



**UNIVERSIDAD CATÓLICA  
DE SANTIAGO DE GUAYAQUIL**

**FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO  
CARRERA DE INGENIERÍA ELECTRÓNICA EN CONTROL Y AUTOMATISMO**

**TEMA:**

**“Simulación de banda clasificadora de Courier, utilizando Factory IO y Tia Portal de Siemens que controlarán actuadores y motores con PLC’s, para evitar pérdidas económicas en sus clientes.”**

**AUTOR:**

Calle Asencio, Franklin Paul

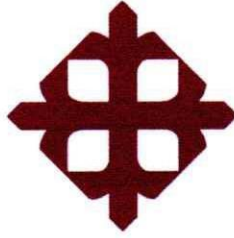
Trabajo de Titulación previo a la obtención del título de  
**INGENIERO ELECTRÓNICO EN CONTROL Y AUTOMATISMO**

**TUTOR:**

Ing. Suarez Murillo Efraín Oswaldo

**Guayaquil, Ecuador**

**12 de febrero del 2023**



**UNIVERSIDAD CATÓLICA  
DE SANTIAGO DE GUAYAQUIL**

**FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO  
INGENIERÍA ELECTRÓNICA EN CONTROL Y AUTOMATISMO**

**CERTIFICACIÓN**

Certificamos que el presente trabajo fue realizado en su totalidad por el Sr. Calle Asencio, Franklin Paul como requerimiento para la obtención del título de INGENIERO ELECTRÓNICO EN CONTROL Y AUTOMATISMO.

**TUTOR**

---

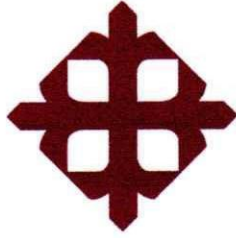
M. Sc. Suarez Murillo Efraín Oswaldo

**DIRECTOR DE CARRERA**

---

M. Sc. Bohórquez Escobar Celso Bayardo

Guayaquil, a los 12 días del mes de febrero del año 2023



**UNIVERSIDAD CATÓLICA  
DE SANTIAGO DE GUAYAQUIL**

**FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO  
INGENIERÍA ELECTRÓNICA EN CONTROL Y AUTOMATISMO**

**DECLARACIÓN DE RESPONSABILIDAD**

**Yo, Calle Asencio, Franklin Paul**

**DECLARÓ QUE:**

El trabajo de titulación **“Simulación de banda clasificadora de Courier, utilizando Factory IO y Tia Portal de Siemens que controlarán actuadores y motores con PLC’s, para evitar pérdidas económicas en sus clientes.”** previo a la obtención del Título de **Ingeniero Electrónico en Control y Automatismo**, ha sido desarrollado respetando derechos intelectuales de terceros conforme las citas que constan en el documento, cuyas fuentes se incorporan en las referencias o bibliografías. Consecuentemente este trabajo es de mi total autoría.

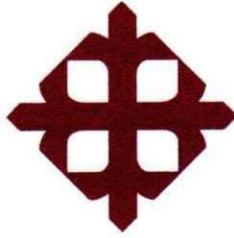
En virtud de esta declaración, me responsabilizo del contenido, veracidad y alcance del Trabajo de Titulación referido.

Guayaquil, a los 12 días del mes de febrero del año 2023

**EL AUTOR**

---

**CALLE ASENCIO, FRANKLIN PAUL**



**UNIVERSIDAD CATÓLICA  
DE SANTIAGO DE GUAYAQUIL**

**FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO  
INGENIERÍA ELECTRÓNICA EN CONTROL Y AUTOMATISMO**

**AUTORIZACIÓN**

**Yo, Calle Asencio, Franklin Paul**

Autorizó a la Universidad Católica de Santiago de Guayaquil, la publicación, en la biblioteca de la institución del Trabajo de Titulación: **“Simulación de banda clasificadora de Courier, utilizando Factory IO y Tia Portal de Siemens que controlarán actuadores y motores con PLC’s, para evitar pérdidas económicas en sus clientes.”**, cuyo contenido, ideas y criterios son de mi exclusiva responsabilidad y total autoría.

Guayaquil, a los 12 días del mes de febrero del año 2023

**EL AUTOR**

---

**CALLE ASECIO, FRANKLIN PAUL**

## REPORTE DE URKUND

### REVISIÓN DE TRABAJO DE TITULACIÓN CON PROGRAMA URKUND

### CARRERA DE INGENIERÍA ELECTRÓNICA EN CONTROL Y AUTOMATISMO

**TÍTULO:** Simulación de banda clasificadora de Courier, utilizando Factory IO y Tia Portal de Siemens que controlarán actuadores y motores con PLC's, para evitar pérdidas económicas en sus clientes.

**AUTOR:** Franklin Paul Calle Asencio



#### Document Information

Analyzed document	Tesis Franklin Calle2.docx (D154930463)
Submitted	1/5/2023 5:28:00 AM
Submitted by	
Submitter email	efrain.velez@cu.ucsg.edu.ec
Similarity	0%
Analysis address	efrain.velez.ucsg@analysis.orkund.com

#### Sources included in the report

<b>W</b>	URL: <a href="https://industriasgsl.com/blogs/automatizacion/arquitectura-del-plc#~:text=La%20arquitectura%...">https://industriasgsl.com/blogs/automatizacion/arquitectura-del-plc# ~:text=La%20arquitectura%...</a> Fetched: 1/5/2023 5:28:00 AM		1
<b>SA</b>	<b>TESIS_CAISABANDA MARCO_CHOLOQUINGA_EDWIN.pdf</b> Document TESIS_CAISABANDA MARCO_CHOLOQUINGA_EDWIN.pdf (D143489417)		1
<b>SA</b>	<b>TESIS PRE-DEFENSA_correccion.pdf</b> Document TESIS PRE-DEFENSA_correccion.pdf (D143366461)		1

Después de analizar el resultado enviado por el programa Urkund, se determinó que el trabajo de titulación del estudiante: Calle Asencio Franklin Paul, observa un porcentaje de 0 % de coincidencias con otros documentos encontrados en el internet.

#### TUTOR

M. Sc. Suarez Murillo Efraín Oswaldo

## **AGRADECIMIENTO**

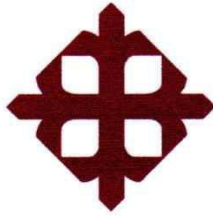
En primer lugar, agradecerle a Dios por otorgarme salud y bendiciones para culminar mis estudios, a pesar de muchas veces cuestionar mi fe, estuvo presente en todo momento.

Le agradezco a mis padres y al resto de mi familia por su apoyo incondicional a pesar de la distancia que solemos llevar.



## **DEDICATORIA**

Deseo dedicar este trabajo a mis queridos padres y demás familiares quienes me dieron su apoyo incondicional en todo momento, a mi cardiólogo Michael Chasin que con su tratamiento pude recuperarme de las extrasístoles cardíaca por la que estaba pasando mientras realizaba esta investigación.



UNIVERSIDAD CATÓLICA  
DE SANTIAGO DE GUAYAQUIL  
FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO  
INGENIERÍA ELECTRÓNICA EN CONTROL Y AUTOMATISMO

**TRIBUNAL DE SUSTENTACIÓN**

f.

  
\_\_\_\_\_

**JOHN E. FRANCO RODRÍGUEZ, PH.D.**

DECANO

f.

  
\_\_\_\_\_

**M. Sc. BOHÓRQUEZ ESCOBAR, CELSO BAYARDO**

DIRECTOR DE CARRERA

f.

  
\_\_\_\_\_

**ING. ROMERO ROSERO CARLOS**

OPONENTE



## ÍNDICE GENERAL

CAPÍTULO 1: ASPECTOS GENERALES .....	2
1.1. Introducción .....	2
1.2. Definición del problema .....	2
1.3. Justificación del problema .....	3
1.4. Objetivos .....	4
1.4.1. Objetivo General .....	4
1.4.2. Objetivos específicos .....	4
1.5. Hipótesis.....	4
1.6. Metodología de investigación .....	5
CAPÍTULO 2: MARCO TEÓRICO .....	6
2.1. Automatización .....	6
2.2. Controlador Lógico Programable (PLC) .....	8
2.2.1. Arquitectura y configuración de PLC .....	8
2.3. Redes de comunicación industrial.....	10
2.3.1. Nivel de campo.....	12
2.3.1. Tipos de protocolo en comunicaciones industriales .....	14
2.3.1.1. Modbus RTU .....	15
2.3.1.2. Ethernet/IP .....	16
2.3.1.3. Ethernet TCP/IP .....	18
2.3.1.4. Profinet.....	20
2.3.1.5. Profibus.....	21
2.4. Softwares de programación.....	22
2.4.1. Simatic Step7 .....	22
2.4.2. Totally Integrated Automation Portal (TIA Portal) .....	24
2.4.3. Factory IO .....	30
2.5. Leguajes de programación .....	35
2.5.1. PYTHON .....	36
2.5.2. C++ .....	39
2.5.5. Diagrama Ladder.....	41
2.5.6. Esquema de contactos .....	43
2.7. Sensores.....	45
2.7.1. Sensores de proximidad.....	46
2.8. Actuadores .....	47
2.8.1. Actuador neumático .....	47

