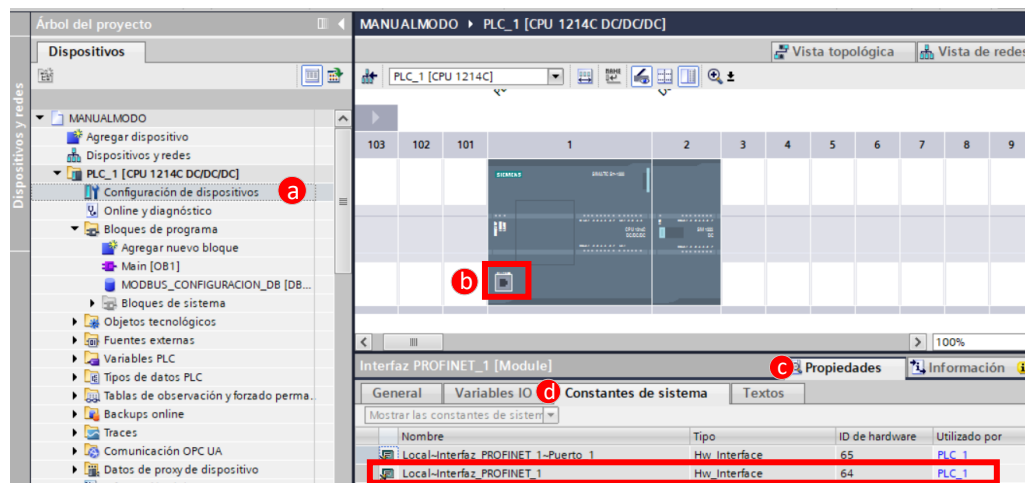
25	CONNECT_ESCRIBIR	TCON_IP_v4		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
26	Interfaceld	HW_ANY	64	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	HWIdentifier of IE-interface submodule
27	ID	CONN_OUC	91	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	connection reference / identifier
28	ConnectionType	Byte	11	<input 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 (17=TCP)
29	ActiveEstablished	Bool	TRUE	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	active/passive connection establishment
30	RemoteAddress	IP_V4		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	remote IP address (IPv4)
31	ADDR	Array[1..4] of Byte		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	IPv4 address
32	ADDR[1]	Byte	192	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	IPv4 address
33	ADDR[2]	Byte	168	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	IPv4 address
34	ADDR[3]	Byte	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	IPv4 address
35	ADDR[4]	Byte	30	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	IPv4 address
36	RemotePort	UInt	502	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	remote UDP/TCP port number
37	LocalPort	UInt	502	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	local UDP/TCP port number

Para completar la configuración de las variables CONNECT, hace falta identificar el parámetro **HW_ANY**, este dato se lo obtiene de la interfaz profinet del PLC, ya que es un dato propio del hardware, como se lo observa en Figura 68.

- a. Configuración de dispositivos
- b. Clic en Interfaz Profinet
- c. Clic en Propiedades
- d. Clic en Constantes del sistema

Figura 68

Identificador HW del submódulo de interfaz IE



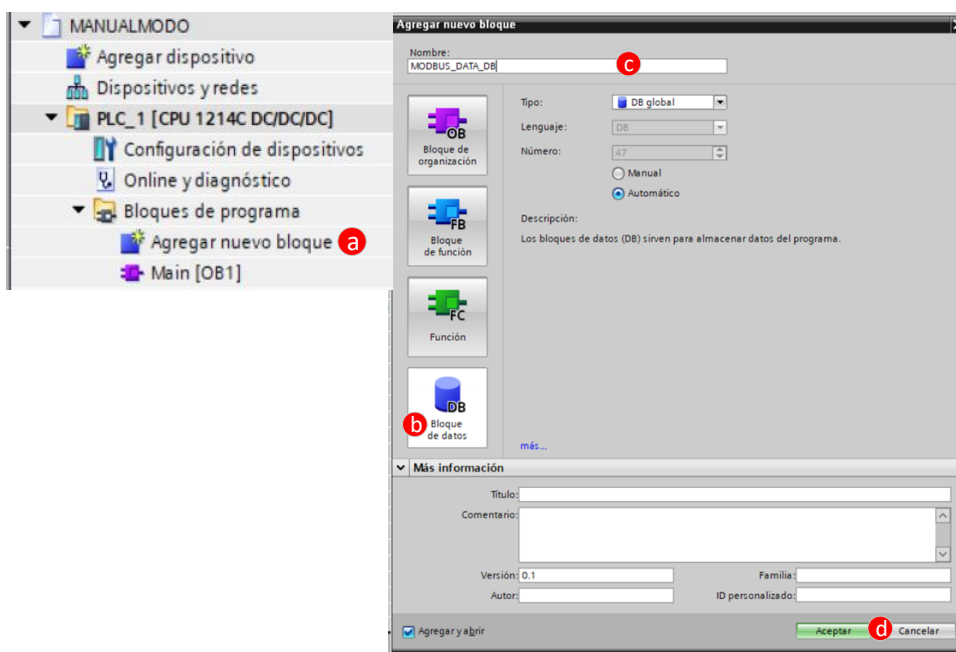
5. Creación de base de datos MODBUS_DATA_DB

Se ve necesaria la creación de una segunda base de datos, en esta se van almacenar los datos enviados y recibidos desde el PLC hacia el Arduino y del Arduino hacia el PLC, como se visualiza en Figura 69.

- a. Dentro de la carpeta “Bloques de programa”, escoger la opción “Agregar nuevo Bloque”
- b. Escoger la opción “Bloque de datos”
- c. Ingresar el nombre *MODBUS_DATA_DB* para esta base de datos.
- d. Clic en Aceptar.

Figura 69

Creación de base de datos MODBUS_DATA_DB



Una vez dentro de la base de datos MODBUS_DATA_DB, se debe crear un arreglo de 10 registros con el nombre Recibidos y otro arreglo de 3 registros con el nombre de Enviados como se observa en Figura 70.