2.8.2. Actuador hidráulico .....	49
CAPÍTULO 3: CONSTRUCCIÓN Y SIMULACIÓN .....	50
3.1. Conexión de TIA Portal con FactoryIO.....	50
3.2. Diseño gráfico de clasificadora de paquetes.....	55
3.3. Diseño de programación en TIA Portal .....	58
3.3.1. Tabla de Entradas.....	58
3.3.2. Tabla de Salidas .....	59
3.3.3. Tabla de Estados .....	60
3.3.4. Tabla de Marcas .....	60
3.3.5. Bloques de programación.....	61
CAPÍTULO 4: CONCLUSIONES Y RECOMENDACIONES .....	70
4.1. Conclusiones.....	70
4.2. Recomendaciones.....	71
Fuentes Consultadas .....	72
Anexos.....	77
Anexo 1.....	77
Anexo 2.....	78

## ÍNDICE DE FIGURAS

<b>Figura 1:</b> Estructura de un sistema automatizado.....	7
<b>definido.</b>	
<b>Figura 2:</b> Arquitectura de los PLC .....	8
<b>Figura 3:</b> Niveles de comunicación de un sistema de automatización.....	11
<b>Figura 4:</b> Arquitectura de una Red de Comunicación Industrial.....	14
<b>Figura 5:</b> Entorno virtual de FactoryIO .....	30
<b>Figura 6:</b> Ejemplos de programación Python .....	36
<b>Figura 7:</b> Símbolos básicos del lenguaje Ladder.....	42
<b>Figura 8:</b> Implementación de funciones básicas: AND (izquierda) y OR (derecha).....	43
<b>Figura 9:</b> Elementos básicos del esquema de contactos .....	44
<b>Figura 10:</b> Funciones lógicas con diagrama de contactos: AND, OR, XOR .....	44
<b>Figura 11:</b> Sensores de proximidad Pepperl+Fuchs.....	46
<b>Figura 12:</b> Actuador neumático marca Cy. Pag.....	48
<b>Figura 13:</b> Actuador hidráulico marca Eckart .....	49
<b>Figura 14:</b> Interfaz inicial TIA Portal V16 .....	50
<b>Figura 15:</b> Lista de controladores disponibles en TIA Portal V16.....	51
<b>Figura 16:</b> Interfaz inicial FactoryIO.....	52
<b>Figura 17:</b> Árbol del proyecto en TIA Portal V16 .....	53
<b>Figura 18:</b> Bloque de función de programa con C++ .....	54
<b>Figura 19:</b> Lista de Drivers compatibles en FactoryIO.....	54
<b>Figura 20:</b> Variables expuestas por Siemens S7-PLCSIM .....	55
<b>Figura 21:</b> Elementos disponibles para diseñar en FactoryIO .....	55
<b>Figura 22:</b> Inicio de diseño de clasificadora con variables.....	56
<b>Figura 23:</b> Diseño de clasificadora en FactoryIO finalizado .....	57
<b>Figura 24:</b> Variables declaradas con la clasificadora diseñada .....	57
<b>Figura 25:</b> Bloque de organización StartUp.....	61
<b>Figura 26:</b> Segmentos de programación en Bloque Inicial .....	62
<b>Figura 27:</b> Segmento 1 del bloque de transiciones.....	63
<b>Figura 28:</b> Segmento 2 del bloque de transiciones.....	64
<b>Figura 29:</b> Segmento 1 del bloque de acciones.....	65
<b>Figura 30:</b> Segmento 2 del bloque de acciones.....	66
<b>Figura 31:</b> Segmento 1 del bloque piloto .....	67
<b>Figura 32:</b> Segmento 1 del bloque contadores .....	68
<b>Figura 33:</b> Bloque organizador Main.....	69

## ÍNDICE DE TABLAS

<b>Tabla 1:</b> Resultados de implementación de compuerta AND en Esquema de contactos .....	44
<b>Tabla 2:</b> Resultados de implementación de compuerta OR en Esquema de contactos .....	45
<b>Tabla 3:</b> Entradas declaradas en TIA Portal .....	58
<b>Tabla 4:</b> Salidas declaradas en TIA Portal .....	59
<b>Tabla 5:</b> Estados declarados en TIA Portal .....	60
<b>Tabla 6:</b> Marcas declaradas en TIA Portal .....	60

## RESUMEN

En el presente documento se realiza la simulación de una línea clasificadora de paquetes sin la utilización excesiva de recursos, se propone integrarlas en agencias sucursales de Courier provinciales, se implementará dos softwares, uno de programación y simulación de PLC virtual llamado TIA Portal de Siemens, el segundo software es dedicado a la simulación gráfica y de compatibilidad con PLC's virtuales, es llamado Factory I/O. Estos programas simulan la comunicación con óptimos protocolos con la finalidad de reducir tiempos y agilizar el proceso, todo será controlado por un tablero virtual de operador graficado en el software Factory I/O, este va a tener los controles suficientes e información necesaria para el operador. Al tener una línea de clasificación menos manual y sistemática, se podrá reducir los tiempos de entrega dentro de las sucursales de empresas Courier.

**Palabras claves:** CLASIFICADORA, SOFTWARE DE PROGRAMACIÓN, PLC, TIA PORTAL, SIEMENS FACTORY I/O.

## **Abstract**

In this document the design and modeling of a package sorting line without excessive use of resources, it is proposed to integrate them in Courier branch agencies, two software will be implemented, one for programming and simulation of virtual PLC called TIA Portal from Siemens, the second software is dedicated to graphical simulation and compatibility with virtual PLC's, it is called Factory I/O. These programs simulate communication with best protocols to reduce time and streamline the process, everything will be controlled by a virtual operator panel graphed in the Factory I/O software, this will have sufficient controls and information necessary for the operator. By having a less manual and systematic sorting line, it will be possible to reduce delivery times within the branches of Courier companies.

**Keywords:** CLASSIFIER, PROGRAMMING SOFTWARE, PLC, TIA PORTAL, SIEMENS FACTORY I/



## **CAPÍTULO 1: ASPECTOS GENERALES**

### **1.1. Introducción**

Las grandes empresas Courier como Servicios Postales del Ecuador, el nuevo servicio que pretendía mejorar los servicios de paquetería en Ecuador consta con clasificadoras en sus matrices, pero al momento de repartir a diferentes sucursales no cuentan con estas, aquí es cuando una persona o dos son encargadas de registrar manualmente el código de barras, y apilar los paquetes. (Celi, 2021)

El correcto funcionamiento de una banda transportadora y su operador en una determinada empresa de mensajería o Courier evitan ocasionar congestión, confusión, mal desempeño de funciones, controles apresurados, caída de mercancías, entre otros.

### **1.2. Definición del problema**

Las empresas de Courier deben cumplir estrictamente con su itinerario de entrega, en este proceso se tienen errores comunes y repetitivos, que conllevan a la demora de la entrega de paquetes, debido a la inexistencia de un sistema automático que permita la clasificación correcta de la mercadería antes de ser entregada.

Las empresas Couriers de la provincia de Santa Elena, presentan estas falencias sumadas a la no presencia de las mismas en cada rincón de la provincia y la urgencia de retirar los paquetes, conlleva a un caos administrativo y logístico de la sucursal, debido a esto el encargado de la entrega de paquetes

tarda entre 10 a 15 minutos por persona, lo que conlleva a un retraso catastrófico en las entregas

### **1.3. Justificación del problema**

El proceso para solucionar la tardanza en la entrega de la mercancía de las empresas Courier, es la utilización de bandas transportadoras automatizadas, los cuáles minimizan riesgos de trabajo y procesan la siguiente información:

- Por lugar de entrega
- Tamaño del paquete
- Tipo de mercadería
- Conteo de bultos

La utilización de bandas clasificadoras reducirá los tiempos de entrega de paquetes, debido a que contienen sistemas PLC's que controlan actuadores y motores, acompañados de software Tia Portal de Siemens y Factory IO, para evitar el retraso y pérdidas económicas en empresas, microempresas o emprendimientos de fabricación. Los avances tecnológicos son un reto para las empresas en relación con su automatización, este progreso impulsa la mejora de sus procesos con la finalidad de adaptarse incluyendo actualizaciones a corto o largo plazo, de acuerdo con este análisis se reconoce que es preciso optimizar los tiempos de entrega de paquetes en empresas Courier de la provincia Santa Elena.

La sistematización de la clasificación de paquetes dará resultados instantáneos, modificando el plan de tiempos de espera de acuerdo con las

necesidades del cliente destinatario, manteniendo el servicio al cliente con excelencia ya que no existiría el típico cuello de botella al momento de despachos, entregando así el producto en óptimas condiciones y en un tiempo de espera adecuado.

## **1.4. Objetivos**

### **1.4.1. Objetivo General**

Controlar una simulación de banda clasificadora para una empresa Courier utilizando un PLC virtual y la conexión entre los softwares Tia Portal y Factory IO, para evitar pérdidas económicas y atrasos en su clientela.

### **1.4.2. Objetivos específicos**

- Conocer a detalle los procesos operativos actuales de empresa Courier interprovincial.
- Compilar PLC virtual en TIA Portal de Siemens para hacer la conexión con el programa simulador gráfico Factory I/O.
- Evaluar el beneficio de la implementación de una banda clasificadora en empresa Courier

## **1.5. Hipótesis**

Mediante la simulación propuesta se pretende dar una mejora eficaz al proceso de clasificación de paquetes debido a que permitirá tener el control total de bultos en las sucursales Courier interprovinciales y evitar las largas esperas de los clientes destinatarios en las oficinas, reduciendo así en un 80% el tiempo total que tarda el despacho de un paquete.

## **1.6. Metodología de investigación**

Este proyecto trabaja con tres metodologías de investigación: descriptiva, documental y científica. La investigación descriptiva analiza las características de una población o fenómeno sin conocer las relaciones entre ellos. (Binda, 2013)

La investigación documental, como su nombre muestra, se realiza a partir de la información contenida en todo tipo de documentos, como fuentes bibliográficas, hemerográficas o de archivo. (Guevara, 2020)

La investigación científica es un conjunto de etapas regladas, que contienen los principios generales de las actividades científicas y son consideradas como los elementos principales de la actividad universitaria, pues gracias a ellas se lleva a cabo la relación entre las funciones docente y observación. (EnagoAcademy, 2021)

## **CAPÍTULO 2: MARCO TEÓRICO**

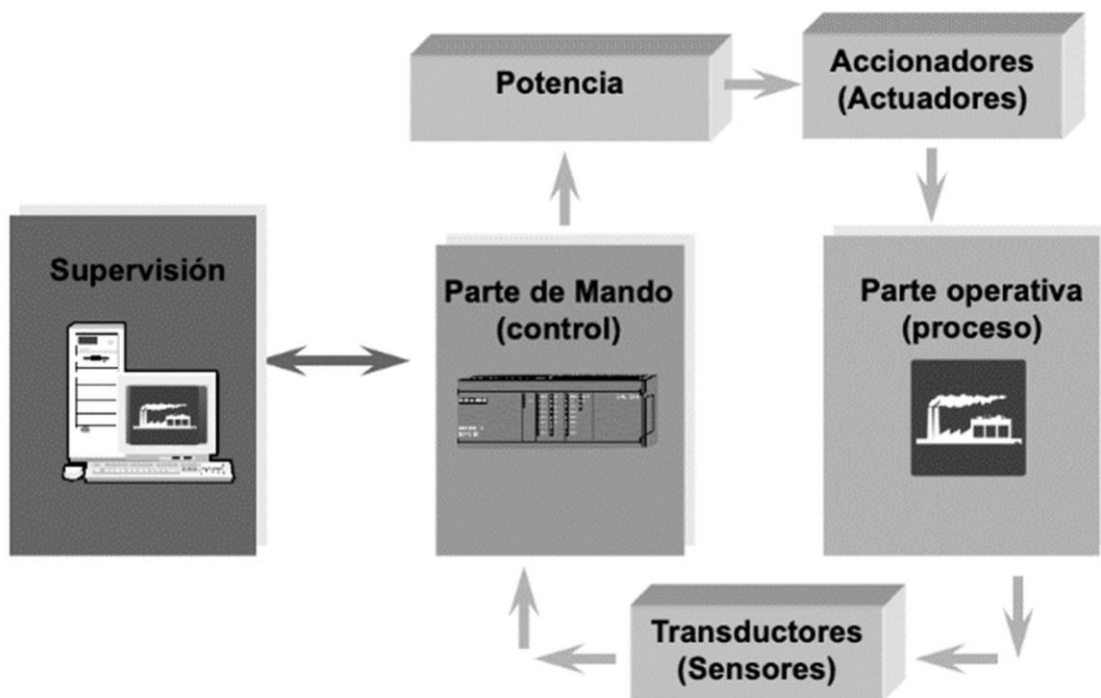
### **2.1. Automatización**

A lo largo de la historia, las personas y culturas del mundo se han enfrentado al dilema: cómo hacer más con menos esfuerzo, con el advenimiento de la revolución industrial en el siglo XVIII, se introdujo el concepto de automatización. La automatización ha recorrido un largo camino desde las primeras líneas de montaje móviles y la desmotadora de Henry Ford. De hecho, con la tecnología avanzada actual, aproximadamente el 75 % de las actividades laborales pueden automatizarse, lo que equivale a alrededor de 2 billones de dólares en salarios solo en los Estados Unidos. Pero más allá de ser una medida de reducción de costos, la automatización mejora la experiencia de los empleados al hacerse cargo de las tareas rutinarias, brindándole más tiempo y recursos para concentrarse en proyectos interesantes y creativos. (ServiceNow, 2020)

Es una de las áreas del progreso científico y tecnológico que utiliza medios técnicos y métodos matemáticos de autorregulación con el fin de liberar a las personas para que participen en los procesos de adquisición, transformación, entrega y uso de energía, materiales, productos o información, o reducir significativamente este nivel de implicación o complejidad de las operaciones ejecutadas. Se asocia a la eliminación o reducción de la participación humana en los diversos procesos productivos, teniendo en cuenta la aplicación de sistemas mecánicos, electrónicos y para operar y controlar la producción con alta eficiencia y eficacia. Además, este concepto significa la integración, a efectos de una amplia gama de información avanzada y descubrimientos de ingeniería de última generación en los procesos de

producción. En este sentido, la automatización está compuesta por una parte operativa de realizar las diferentes actividades que forman parte de un proceso a través de los diferentes elementos que lo componen; y una parte de control que es responsable de coordinar las actividades del proceso, incluidas las operaciones de calidad, gestión de herramientas y supervisión. (LogicBus, 2020)

La automatización de la infraestructura (TI) es el uso de un sistema de software para crear instrucciones y procesos repetibles que reemplazan o reducen la interacción humana con un sistema informático. El software de automatización funciona dentro de los límites de estas instrucciones, herramientas y marcos para completar la tarea sin o sin intervención humana. Este proceso es fundamental tanto para la optimización de TI como para la transformación digital. (RedHat, 2018)



**Figura 1:** Estructura de un sistema automatizado  
Fuente: (Lopez, 2017)

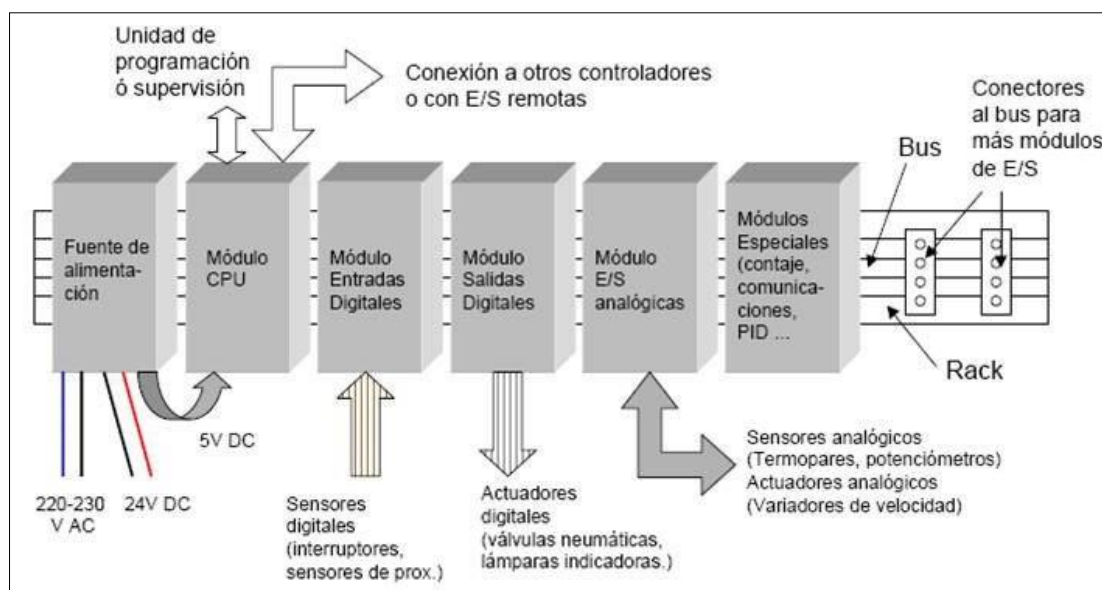


## 2.2. Controlador Lógico Programable (PLC)

Es un dispositivo electrónico utilizado en la automatización industrial. Los PLC controlan la lógica de operación de máquinas, plantas y procesos industriales, procesan y reciben señales digitales y analógicas, y pueden aplicar estrategias de control. Controlador Lógico Programable o Controlador Lógico Programable. Es un dispositivo electrónico que, como su nombre lo indica, está diseñado para programar y controlar procesos secuenciales en tiempo real. Generalmente, este tipo de equipos se pueden encontrar en entornos industriales. (LogicBus, 2020)

### 2.2.1. Arquitectura y configuración de PLC

Se describe así a los componentes o partes de los equipos y los circuitos electrónicos que los componen de manera continua. El sistema operativo para cada uno de estos diagramas también se enumera allí. La estructura de un controlador lógico programable (PLC) consta de 5 componentes o elementos incluidos en su composición: memoria, procesador, módulo de entrada, módulo de salida y fuente de alimentación. (LogicBus, 2020)



**Figura 2:** Arquitectura de los PLC  
Fuente: (Rojas, 2018)

#### **2.2.1.1. Memoria**

Las instrucciones y los datos se almacenan o envían aquí. La memoria del procesador lógico programable almacena programas de usuario, toda la información de entrada y salida, datos temporales y una gran cantidad de constantes de control de programa. (ServiceNow, 2020)

#### **2.2.1.2. Procesador**

Es la parte principal de la arquitectura de automatización, responsable del procesamiento continuo de los datos recibidos con fines industriales. Tome decisiones lógicas y coordina diversas tareas, incluidas operaciones aritméticas y lógicas. (Ripipsa, 2021)

#### **2.2.1.3. Módulo de entrada**

Suministra de forma eficaz la conexión a una variedad de disparadores y señales específicos de la aplicación en distintas áreas industriales. Estas entradas están conectadas a un interruptor que será activado automáticamente por el dispositivo o el operador. (Lopez, 2017)

#### **2.2.1.4. Módulos de Salida**

Modulan las señales de salida, que pueden ser señales digitales o analógicas. El PLC controla o emite comandos de salida usando voltaje al equipo usado en el circuito. La salida, como la entrada, muestra dos tipos de valores de voltaje o capacitancia; Pueden estar ausentes (inactivos) o presentes (activos). (ServiceNow, 2020)