Figura 70

Creación de Array Recibidos y Enviados

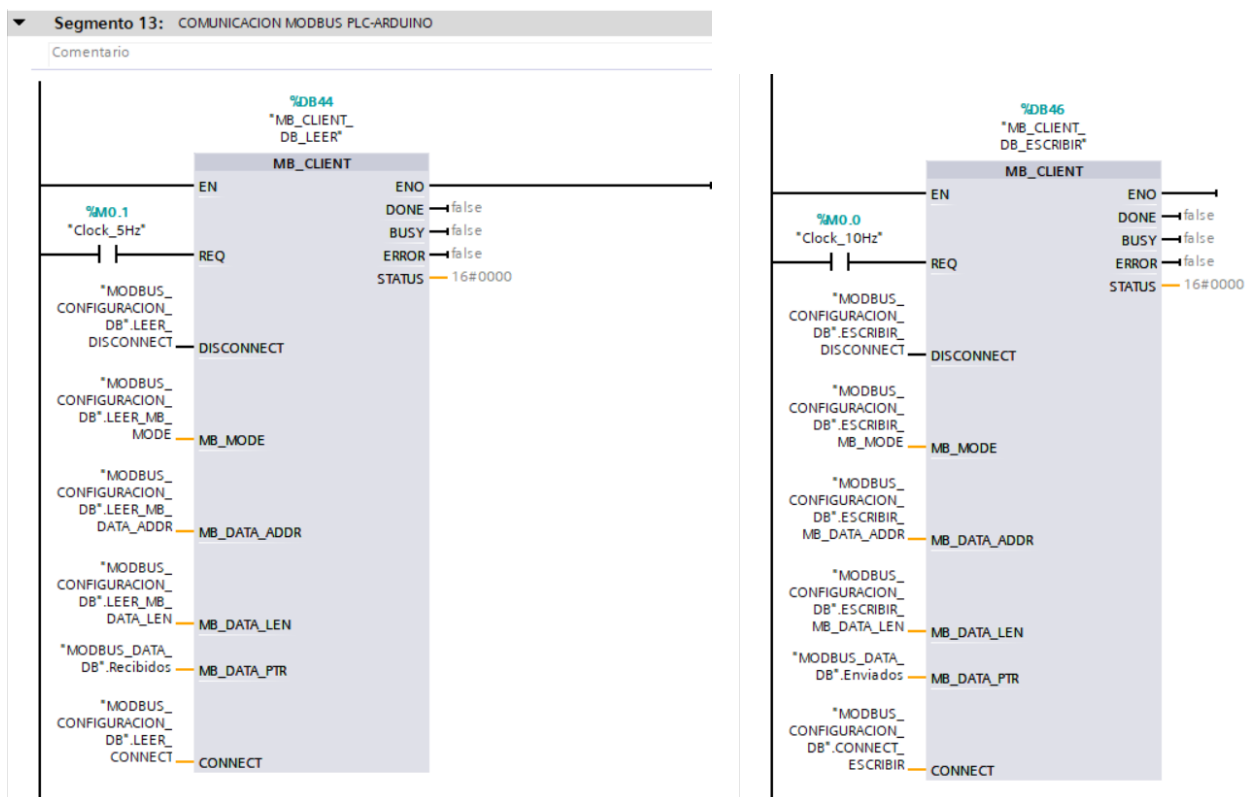
MANUALMODO > PLC_1 [CPU 1214C DC/DC/DC] > Bloques de programa > MODBUS_DATA_DB [DB47]								
MODBUS_DATA_DB								
	Nombre	Tipo de datos	Valor de arranq...	Remanen...	Accesible d...	Escrib...	Visible en ..	Valor de a...
1	Static			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2	Recibidos	Array[0..9] of Int		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
3	Recibidos[0]	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
4	Recibidos[1]	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
5	Recibidos[2]	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
6	Recibidos[3]	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
7	Recibidos[4]	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
8	Recibidos[5]	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
9	Recibidos[6]	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
10	Recibidos[7]	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
11	Recibidos[8]	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
12	Recibidos[9]	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
13	Enviados	Array[0..9] of Int		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
14	Enviados[0]	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
15	Enviados[1]	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
16	Enviados[2]	Int	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

6. Asignación de variables creadas en bases de datos al bloque MB_CLIENT

Finalmente, después de haber configurado todos los parámetros necesarios para la comunicación MODBUS TCP/IP, se procede a crear dos bloques MB_CLIENT, el primero será para lectura y el segundo será utilizado para escritura tal como se observa en Figura 71, la lectura esta configura a una frecuencia de 5Hz y la escritura a una frecuencia de 10Hz.

Figura 71

Segmento Comunicaciones MODBUS TPC/IP entre PLC- Arduino



Capítulo IV

Pruebas y análisis de resultados

En este capítulo se muestran los resultados obtenidos de las pruebas realizadas al módulo didáctico de transporte y organización de envases para bebidas en modo manual, se analiza el desenvolvimiento de las cuatro etapas y a la vez se presenta un análisis estadístico de las pruebas realizadas al sistema.

Los resultados que se presentan a continuación han sido tomados en modo manual, la selección del modo de operación se lo realiza desde el tablero de control, posicionando el selector en el apartado “Manual” como se observa en Figura 72.

Figura 72

Selector en Modo Manual

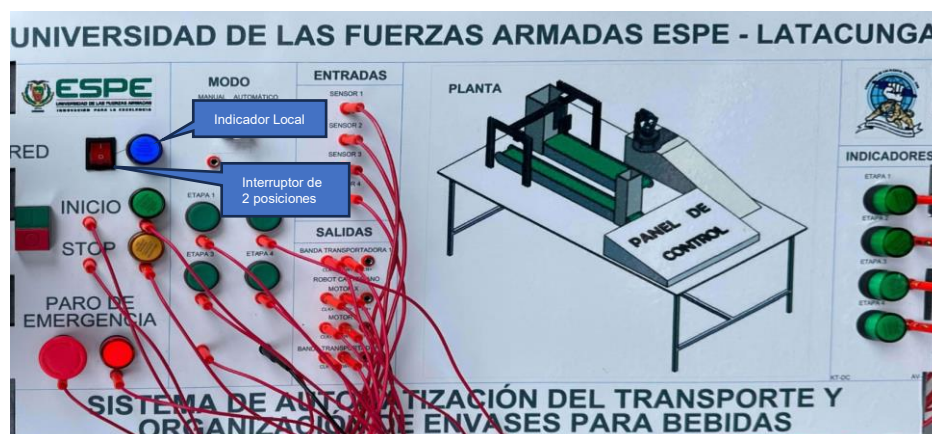


Conexión de la estación didáctica a la red de 110VAC

La estación didáctica posee un indicador local de color azul, el cual se enciende cuando hay conexión a la red de 110 VAC, esto se lo hace mediante un Interruptor de dos posiciones, como se muestra en Figura 73.