#### **2.2.1.5. Fuente de Alimentación**

Suministra o genera toda la energía que se necesita para que el CPU y otras tarjetas en un momento en que se formó o programó el PLC. La fuente de sustento actúa como una unidad de control. (IndustriasGSL, 2019)

En la estructura del PLC encontramos los siguientes componentes externos:

#### **2.2.1.6. Terminal programable**

Es un elemento auxiliar útil en el funcionamiento u operación del PLC. Con este dispositivo se puede acceder a las instrucciones que componen el plan de usuario para la ejecución de actividades de inspección en ambientes industriales. Muchos PLC tienen un programador o dispositivo similar a una computadora en el teclado con todos los símbolos precisos para establecer un programa o proyecto de control. Además, muchas cuentan con una diminuta pantalla LCD en la que se ejemplifica gráficamente el retrato de la tecla presionada. (Ripipsa, 2021)

#### **2.2.1.7. Periféricos**

Son interfaces HMI, enlace a una diversidad de impresoras, esclusas de comunicación, etc. La supermemoria del PLC acopia dos tipos de datos: data del procesamiento y datos de control, industria, función y diseño de Controlador Lógico Programable. (IndustriasGSL, 2019)

### **2.3. Redes de comunicación industrial**

En automatización, entre los dispositivos que interfieren en la gestión de estos sistemas, son el principal factor que consiente no solo el funcionamiento

estándar, sino también el control e intervención de estos procesos. Las arquitecturas de sistemas de automatización industrial deben basarse en redes de comunicación confiables para permitir una mayor transformación digital de los procesos de fabricación. Son necesarios para el intercambio de datos, el control de datos y la flexibilidad para combinar dispositivos o máquinas de diferentes fabricantes en una sola instalación. (Rojas, 2018)

La comunicación entre diversos dispositivos se ejecuta mediante varios protocolos de comunicación técnica. Un protocolo de comunicación es un ligado de reglas que permiten la transmisión y el intercambio de datos entre dispositivos que cooperan. En esta temprana industrialización de la tecnología, la antigua automatización se limitaba al control local de máquinas o líneas de producción específicas, por lo que los sistemas automatizados estaban aislados entre sí y se intercambiaba información para facilitar la producción a gran escala. La medida aumenta la disponibilidad de la información de los dispositivos de campo centralizados a nivel de planta, lo que consiente una gestión eficiente de los procesos de producción. La integración de estos sistemas generalmente se logra mediante la distribución de comunicaciones a través de múltiples capas, como buses de campo, LAN y WAN. (Gil, 2021)



**Figura 3:** Niveles de comunicación de un sistema de automatización  
Fuente: (Gil, 2021)

### **2.3.1. Nivel de campo**

Contiene dispositivos distribuidos como sensores (entradas) y actuadores (salidas), módulos de E/S, unidades de accionamiento de las máquinas/plantas del fabricante y sistemas de comunicación en tiempo real. Los datos se transmiten periódicamente y se determinan por ciclos de bus cortos. La permanencia real de un período de autobús es dependiente a su estudio. Puede ser alterado desde un lapso de décimas de microsegundo hasta cientos de milisegundos. Aquí es cuando solo se envían unos escasos bytes para intervenir en actuadores, receptor señales de sensores, entre otros. Al mismo tiempo, sin embargo, las demandas sobre la demora máxima también son altas. Para controlar correctamente el proceso, la señal de control debe enviarse en milisegundos. El Profibus DP y Profinet IO cumplen con estos criterios y ofrecen una solución común para la automatización de fábricas y procesos. (Hurtado, 2017)

### **2.3.2. Nivel de control**

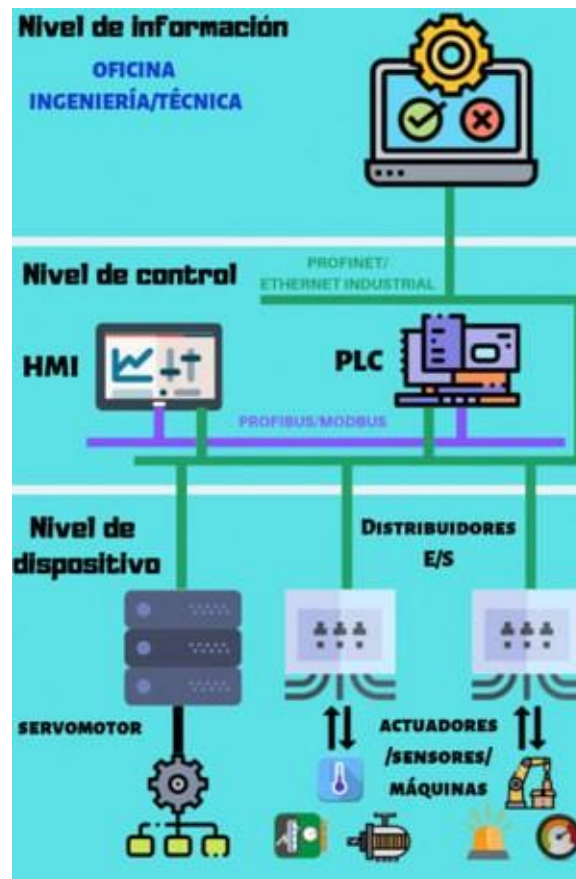
Los controladores lógicos programables (PLC) se participan entre sí con los sistemas informáticos técnicos o de ingeniería a través de patrones estándares como intranets, Internet y Ethernet TCP/IP. La gran mayoría de métodos informáticos computados como funciones que inspeccionan el proceso están dentro del nivel de control. El controlador está conectado a sensores/actuadores de nivel de campo, cada uno de los cuales controla una parte del sistema. Los controles también están interconectados y conectados a niveles superiores. A este nivel se envía información que varía en tamaño desde unos pocos bytes hasta varios kilobytes. La latencia es una fracción de segundo.

Este flujo de información requiere paquetes de datos y una serie de protocolos de comunicación. Al igual que Profibus, Profinet basado en Ethernet ofrece una solución para este propósito que es utilizada por las empresas. (Gil, 2021)

### **2.3.3. Nivel de información**

El controlador a nivel de fábrica recopila información de gestión a nivel de planta y administra todo el sistema de automatización a través de los sistemas MES y ERP. El sistema MES está directamente vinculado al nivel actual de control y datos de producción. Finalmente, rastrea la información de pedidos y productos en la lista, recopila e informa transacciones a los sistemas financieros y de planificación, luego transmite órdenes de compra y órdenes de ensamblaje a los empleados de las talleres eléctricos. ERP, por otro lado, consiente la recopilación y organización de data comercial a través de un conjunto de softwares integrados. El software ERP contiene aplicaciones que computan funciones comerciales como elaboración, cotización de ventas y contabilidad. Este nivel tiene una gran red. WAN Ethernet para intercambiar información de gestión y planificación de fábrica. Los datos se transfieren de unos pocos megabytes a gigabytes. La latencia es de segundos. (Valencia, 2019)





**Figura 4:** Arquitectura de una Red de Comunicación Industrial  
Fuente: (Gandhi, 2019)

### 2.3.1. Tipos de protocolo en comunicaciones industriales

Es un conjunto de reglas para la comunicación entre dispositivos en una red. Algunos de los protocolos más utilizados en la industria incluyen: Modbus RTU, Ethernet TCP/IP, Ethernet/IP, Modbus TCP/IP, Profinet, Profinet. Funcionan con redes RS- 85. Este protocolo serial y otros protocolos seriales comunes son compatibles con una variedad de proveedores y son bien conocidos por un amplio grupo de profesionales de la automatización. Sin embargo, debido a su rendimiento limitado, el protocolo serie no es adecuado para aplicaciones de alta velocidad u otras aplicaciones exigentes. Ethernet se ha transformado en el estándar de capa física guía para variados protocolos de plantas como Ethernet/IP, Modbus TCP/IP, Profinet y Ethernet TCP/IP, debido

a su beneficio y otras superioridades. Con Ethernet, la conexión de varios dispositivos, como PLC, HMI, E/S de campo, bancos de válvulas, entre otras, no es tan difícil. Ethernet también es rápido en comparación con las redes seriales antiguas, por lo que la comunicación sigue siendo rápida incluso cuando se comunica con varios dispositivos diferentes a través del mismo cable. (IndustriasGSL, 2019)

### **2.3.1.1. Modbus RTU**

Es un protocolo abierto serie (RS-232 o RS-485) procedente de la arquitectura Maestro/Esclavo, es largamente admitido por su disposición de uso y seguridad, se utiliza ampliamente en métodos de cometido de edificios y métodos de sistematización industrial. Para montar la unidad de datos de la adaptación MODBUS, el cliente debe instruir una arreglo MODBUS, es la función que comunica al servidor acerca del arquetipo de trabajo a realizar, la forma de una solicitud partidaria por un Maestro es determinado por el protocolo de estudio MODBUS, el campo de cifra de cargo se recopila en un byte, solo las cifras dentro de la categoría de 1 a 255 se estiman válidos, con 128-255 prudentes para réplicas de excepción, cuando el Maestro remite un mensaje al Esclavo, es el campo de cifra de función el que comunica al servidor sobre qué tipo de labor realizar. Para precisar variadas labores, algunas funciones apalearán códigos de subfunción suplementarios, por ejemplo, el Maestro puede examinar los estados de ENCENDIDO/APAGADO de un conjunto de salidas o entradas cautas, además podría leer/escribir el adjunto de los datos de un conjunto de registros MODBUS, cuando el Maestro recoge la respuesta del Esclavo, el campo del código de ocupación es usado por el Esclavo para mostrar

una réplica sin errores o una contestación de excepción, el esclavo se crea eco de la deferencia del código de función originaria en el caso de una réplica normal. (LogicBus, 2019)

### **2.3.1.2. Ethernet/IP**

Fue desarrollado por el Protocolo Industrial Común (CIPTM), un conjunto de estándares abiertos que todas las empresas de automatización deben mantener. ODVA es la organización que apoya CIP a través de su protocolo industrial. En general, Ethernet/IP es simplemente una implementación de un protocolo industrial común. Está esbozado para comunicarse a través de Ethernet default que se utiliza en nuestras redes domésticas y comerciales. El "IP" en Ethernet/IP significa Protocolo Industrial, a diferencia del lenguaje de comunicación de Internet de la computadora conocido como TCP/IP. Desde la introducción de la comunicación Ethernet, la pregunta ha sido cómo se puede utilizar este estándar de red Ethernet en la automatización industrial compleja y de alta velocidad. Uno de los temas de discusión en automatización es cómo mantener o aumentar el llamado determinismo. El determinismo es la capacidad de determinar o garantizar que un mensaje llegará en el momento deseado. Aunque otros protocolos de comunicación son más deterministas que Ethernet/IP o Profinet (como DeviceNet o Profibus), los avances en la tecnología de protocolos y el aumento de la velocidad han llevado a su aceptación como un estándar industrial útil. (Dynapar, 2019)

Los codificadores Ethernet/IP admiten una amplia gama de topologías de red, incluidas redes lineales, en estrella y en anillo, y están diseñados para integrar funciones para optimizar la transmisión continua de datos para cualquier

topología de red. Algunas de las funciones opcionales que se pueden agregar a los dispositivos Ethernet/IP, como variadores, PLC y sensores, incluyen: CIP Sync, CIP Motion, CIP Protection, IP Address Collision Detection y Device Level Loopback (DLR). CIP Sync es un complemento para la sincronización precisa del reloj entre los componentes de la red cuando la precisión en tiempo real es fundamental, como la coordinación de varios ejes. CIP Motion simplemente elimina el requisito de determinismo estricto y cambia la ejecución de comandos/acciones a una ejecución secuencial acción por acción en un intervalo de tiempo predeterminado (por ejemplo, tiempo X, haga esto). El CIP Safety proporciona transmisión de datos segura en nodos hasta el nivel SIL 3. La detección de colisión de direcciones IP está diseñada para informar al usuario cuando dos dispositivos en la red tienen asignada (o tienen) la misma dirección IP. El anillo de nivel de dispositivo (DLR) es donde el codificador puede detectar una interrupción en el enlace de comunicación principal y comunicar la información de interrupción al controlador. Esta interrupción le indica al controlador que envíe datos en ambas direcciones (no solo en una dirección en condiciones normales de funcionamiento) para garantizar un funcionamiento fluido y sin demoras. Actualmente, el codificador Ethernet/IP de Dynapar solo admite la detección de colisiones de direcciones IP. La funcionalidad DLR se está desarrollando para futuras integraciones. Rockwell es el único proveedor de codificadores con sincronización CIP y movimiento CIP integrados en el codificador. (Dynapar, 2019)

### **2.3.1.3. Ethernet TCP/IP**

Es la definición de un grupo de protocolos de red que permiten la transferencia de datos a través de redes entre dispositivos informáticos e Internet. El acrónimo TCP/IP hace referencia a este grupo de protocolos. TCP es un protocolo de control de transmisión que permite que dos hosts establezcan una conexión e intercambien datos. Este protocolo garantiza una transferencia de datos fiable. IP o Protocolo de Internet utilizan una dirección serial de cuatro octetos formateada como un punto decimal (por ejemplo, 75. .160.25). Este protocolo transfiere datos a otras computadoras en la red. El modelo TCP/IP permite un intercambio de datos confiable dentro de una red mediante la definición de procedimientos para enviar en paquetes y recibir datos. Para conseguirlo utiliza un sistema de capas con capas una construida a partir de la capa anterior que se comunican únicamente con la capa superior donde se envían los resultados y la capa inferior donde se solicitan los servicios. (Robledano, 2019)

El modelo TCP/IP tiene cuatro niveles o capas que se deben considerar. Capa de enlace o acceso a la red; esta es la primera capa del modelo, que brinda la capacidad de acceder físicamente a una red (anillo, ethernet, etc.) y especifica cómo enrutar los datos independientemente del tipo. de la red a utilizar. La red o capa de Internet proporciona paquetes de datos o datagramas y administra las direcciones IP. (Un datagrama es un paquete de datos que representa la menor cantidad de información en una red). Esta capa se considera la más importante. En la capa de transporte se conoce el estado de transferencia y los datos de enrutamiento, utiliza puertos para asignar tipos de aplicaciones a tipos de datos. Capa de aplicación: la parte superior del protocolo TCP/IP, que proporciona

aplicaciones de red que se comunican con la capa anterior (usando los protocolos TCP o UDP). Finalmente las capas del modelo TCP/IP corresponden a varias capas del modelo OSI teórico, pero con tareas más diversas. El protocolo TCP/IP es muy importante, esto se debe a que los datos enviados pueden llegar a su destino sin errores por la misma ruta por la que fueron enviados. (Robledano, 2019)

TCP/IP tiene un gran impacto en la precisión e implica varios pasos para garantizar una transmisión de datos adecuada entre dos computadoras. Este es uno de los mecanismos utilizados para tal fin. Si el sistema envió el mensaje completo y encontró algún problema, se debe volver a enviar el mensaje completo. Divide cada mensaje en paquetes que se vuelven a ensamblar en el otro extremo. De hecho, cada paquete puede tomar una ruta diferente a la computadora de destino si esa ruta no está disponible o está muy congestionada. Además, divide varias tareas de comunicación en capas. Cada clase tiene su propia función. Los datos pasan a través de cuatro capas separadas antes de recibirse en el otro extremo (como se explica en la siguiente sección). Luego, TCP/IP atraviesa estas capas en orden inverso para recopilar los datos y presentarlos al receptor. El propósito de las capas es crear un sistema estandarizado en el que los diferentes proveedores de hardware y software no tengan que administrar las comunicaciones por sí mismos. Es como conducir un coche: todos los fabricantes están de acuerdo en la posición de los pedales, por lo que no tenemos que tenerlo en cuenta a la hora de cambiar de coche. Esto también significa que algunos niveles se pueden actualizar, por ejemplo, para mejorar el rendimiento o la seguridad, sin tener que actualizar todo el sistema. (Cilsa, 2019)



#### **2.3.1.4. Profinet**

Es uno de los estándares de comunicación más utilizados en las redes de automatización. El Profinet se basa en Industrial Ethernet, TCP/IP y algunos estándares de comunicación de la industria de TI. Entre sus características destaca Ethernet en tiempo real, donde los dispositivos que se comunican a través del bus de campo acuerdan cooperar en el procesamiento de las solicitudes realizadas dentro del bus. Basado en conexiones básicas como el cable Ethernet y varios marcos de comunicación establecidos correspondientes a los niveles 1 y 2, a través de la interpretación, PROFINET se puede utilizar como "perfiles" de servicios públicos. Gradualmente integre nuevas funciones a medida que Volverse reconocido. Especificado específicamente para el caso de datos salientes, nivel de modificación 7 (aplicación). Con Profisafe, los datos de seguridad (safety) se envían junto con los datos y comandos de ProfiEnergy para el ahorro y control de energía. PROFINET permite conectar dispositivos, sistemas y células (grupos de dispositivos aislados entre sí), aumentando tanto la velocidad como la seguridad de la comunicación, reduciendo costes y optimizando la producción. PROFINET, por sus propiedades, permite la compatibilidad con la comunicación Ethernet más típica en el entorno de TI, todo menos la diferencia de velocidad que tiene la comunicación Ethernet en las redes empresariales en comparación con el rendimiento en tiempo real de las redes industriales. requisito. Además, el uso del estándar PROFINET a nivel de E/S ofrece las siguientes ventajas: Mejora la escalabilidad de la infraestructura. Acceso a dispositivos de campo a través de la red . PROFINET, un protocolo que utiliza Ethernet para la comunicación permite un fácil acceso a los dispositivos de campo desde otras redes. Realiza tareas de mantenimiento y

presta servicios desde cualquier lugar. Se accede a los dispositivos de campo a través de conexiones seguras como VPN para mantenimiento remoto. (INCIBE, 2018)

#### **2.3.1.5. Profibus**

Es un estándar de red digital abierto que maneja la comunicación entre sensores de campo y sistemas de control o controladores. Es importante saber que PROFIBUS significa Process Field Bus. La primera idea en desarrollo del fue implementar soluciones PROFIBUS en la industria de automatización de fábricas, luego en la industria de procesos, fabricación, etc. Esta red de comunicación es adecuada para aplicaciones sensibles de alta velocidad y tareas de comunicación complejas. usuarios de todo el mundo pueden trabajar en el desarrollo de un protocolo estándar internacional destinado a la reducción de costes, la flexibilidad, la fiabilidad, la orientación al futuro, la adaptación a diferentes aplicaciones, la Inter funcionalidad y la adquisición de múltiples proveedores. Por el lado del desarrollo, la tecnología de bus de campo de procesos es estable y está en constante evolución. Las empresas miembros de PROFIBUS International se reúnen constantemente en grupos de trabajo que se enfocan en los nuevos requisitos del mercado y aseguran nuevas ventajas a medida que surgen nuevas características. La tecnología de la información ha jugado un papel importante en el desarrollo de la automatización industrial, cambiando la jerarquía y la estructura de las empresas. Hoy llega a entornos industriales y sus diversas áreas, desde industrias de proceso y fabricación hasta edificios y sistemas logísticos. La posibilidad de comunicación entre dispositivos y el uso de mecanismos estandarizados, abiertos y flexibles son una parte integral de los conceptos de automatización actuales. La comunicación se

propaga rápidamente tanto horizontal como verticalmente a nivel de campo, integrando todos los niveles jerárquicos del sistema de automatización. (Aula21, 2018)