Figura 73

Conexión de la estación didáctica a la RED de 110VAC

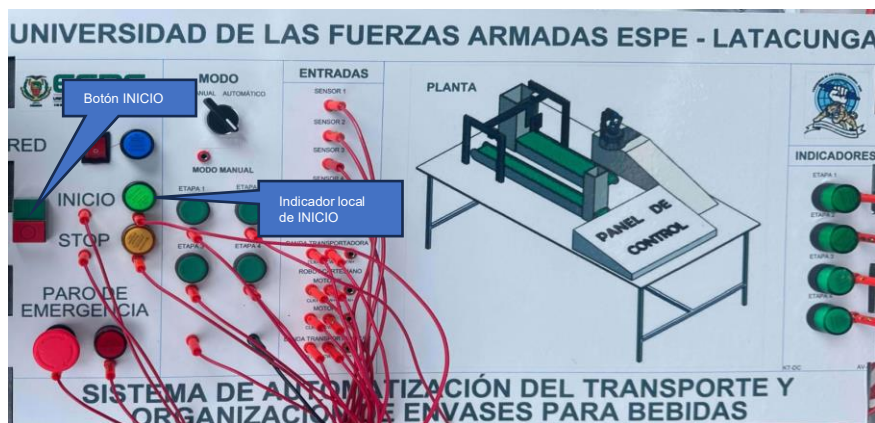


Botonera de Inicio, Stop y Paro de Emergencia de la estación didáctica.

Cuando se presiona el botón INICIO, arranca el funcionamiento en modo manual y enciende un indicador local de color verde como se puede ver en Figura 74.

Figura 74

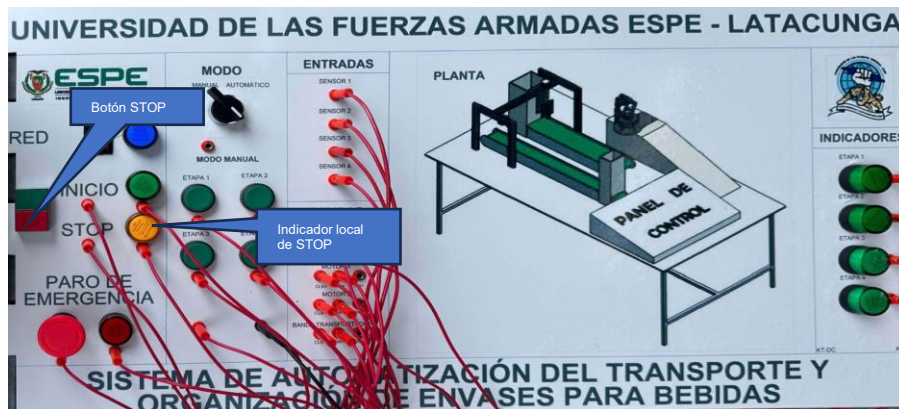
Accionamiento de botón INICIO



El botón STOP, detiene la operación en modo manual, regresa a posición home todos los motores y la manera de verificar que este botón ha sido accionado es mediante el indicador de color naranja que se muestra en Figura 75.

Figura 75

Accionamiento de botón STOP



Al accionar el botón PARO DE EMERGENCIA, todo el sistema se detiene en el estado en que se encuentre, para los motores y desactiva el robot manipulador, la manera de visualizar dicha acción es mediante el indicador local de color rojo que se muestra en Figura 76.

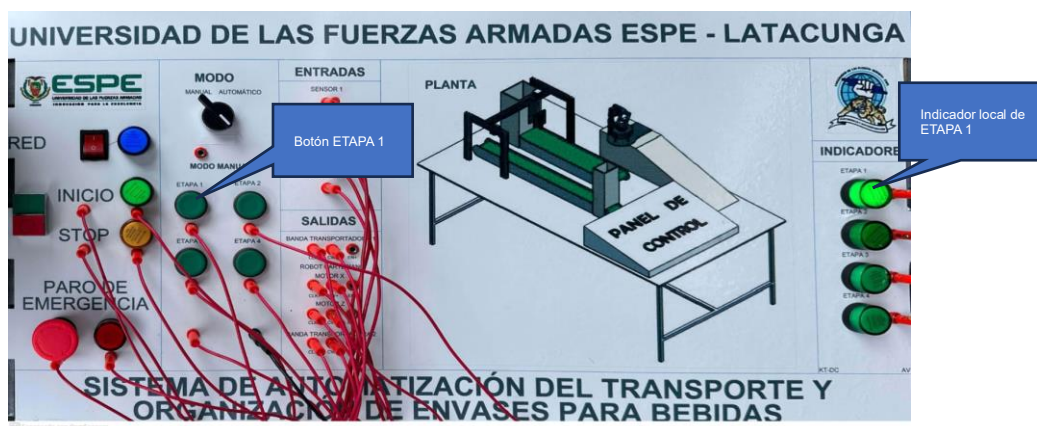
Figura 76

Accionamiento de botón PARO DE EMERGENCIA



Respuesta de Etapa 1

Al presionar el botón ETAPA 1 empieza el funcionamiento de la etapa y enciende un indicador local de color verde como se mira en Figura 77.

Figura 77*Indicador de ETAPA 1*

Dentro de esta etapa se evaluó el comportamiento del sensor fotoeléctrico PE-R05D ante materiales claros, con la finalidad de ver la capacidad de respuesta que proporciona este sensor ante los envases seleccionados en Figura 78, por lo cual se ha realizado 30 pruebas.

Figura 78*Envase seleccionado para pruebas de funcionamiento*

El criterio para dar como válido el intento es que la banda 1 debe detenerse en la posición designada que se muestra en Figura 79, si la banda no se detiene es porque el sensor no logra detectar el envase a tiempo y se considerara rechazada. Los resultados se presentan en Tabla 7, donde “1” es aceptado y “0” es rechazo.

Figura 79*Sensado de envase en Etapa 1*

Los resultados que se obtuvieron de los 30 intentos son:

- Numero de envases detectados correctamente: 28
- Numero de envases detectados incorrectamente: 2

Tabla 7*Resultados de funcionamiento de Etapa 1*

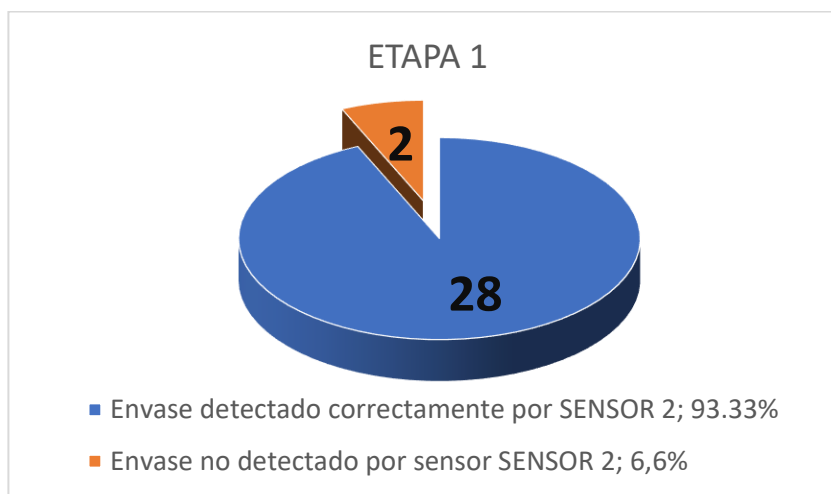
No. de prueba	ETAPA 1	
	Envase detectado correctamente por SENSOR 2	Envase no detectado por sensor SENSOR 2
1	1	0
2	1	0
3	1	0
4	1	0
5	1	0
6	1	0
7	1	0
8	1	0
9	1	0
10	1	0
11	1	0
12	1	0
13	1	0
14	1	0
15	1	0
16	1	0
17	1	0
18	1	0

No. de prueba	ETAPA 1	
	Envase detectado correctamente por SENSOR 2	Envase no detectado por sensor SENSOR 2
19	1	0
20	1	0
21	1	0
22	1	0
23	0	1
24	1	0
25	1	0
26	1	0
27	1	0
28	1	0
29	0	1
30	1	0
TOTAL	28	2

Se presenta una gráfica de pastel en Figura 80, de la cual se observa que el 93,3% de las pruebas han sido satisfactorias y el 6,6% han sido consideradas como rechazadas.

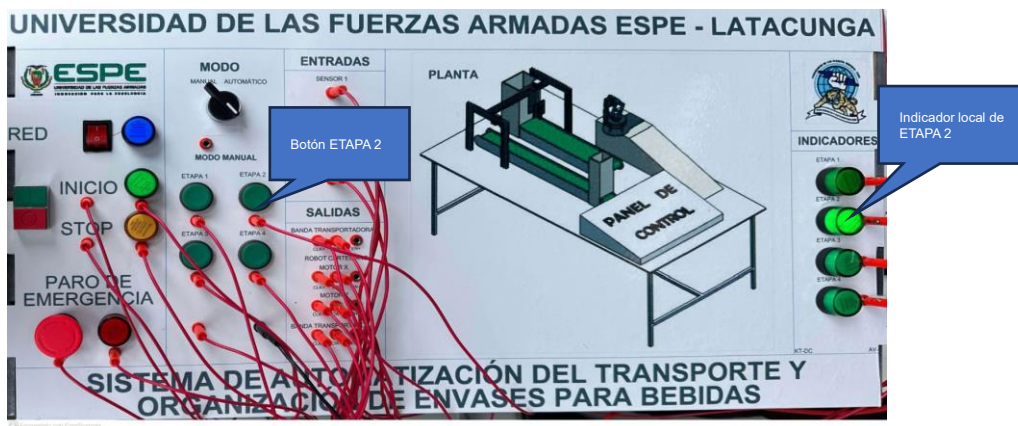
Figura 80

Gráfica de pastel de resultados obtenidos de Etapa 1



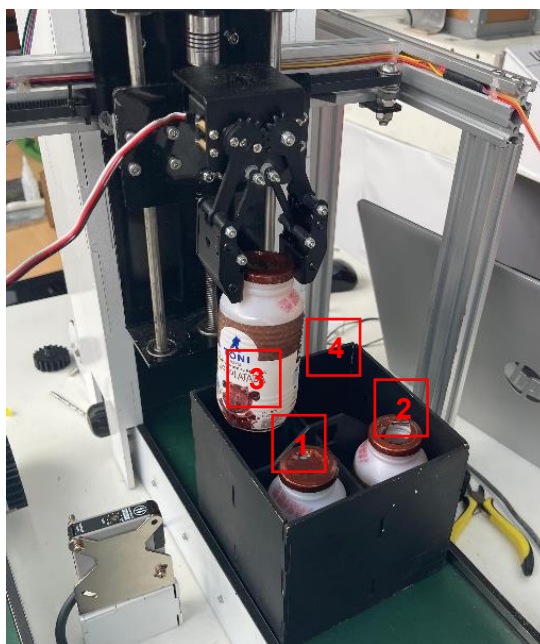
Respuesta de Etapa 2

Al presionar el botón ETAPA 2 empieza el funcionamiento de la etapa y enciende un indicador local de color verde como se mira en Figura 81.

Figura 81*Indicador de ETAPA 2*

La segunda etapa tiene como criterio analizar el agarre correcto del envase con el gripper del robot cartesiano, el depósito correcto en las posiciones 1, 2, 3 y 4 de la jaba. Es necesario mencionar que las pruebas han sido realizadas en secuencia, es decir los intentos de la Etapa 1, Etapa 2, Etapa 3 y Etapa 4 son los mismos respectivamente.

En la Figura 82, se presentan las posiciones internas de cada jaba.

Figura 82*Posiciones de la jaba*

Un dato importante a ser mencionado es que, el gripper del robot cartesiano es activado mediante comunicación MODBUS TCP/IP, así que en este apartado también se evaluara la respuesta de la comunicación implementada. Los resultados obtenidos se presentan en la Tabla 8.

Tabla 8

Resultados de funcionamiento de Etapa 2

No. de prueba	ETAPA 2				
	Agarra correctamente el envase	Deposita correctamente en posición 1	Deposita correctamente en posición 2	Deposita correctamente en posición 3	Deposita correctamente en posición 4
		1	2	3	4
1	1	1	1	1	1
2	1	0	1	1	1
3	1	1	0	1	1
4	1	1	1	1	1
5	1	1	1	0	1
6	1	1	1	1	1
7	1	1	1	1	1
8	1	1	1	1	1
9	1	1	1	1	1
10	0	0	1	1	1
11	1	1	1	0	1
12	1	1	1	1	1
13	1	1	1	1	1
14	0	1	1	1	1
15	1	1	1	1	1
16	1	1	0	0	0
17	1	1	1	1	1
18	1	1	0	1	1
19	1	1	1	1	1
20	1	1	1	1	1
21	1	1	1	1	1
22	1	1	1	1	1
23	0	1	1	1	0
24	1	1	1	1	1
25	1	1	1	1	1
26	1	1	1	1	1
27	1	1	1	0	1
28	1	1	1	1	1

No. de prueba	ETAPA 2				
	Agarra correctamente el envase	Deposita correctamente en posición 1	Deposita correctamente en posición 2	Deposita correctamente en posición 3	Deposita Correctamente en posición 4
29	0	1	1	1	1
30	1	1	1	1	1
TOTAL	28	29	28	27	29

De la Tabla 8, se puede sacar los siguientes datos:

- Agarra correctamente el envase: 28 intentos
- Deposita correctamente en posición 1: 29 intentos
- Deposita correctamente en posición 2: 28 intentos
- Deposita correctamente en posición 3: 27 intentos
- Deposita correctamente en posición 4: 29 intentos

De la Figura 84, se observa que el agarre correcto del envase tiene el mismo porcentaje de aceptación que la etapa 1, es decir que, para tener un buen agarre del envase, es necesario que éste se encuentre en la posición adecuada y para ello se requiere que el sensor 2 detecte correctamente. Para ser considerado un agarre perfecto, el envase debe estar justo en la mitad del gripper, tal como se presenta en Figura 83.