## **2.4. Softwares de programación**

Son sistemas de control de computadoras industriales que monitorean continuamente el estado de los dispositivos de entrada y toman decisiones basadas en programas definidos por el usuario para controlar el estado de los dispositivos de salida. (Gútiez, 2019)

### **2.4.1. Simatic Step7**

En este software se consigue establecer los contextos con las que desclavará el autómata, alternará data y cruces del proceso a ejecutar para consecutivamente dar unas disposiciones de actuación sobre los diversos actuadores y elementos de la instalación. A grandes semblantes la memoria del autómata es fraccionada en las siguientes partes: Imagen de las entradas y salidas (E/S), E/S de la periferia, marcas, valor y estado de temporizadores y contadores, módulos de datos, datos temporales que se pierden al final de cada ciclo. (Gútiez, 2019)

#### **2.4.1.1. Imagen de Entradas y Salidas (E/S)**

Son las señales utilizadas en el programa. Son solo eso, imagen, debido al valor instantáneo de la entrada ni forzamos la salida inmediatamente, cuando la imagen la imagen de entrada se ha refrescado antes iniciar un nuevo ciclo

OB1 (programa principal). De manera similar, en el caso de la salida, cuando el programa establece la salida en uno, no lo hace de inmediato, sino que le da un valor a la imagen de la salida y la misma se representará en la salida real antes de comenzar un nuevo ciclo OB1. (Gil, 2021).

#### **2.4.1.2. E/S periférica**

Este es el importe actualizado de algún sensor analizado claramente. Para ello se pueden leer en bytes (PEB), palabras (PEW) o palabras dobles (PED), ya que a un determinado bit no se puede acceder directamente, lo mismo ocurre con la salida, esto obliga a fijar el estado de la salida. Inmediatamente, acceso a la salida de periféricos. (IndustriasGSL, 2019)

#### **2.4.1.3. Marcas**

Son indexadas y variables globales, por lo que no es necesario declararlas, pero si deben tener un nombre y un tipo (entero, doble o real). Son asignadas con un valor en cualquier parte del programa. Dependiendo de la CPU. El valor se puede guardar después de apagar el PLC o puede fluctuar (se pierde después de apagar el PLC). Su acceso puede variar desde un bit hasta un ancho indefinido si se usa un puntero de área. (Gútiez, 2019)

#### **2.4.1.4. Temporizadores y contadores**

La programación en Step7 presenta dos tipos de temporizadores: por software (IEC) y por temporizadores S7. Estos son temporizadores de hardware y están limitados en función de la CPU, pero dado que son generados por software, el límite es la memoria disponible en sí, pero no hay una cierta

cantidad. El contador es sólo eso, el elemento que sirve para contar o contar hacia atrás. (LogicBus, 2020)

#### **2.4.1.5. Bloque de datos (DB)**

Son bloques o regiones de memoria que, en comparación a las etiquetas, están determinadas por el operador. Pueden ser de diferentes longitudes y contener variedad de contenido. Las variables comprendidas en la propia DB se almacenan de manera standard cuando se detiene el PLC. (IndustriasGSL, 2019)

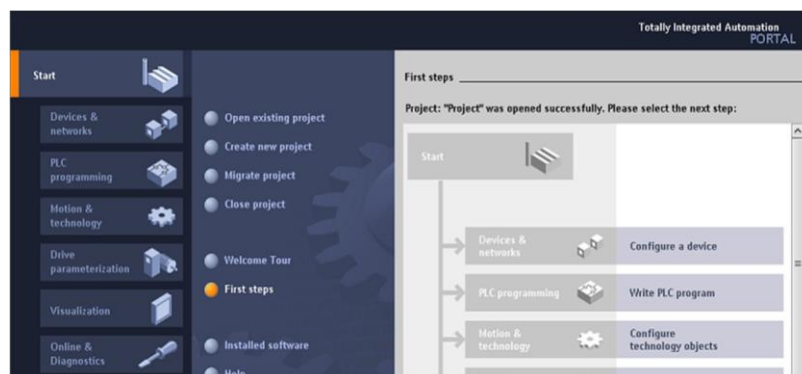
#### **2.4.2. Totally Integrated Automation Portal (TIA Portal)**

Siemens añade todos sus nuevos desarrollos y tecnologías orientadas al futuro en la cartera de hardware TIA e incorporando sus funciones correspondientes. Esto consiente acceder sin limitaciones a la gama completa de servicios de automatización digitalizados, a partir de la organización digital hasta la ingeniería integrada y la labor transparente. Reduce su tiempo de comercialización con herramientas de simulación aumentando la productividad de la planta con funciones adicionales de diagnóstico y gestión de la energía y a su vez la flexibilidad conectándose al nivel de gestión. (Siemens, 2020)

Tiene actualizaciones anuales que lo hacen más flexible, más rápido y productivo. Las innovadoras instrumentos de simulación, la ingeniería afinadamente integrada y el funcionamiento claro de la planta se ajustan perfectamente en TIA Portal. Las nuevas opciones benefician a los integradores de sistemas y a los fabricantes de maquinaria, así como a los operarios de las plantas, lo que convierte al software en la puerta impecable para la

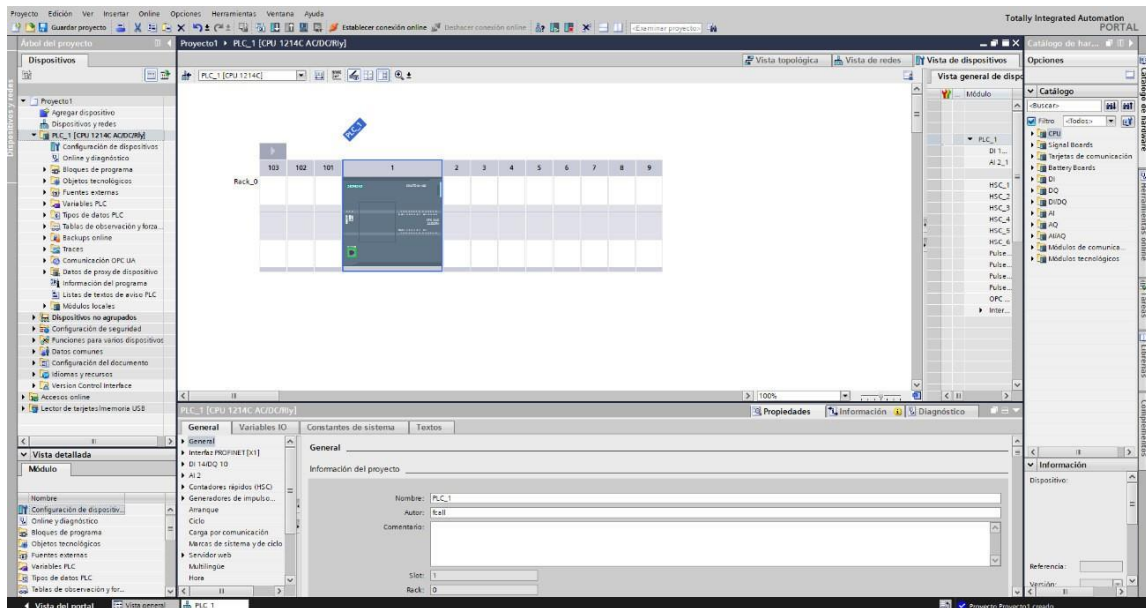
automatización en la sociedad digital. Un flujo de trabajo digital permite utilizar representaciones virtuales de máquinas y sistemas. Se utilizan para simular y probar todos los aspectos antes de construir el producto real, también está acoplado a la TI operativa y a la nube para adquirir aún más flexibilidad y, en última instancia, efectos de mayor calidad. Mantiene un paquete completo de soluciones de automatización para optimizar procesos de ingeniería, posibilita la integración de los softwares (STEP 7, WinCC, SINAMICS Startdrive, SIMOCODE ES y SIMOTION SCOUT TIA) están disponibles funciones adicionales que ofrecen nuevas opciones, como la ingeniería multiusuario en el TIA Portal o la gestión de energía con SIMATIC Energy Suite desde una única interfaz. Por lo tanto, ofrece todo lo que es necesario para una ingeniería integral hoy y en el futuro. (Siemens, 2020)

La principal ventaja de este software es el entorno común para la configuración y programación/parametrización de los elementos clave utilizados en la automatización. analiza este punto en detalle sobre los espacios de trabajo y su uso. Puede conectarse a su dispositivo en línea sin crear un proyecto. Esta opción es muy interesante. Por ejemplo, puede monitorear el estado de su sistema y ejecutar diagnósticos en respuesta a un comportamiento inesperado. (Flix, 2021)