Figura 83

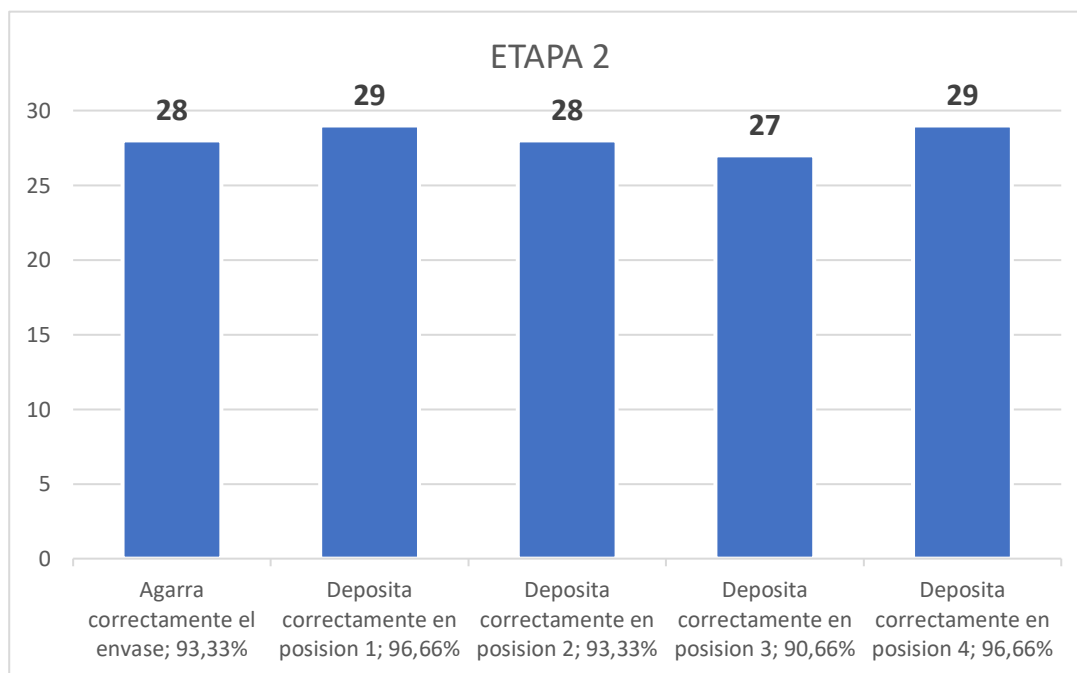
Agarre perfecto del envase.



Se observa en la Figura 84 que, el porcentaje de aceptación del depósito de las 4 posiciones supera el 90%.

Figura 84

Grafica de barras agrupadas de resultados obtenidos de Etapa 2

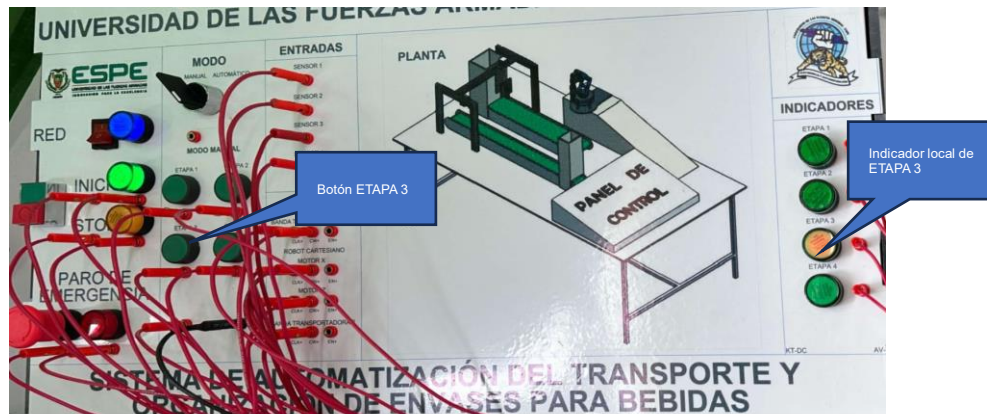


Respuesta de Etapa 3

Al presionar el botón ETAPA 3 empieza el funcionamiento de la etapa y enciende un indicador local de color naranja como se mira en Figura 85.

Figura 85

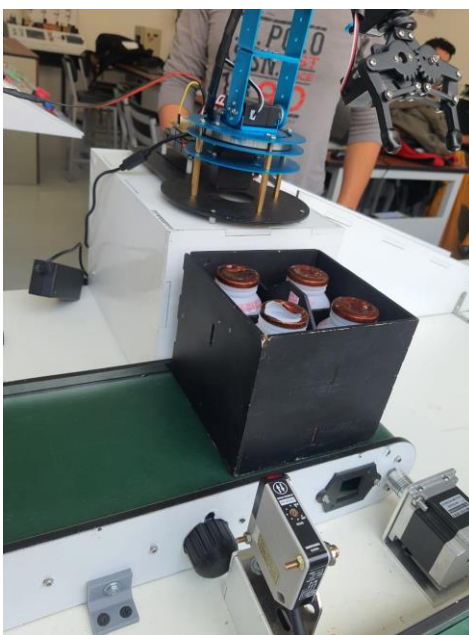
Indicador de ETAPA 3



En la tercera etapa se analiza el comportamiento del sensor fotoeléctrico PE-R05D ante materiales oscuros. Se va a evaluar la detección correcta de las jabs llenas de producto por el sensor 4 como se mira en Figura 86, el criterio que se toma en consideración es que el sensor de una respuesta rápida y adecuada para frenar la banda 2 en la posición predeterminada.

Figura 86

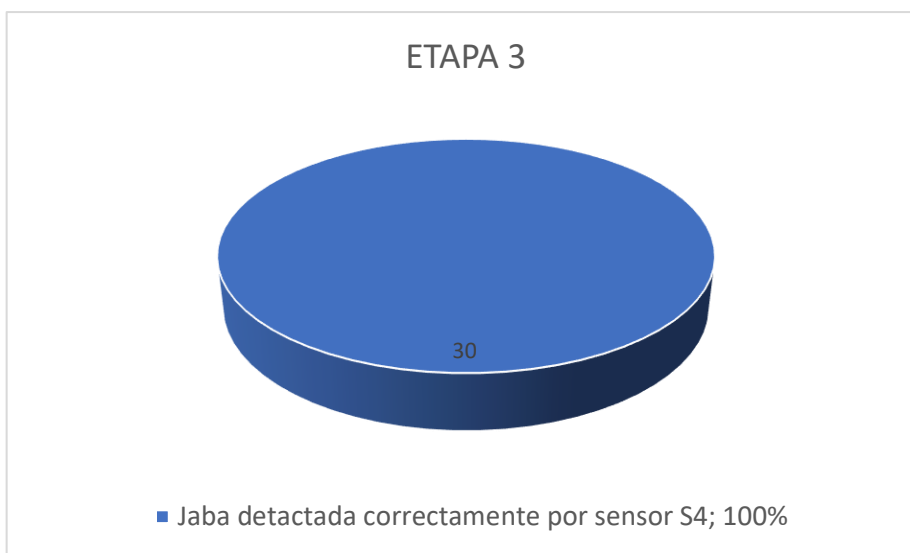
Sensado de jaba por sensor PE-R05D



Se continúan con los 30 intentos de las anteriores etapas, dando como resultado el 100% de aceptación, ya que en todas las pruebas realizadas el sensor detecto de una buena manera a la jaba de color negro. Se presenta una gráfica de pastel en la Figura 87.

Figura 87

Grafica de pastel de resultados obtenidos de Etapa 3



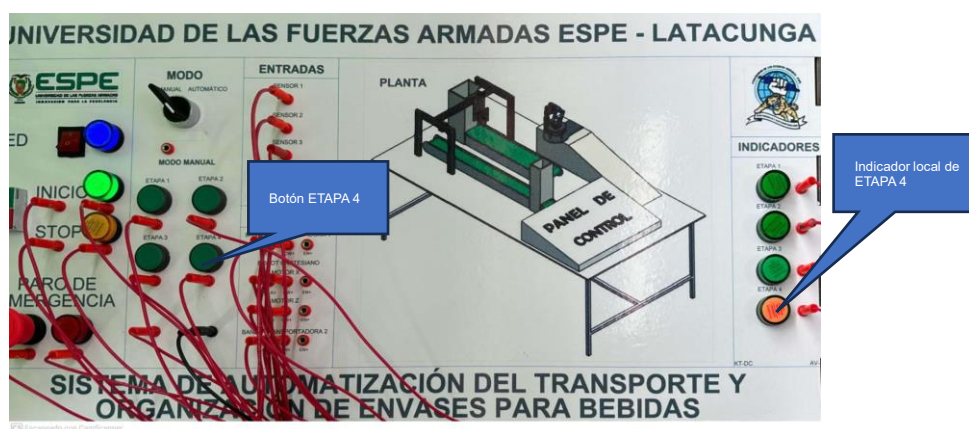
El resultado que se obtuvo de esta prueba es que, el sensor fotoeléctrico PE-R05D detecta de mejor manera a materiales oscuros.

Respuesta de Etapa 4

Al presionar el botón ETAPA 4 empieza el funcionamiento de la etapa y enciende un indicador local de color naranja como se mira en Figura 88.

Figura 88

Indicador ETAPA 4



En la cuarta etapa se analiza la comunicación MODBUS TCP/IP de la señal que se envía desde el PLC hasta la tarjeta Arduino. A la vez se evalúa el comportamiento del robot ante una carga de 210 g.

Los datos que envía y recibe el PLC se puede visualizar en Figura 89, cuando se envía un dato desde el PLC se marca con el valor “11” en **Enviados[2]**, y si la comunicación es satisfactoria la placa Arduino envía “11” y se lo puede visualizar en **Recibidos[2]**, entonces de esta manera se comprobaba si la comunicación es exitosa en los 30 intentos que se desarrollaron.

Figura 89

Comprobación de comunicación mediante MODBUS_DATA_DB

MODBUS_DATA_DB			
	Nombre	Tipo de datos	V
1	Static		
2	Recibidos	Array[0..9] of Int	
3	Recibidos[0]	Int	0
4	Recibidos[1]	Int	0
5	Recibidos[2]	Int	0
6	Recibidos[3]	Int	0
7	Recibidos[4]	Int	0
8	Recibidos[5]	Int	0
9	Recibidos[6]	Int	0
10	Recibidos[7]	Int	0
11	Recibidos[8]	Int	0
12	Recibidos[9]	Int	0
13	Enviados	Array[0..2] o...	
14	Enviados[0]	Int	0
15	Enviados[1]	Int	0
16	Enviados[2]	Int	0

Para el análisis del comportamiento del robot manipulador, se considera si el robot agarra correctamente a la jaba y soporta el peso hasta llegar a la zona de almacén, como se mira en Figura 90.

Figura 90

Levantamiento de jaba con robot manipulador



La Tabla 9 muestra los detalles de las pruebas realizadas en la etapa 4.

Tabla 9

Resultados de funcionamiento de Etapa 4

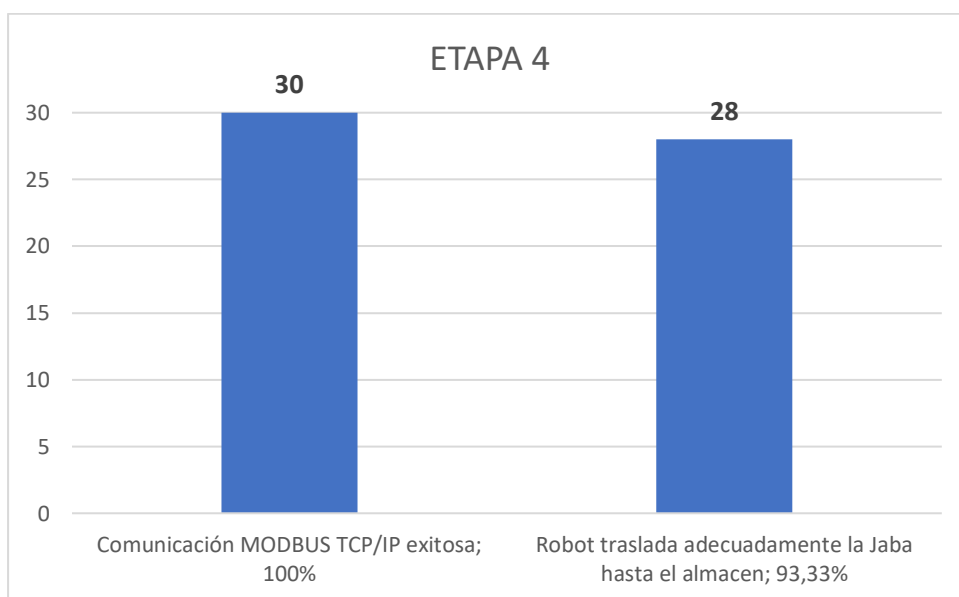
No. de prueba	ETAPA 4	
	Comunicación MODBUS TCP/IP exitosa	Robot trasladada adecuadamente la Jaba hasta el almacén
1	1	1
2	1	1
3	1	1
4	1	1
5	1	1
6	1	1
7	1	1
8	1	1
9	1	1
10	1	1
11	1	1
12	1	1
13	1	1
14	1	1
15	1	1
16	1	1
17	1	1
18	1	1
19	1	1
20	1	1
21	1	1
22	1	1
23	1	1
24	1	1
25	1	1
26	1	1
27	1	0
28	1	1
29	1	1
30	1	0
TOTAL	30	28