**Figura 5:** Vista inicial de TIA Portal

**Fuente:** (Software TIA Portal – Tomada por el autor, 2022)

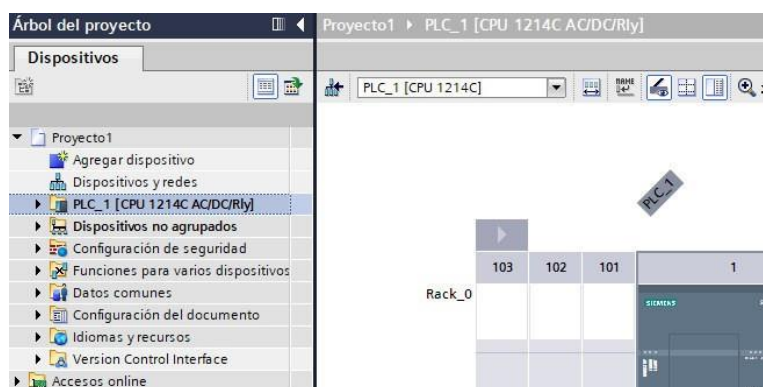


**Figura 6:** Inicio de espacio de trabajo en TIA Portal

**Fuente:** (Software TIA Portal – Tomada por el autor, 2022)

Para mover un elemento dentro del área de edición: B. Un área de programación para trabajar. Todos los usuarios de Windows estarán familiarizados con estas características y sus funciones: cortar, copiar, pegar, eliminar, flecha, deshacer y rehacer. Los símbolos restantes son típicos de los portales TIA. Además, está destinado a la gestión de más dispositivos. Entre esta gestión de están las opciones; Compilar: Verifica que la programación y/o la configuración se puedan configurar de manera completamente consistente y segura en el dispositivo. Es una especie de filtro para evitar subidas de errores al dispositivo. Subir a dispositivo: después de que la opción anterior confirme que la configuración de es correcta, puede enviarla a un dispositivo específico. Por lo tanto, debe seleccionar este dispositivo en el árbol del proyecto y luego hacer click en esta opción. Por supuesto, la configuración no se transfiere mágicamente, pero debe establecer comunicación con el dispositivo. Subir desde el dispositivo a la computadora: esta opción no está disponible para todos los dispositivos. Por ejemplo, es útil para restaurar la programación de un PLC .

Por ejemplo, puede introducir cambios de programa en su proyecto que pueden haber sido cargados desde otra computadora. También es útil para recuperar los valores de los parámetros de memoria volátil de la unidad. Tenga en cuenta que restaurar un programa de PLC requiere una conexión al dispositivo. Por otro lado, la restauración de parámetros desde una unidad se puede realizar fuera de línea simplemente seleccionando el dispositivo en el árbol del proyecto. Simulación: esta opción no está disponible en todos los dispositivos. Puede probar su programación con un PLC o simulación de pantalla. Esto es muy conveniente. En la práctica, la recarga del dispositivo en la configuración usada no es tan rápida como nos gustaría, y con esta opción, la recarga no tiene por qué realizarse únicamente durante la fase de arranque. (Centeno, 2018)



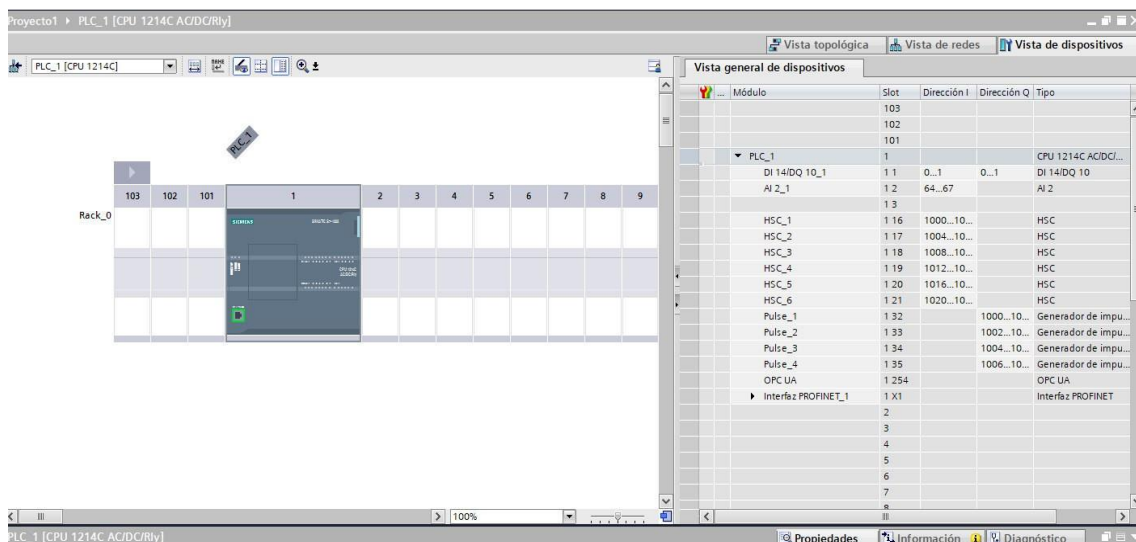
**Figura 7:** *Árbol del proyecto en TIA Portal*

**Fuente:** (Software TIA Portal – Tomada por el autor, 2022)

En el lado posterior del aplicativo se encuentra el árbol del proyecto, esta muestra todos los dispositivos existentes, haciendo clic en cada dispositivo para ver varias opciones de configuración/programación de estos variarán según el tipo de dispositivo, pero debe tenerse en cuenta que algunos menús son iguales para todos los dispositivos. El uso se vuelve mucho más intuitivo. Debajo hay dos paneles que son completamente opcionales, lo que significa que se pueden mostrar u ocultar a elección del usuario. Desde el proyecto de referencia puede



acceder a los elementos de otros proyectos, y desde la vista de detalle puede acceder a varias señales existentes dentro del proyecto. Este panel es muy útil porque le da acceso a variables que se pueden agregar a su programa arrastrando directamente sin cambiar la vista. El área central muestra el área superior denominada área de trabajo o edición, donde puede cambiar diferentes elementos del árbol del proyecto haciendo doble clic en ellos. Esta es el área de configuración de hardware y red, el área de programación de los dos bloques de PLC, también hay diferentes imágenes de la parte de visualización y varias ventanas para parametrizar el variador. (Centeno, 2018)



**Figura 8:** Área central TIA Portal

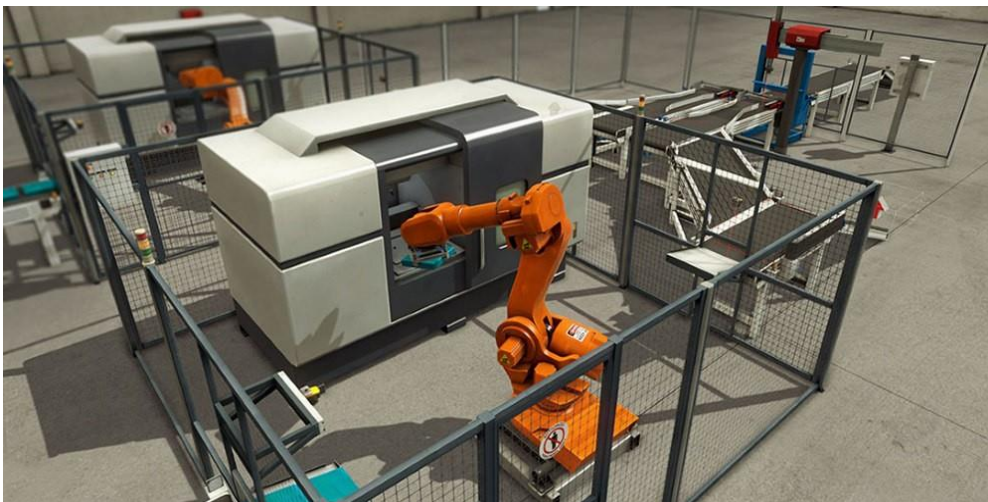
**Fuente:** (Software TIA Portal – Tomada por el autor, 2022)

Antes de compilar, se verifica que el proceso de desarrollo y/o configuración sea completamente consistente y seguro para instalar en el dispositivo. Este es un tipo de filtro para evitar que se carguen errores en los comandos. Y descarga el dispositivo después de usar la opción anterior para validar la configuración puedes empezar a subir a un dispositivo específico. Por lo tanto,

deberá seleccionar el dispositivo específico en el árbol del proyecto y luego hacer clic en esta opción. Por supuesto, la configuración no se transferirá mágicamente, pero será necesario establecer comunicación con el hardware. Descargar del dispositivo a la computadora Esta opción no está disponible para todos los dispositivos. Por ejemplo, será útil para restaurar la programación del PLC S7 1200 AC/DC/RLY, es decir, hacer cambios de programa en un proyecto que puede haber sido descargado, como desde otra computadora. Tenga en cuenta que, para realizar la operación de restauración del programa del PLC, es necesario conectarse al dispositivo, de lo contrario, para restaurar los parámetros del inversor, se realizará sin conexión en línea simplemente seleccionando el equipo en el proyecto. madera. En la emulación, esta opción tampoco está disponible para todos los dispositivos. Esto le permitirá simular el PLC o monitor para probar su programa. Esto es muy conveniente, ya que la carga de los dispositivos en las configuraciones que se utilizan en la práctica no es tan rápida y en este caso no habrá que hacerlo antes de la fase de puesta en marcha y no hacer test de equipos. (Centeno, 2018)