Los resultados que presenta la Tabla 9 son los siguientes:

- La comunicación MODBUS TCP/IP es muy segura y confiable ya que presenta un porcentaje del 100% de aceptación en los intentos realizados, como se observa en Figura 91.
- Por otra parte, el robot manipulador presenta una aceptación del 93,3% en sus intentos. Las dos pruebas que se consideran rechazadas es porque el gripper del robot no se cerró en el gancho de la jaba adecuadamente.

Figura 91

Gráfica de barras de Resultados obtenidos de Etapa 4



Comprobación de hipótesis

Se diseñó e implementó la estación didáctica de transporte y organización de envases, obteniendo como resultados valores mayores al 93% de eficiencia en las pruebas de funcionamiento realizadas a las cuatro etapas, en las que se evaluó el desempeño del sensor fotoeléctrico PE-R05D, el comportamiento del robot manipulador ante cargas de 210g y la comunicación MODBUS TCP/IP. El motivo por el cual no llega al 100% de eficiencia es por el criterio estricto que se tomó en las pruebas, no se consideran válidas las pruebas que rozan a la

jaba, pero cabe mencionar que todas las jabas van llenas. Por lo antes mencionado se valida la hipótesis planteada.

Alcances

- La estación didáctica permite el transporte y organización de envases de una manera automatizada, por medio del PLC S7-1200 CPU 1214C DC/DC/DC.
- Las etapas del proceso son activadas en modo manual desde la botonera del tablero de control ubicado en la parte interior de la estructura (mesa), y cuentan con luces indicadores de 24VDC que permite al usuario conocer si dicha etapa está operando.
- La estación cuenta con tecnología plug and play, permitiéndola ser usada como material de estudio para los estudiantes de la carrera de Electrónica y Automatización en el laboratorio de Comunicaciones Industriales.
- El protocolo de comunicación que se usa en la estación didáctica es MODBUS TCP/IP, el cual permite la interacción entre el PLC S7-1200 y la shield Ethernet.

Limitaciones

- Los PLC S7-1200 CPU 1214C DC/DC/DC pueden hacer el control de 4 motores paso a paso como máximo, ya que en la configuración de objetos tecnológicos se puede hacer uso de 4 pulsos sin opción a expansión.
- La estación didáctica debe trabajar con elementos livianos ya que la carga que soporta el robot manipulador es de un máximo $\frac{1}{2}$ libra.
- Ciertos elementos de la estación didáctica no cuentan con características industriales por lo que la planta no puede trabajar por largos periodos de tiempo.

Capítulo V

Conclusiones y recomendaciones

Conclusiones

- El diseño y construcción de la estructura metálica ayudo en la implementación de la estación didáctica, ya que puede soportar el peso de todos los elementos que están sujetos a ésta.
- El sensor fotoeléctrico PE-R05D tiene una mejor respuesta ante objetos de color obscuro, ya que en las pruebas realizadas demostró una eficiencia del 100% con colores oscuros frente a una eficiencia del 93,3% con colores claros.
- La investigación realizada fue de gran ayuda al momento de configurar los dispositivos que intervienen en la automatización, como el Driver TB6560 en el cual se estableció la corriente máxima a la que operara el motor Nema 17 (1.2 A) y el número de micro pasos con los que trabajara (8).
- La característica plug and play que posee la estación didáctica permite a los usuarios realizar el control de la planta de una forma más fácil y simplificada, sin tener que realizar alguna configuración extra y sin la necesidad de conocer a fondo el hardware que está manipulando, y enfocarse a la parte del control y automatización.
- Los motores paso a paso Nema 17 requieren específicamente de un PLC con salidas transistorizadas en las que se pueda hacer uso de los PTO (Pulse Train Output), es decir se requiere un PLC que disponga salidas de impulso y sentido que trabajen a una frecuencia de 100khz.
- EL protocolo de comunicación MODBUS TCP/IP implementado en la estación didáctica requiere de una configuración detallada, pero con ayuda del bloque MB_CLIENT se simplifica de una manera considerable a la vez que su respuesta es

excelente una vez que entra en funcionamiento, no existe perdidas de datos y su respuesta es instantánea alrededor de los 100ms.

- La estación didáctica dispone de un control secuencial por cuatro etapas basado en la normativa ISA S 88, en el cual se describe de forma coherente las operaciones que realiza cada etapa, además incluye la adaptabilidad de procesos, es decir que el sistema puede ser reconfigurado para cambiar el producto según la demanda requerida y así usar la estación para el traslado y organización de diferentes envases.

Recomendaciones

- Verificar que el robot cartesiano inicie su operación desde los puntos exactos preestablecidos, ya que si el robot inicia su operación desde algún otro punto puede causar golpes entre los envases y el gripper o desgastes de otros elementos mecánicos.
- Verificar que los conectores banana estén correctamente introducidos, para que no existan falsos contactos y así prevenir un mal funcionamiento del sistema.
- Antes de energizar el sistema se recomienda mover suavemente los engranes de las dos bandas, con la finalidad de quitar alguna adherencia existente entre la lona de la banda con la estructura interna de la misma, con la finalidad de evitar carga excesiva para el motor.

Bibliografía

- Abad, J. M. H., Sandoval, J. A. H., Carrillo, J. A. V., & Flores, E. (2023). Sistema de Modulo Didáctico para el Control del Proceso de Llenado de Recipientes Mediante PLC. *Ciencia Latina Revista Científica Multidisciplinar*, 7(5), 1929-1944.
https://doi.org/10.37811/cl_rcm.v7i5.7853
- Andrea, E. (1980). *CINTAS' TRANSPORTADORAS*.
https://ocw.unican.es/pluginfile.php/2439/course/section/2474/12._cintas.pdf
- Ayuningtyas, B., & Ilman, S. (2021). IP CAMERA SURVEILLANCE SYSTEM USING ANDROID APPLICATION BASED ON ARDUINO. *Jurnal Teknik dan Informatika (JTI)*, 1(1), 1-12.
<https://doi.org/10.52909/jti.v1i1.6>
- Cadena, W. (2016). *Desarrollo de la automatización de las estaciones que componen la línea de producción Crown Pet dentro de la planta de Orangine durante el año 2015*.
<https://dspace.ups.edu.ec/bitstream/123456789/12359/1/UPS%20-%20ST002200.pdf>
- Caisabanda, M., & Choloquina, E. (2022). *AUTOMATIZACIÓN DEL PROCESO DE PASTEURIZACIÓN DE LECHE EN LA FÁBRICA DE HELADOS IGLÚ DEL CANTÓN PUJILÍ PROVINCIA DE COTOPAXI*.
<http://repositorio.utc.edu.ec/bitstream/27000/9690/1/PI-002201.pdf>
- Cera-Martínez, D., Ortiz-Sandoval, J. E., & Gualdrón-Guerrero, O. E. (2018). Sintonización de un controlador de temperatura a través de un autómeta programable. *Revista de Investigación, Desarrollo e Innovación*, 9(1), Article 1.
<https://doi.org/10.19053/20278306.v9.n1.2018.8513>
- Changzhou, S. (2020). *NEMA 17 Stepper Motor*. Components101.
<https://components101.com/motors/nema17-stepper-motor>
- Dolinay, J., Dostálek, P., & Vašek, V. (2021). Advanced Debugger for Arduino. *International Journal of Advanced Computer Science and Applications*, 12(2).
<https://doi.org/10.14569/IJACSA.2021.0120204>

- ESA Education. (2020). *Brazo robotico. Ingenieros espaciales por un día*. <https://esero.es/wp-content/uploads/2021/05/brazo-robotico.pdf>
- Gogineni, V. R., Matcha, K., & K, R. R. (2015). Real Time Domestic Power Consumption Monitoring using Wireless Sensor Networks. *International Journal of Electrical and Computer Engineering (IJECE)*, 5(4), Article 4. <https://doi.org/10.11591/ijece.v5i4.pp685-694>
- González, O. (2024). *Guía de modelos Arduino y sus características | Arduino MEGA 2560 | BricoGeek Lab*. Modelos de Arduino. <https://lab.bricogeek.com/tutorial/guia-de-modelos-arduino-y-sus-caracteristicas/arduino-mega-2560>
- HANYOUNG NUX. (2012). *PE series*.
https://hanyoungnux.com/sub01_01_01_view.php?lcode=05&mcode=0503&pcode=1305100080&score=0503002
- Iza Laica, K. A., & Tapia Velásquez, R. I. (2023). *“Repotenciación y automatización de una clasificadora de granos para el Proyecto de Granos Andinos de la Universidad Técnica de Cotopaxi”* [bachelorThesis, Ecuador : Latacunga: Universidad Técnica de Cotopaxi (UTC)]. <http://repositorio.utc.edu.ec/handle/27000/10349>
- Jaramillo, M. T. (2015). *Repositorio de la Universidad de Fuerzas Armadas ESPE: Diseño e implementación de un sistema de clasificación didáctico para supervisar un proceso industrial en el laboratorio de hidrónica y neutrónica*.
<http://repositorio.espe.edu.ec/jspui/handle/21000/10049>
- LewanSoul. (2023). *Amazon.com: LewanSoul Kit de brazo robótico 6DOF brazo de robot de programación con mango PC Software y control de APP con tutorial: Juguetes y Juegos*.
https://www.amazon.com/LewanSoul-rob%C3%B3tico-programaci%C3%B3n-Software-tutorial/dp/B074T6DPKX?ref_=ast_sto_dp
- Muñoz, D. R. (2023). *Robots cartesianos: Qué son y cómo funcionan*. Plastico.
<https://www.plastico.com/es/noticias/robots-cartesianos-que-son-y-como-funcionan>

- Naranjo, G. R. H. (2019). *AUTOMATIZACIÓN DEL SISTEMA DE TRANSPORTE POR CADENAS EN UNA LÍNEA EMBOTELLADORA DE BEBIDAS*.
- Naylamp. (2024). *Shield Ethernet W5100*. Naylamp Mechatronics - Perú.
<https://naylampmechatronics.com/ardusystem-shields/12-shield-ethernet-w5100.html>
- Núñez Solano, S. J. (2017). Automatización de los procesos de secado y selección del cacao ecuatoriano conservando las características sensoriales y previniendo riesgos laborales. *Repositorio de Tesis - UNMSM*.
<https://cybertesis.unmsm.edu.pe/handle/20.500.12672/6841>
- PBC Linear. (2020). *Data-Sheet-Stepper-Motor-Support*. <https://pages.pbclinear.com/rs/909-BFY-775/images/Data-Sheet-Stepper-Motor-Support.pdf>
- Santos, A. M. dos, Santos, G. S., & Coelho, A. S. (2021). Internet das Coisas: Utilização do Arduino para o combate ao desperdício elétrico no IFMA Campus São João dos Patos. *Research, Society and Development*, 10(3), Article 3. <https://doi.org/10.33448/rsd-v10i3.5794>
- SIEMENS. (2023). *Data sheet 6ES7214-1AG40-0XB0 .pdf*. <https://docs.rs-online.com/4ed5/0900766b81397276.pdf>
- Tufiño, B., Stalin, V., Amaguaña, T., Felipe, L., Cañizares, I. P., & Mg, Z. A. (2023). *AUTOMATIZACIÓN DE LAS CAMPANAS DE LA IGLESIA NUESTRA SEÑORA DE EL SALTO DE LA CIUDAD DE LATACUNGA, PROVINCIA DE COTOPAXI*.
- Turbosquid. (2017, febrero 5). *Banda transportadora*. <https://www.turbosquid.com/es/3d-models/3d-model-conveyor/1120565>
- Unit Electronics. (2024). Driver TB6560 Controlador Motor A Pasos. *UNIT Electronics*.
<https://uelectronics.com/producto/driver-tb6560-controlador-motor-a-pasos/>
- Yamazato, T., & Haruyama, S. (2014). Image Sensor Based Visible Light Communication and Its Application to Pose, Position, and Range Estimations. *IEICE Transactions on Communications*, E97.B(9), 1759-1765. <https://doi.org/10.1587/transcom.E97.B.1759>

Anexos