### 2.4.3. Factory IO

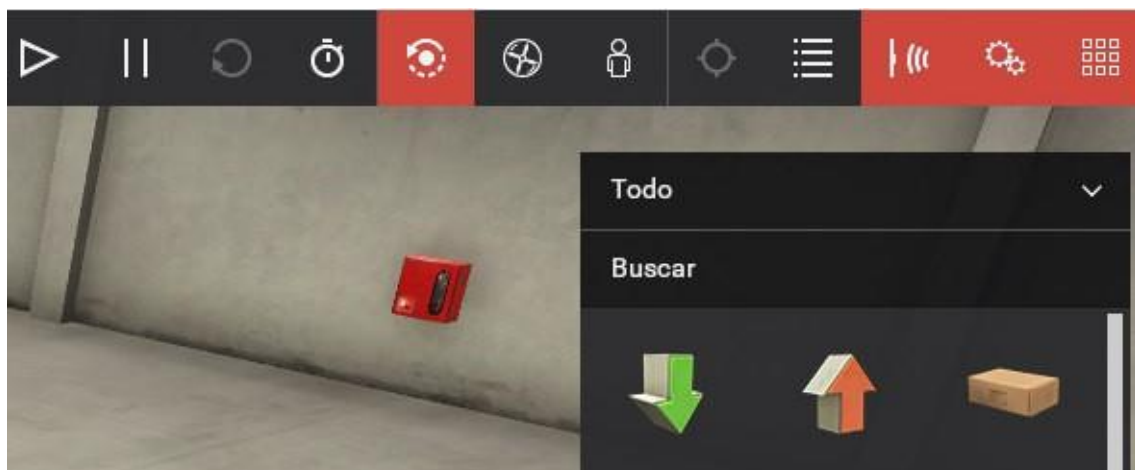
Es un software de simulación 3D desarrollado por Real Games para construir, diseñar y controlar procesos industriales en tiempo real. Este tipo de simulación es totalmente interactiva e incluye gráficos y sonidos de alta calidad que proporcionan un entorno industrial realista. Al mismo tiempo, al conectar el software con un dispositivo externo (PLC o microprocesador), se utiliza tecnología innovadora que permiten crear fácil y rápidamente sistemas 3D automatizados que se pueden controlar en tiempo real. Las principales características que hacen del Factory IO un excelente software para aprender tareas de control realistas a través de PLC son: una fábrica virtual, la capacidad de desarrollar estrategias para diagnosticar fallas debido a conectores abiertos o en cortocircuito; Seguridad y eficiencia del software. Esto es para mostrar aplicaciones del mundo real en la industria y evitar problemas de costos, daños al personal y/o propiedad. Los requisitos mínimos para usar Factory IO son Windows Vista o posterior, Intel Core 2 Dúo 2 GHz, 1 GB de RAM y 500 MB de disco duro. (FactoryIO, 2020)



**Figura 9:** Entorno virtual de FactoryIO  
**Fuente:** (FactoryIO, 2020)

Incluye escenas inspiradas en aplicaciones industriales típicas, que varían en dificultad desde principiante hasta avanzado. El escenario más común es utilizar E/S de fábrica como plataforma de formación de PLC, ya que los PLC son los controladores más utilizados en aplicaciones industriales. Sin embargo, también se puede utilizar con microcontroladores, SoftPLC, Modbus y muchas otras tecnologías. (MHJTools, 2019)

El control de las distintas cámaras disponibles es totalmente necesario para la creación/edición fluida de escenas y mucho más. La capacidad de usar cámaras con FACTORY I/O juega un papel importante. Se sabe que el software en cuestión genera una escena 3D, por lo que debe poder moverse en este espacio 3D sin problemas. La cámara es la clave para interactuar con diferentes partes del programa y es absolutamente necesaria al crear o editar escenas. (Marín, 2018)

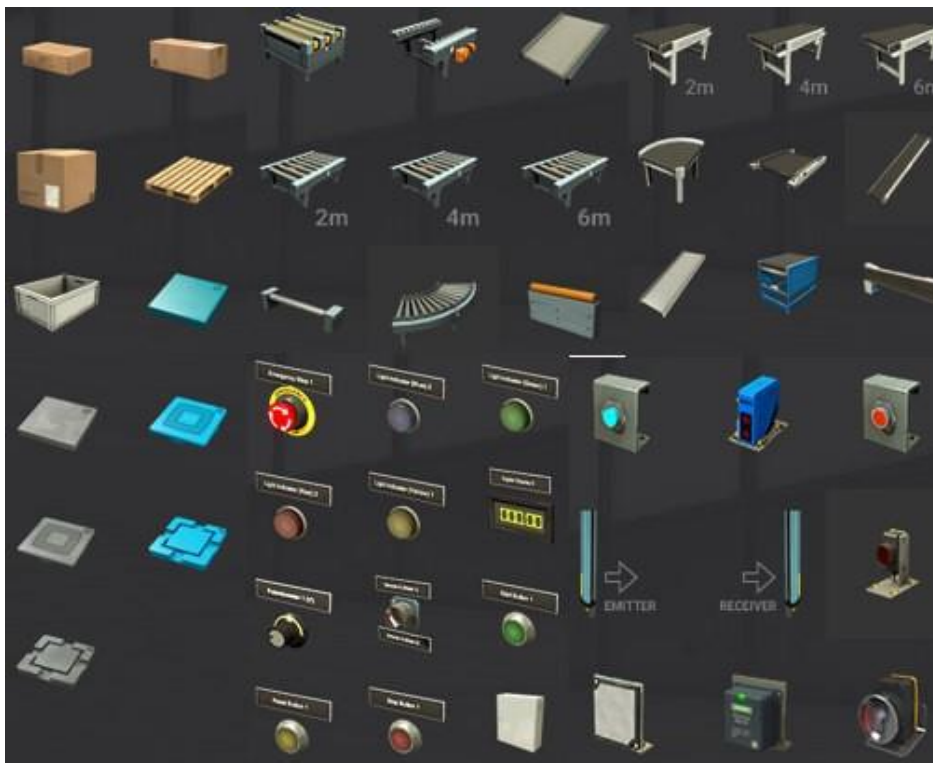


**Figura 10:** Modos de cámara en Factory IO

**Fuente:** (Software Factory IO – Tomada por el autor, 2022)

El modo en órbita es la cámara predeterminada y la única que permite al usuario mover partes presentes en la escena sin producir colisiones, por lo que es fundamental para la construcción de la escena. El modo de uso es muy

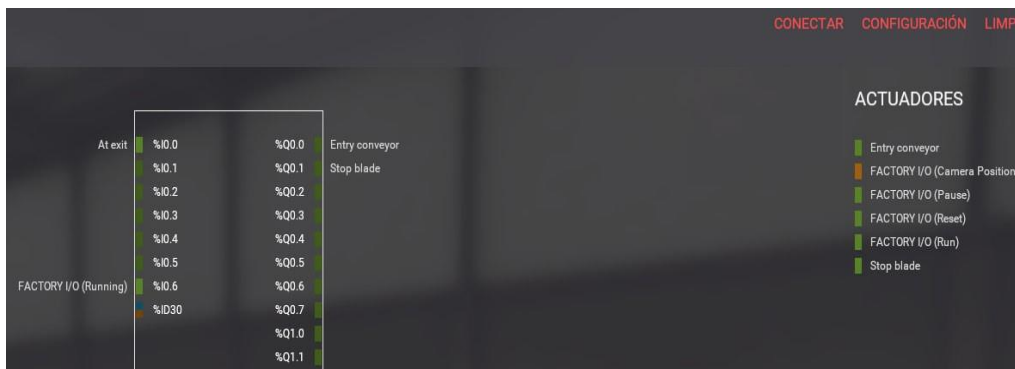
intuitivo y gira en torno a un punto de interés marcado con un punto blanco, este punto se establece haciendo doble click con el botón izquierdo del ratón sobre una de las partes presentes en la escena. Una vez que se define el punto de interés, la cámara se puede girar a su alrededor manteniendo presionado el botón derecho del ratón y arrastrando. Es importante saber que los elementos arrastrados desde la paleta se crearán con la altura definida por este punto, excepción de las partes que normalmente se colocan en el suelo como cintas transportadoras, estaciones, etc. (MHJTools, 2019)



**Figura 11:** Elementos típicos disponibles para simulación en Factory IO

**Fuente:** (Software Factory IO – Tomada por el autor, 2022)

La mayoría de las veces la cámara dron se usa para moverse libremente en el espacio 3D. A diferencia de su antecesora, esta cámara choca con varias partes presentes en la escena, completamente ajena a los sensores colocados en su interior. Con esta cámara, puede inspeccionar el espacio 3D de manera segura y rápida, logrando perspectivas que de otro modo serían imposibles. (Marín, 2018)



**Figura 12:** Tabla de variables o etiquetas en Factory IO

**Fuente:** (Software Factory IO – Tomada por el autor, 2022)

Se pueden editar varias configuraciones de E/S predeterminadas de fábrica mediante la consola. Tomar en cuenta que la mayoría de estas configuraciones no están utilizables a través de la interfaz de usuario y solo se pueden cambiar a través de la consola. Para tener acceso a la consola, hacer clic en Ver > Abrir consola. Puede ejecutar comandos al inicio escribiendo los comandos en un archivo auto.cfg y colocando ese archivo en la carpeta de instalación de Factory I/O, ProgramData\Real Games\Factory IO o Documents\Factory IO. Tomar en cuenta que si este archivo se coloca en la carpeta de instalación Factory I/O o ProgramData\Real Games\Factory IO, siempre se ejecutará independientemente de la cuenta de Windows. Estos

archivos se ejecutan primero, seguidos de los archivos en Documents\Factory

IO

```
CONSOLE
scene.force_quit() : Forces the application to quit without prompting the user about possible lost changes.
scene.load_from_path(string scene_path) : Loads a scene from a path. Use two backslashes when specifying the scene's path.
scene.my_scenes_path : Sets the "My Scenes" path. This command only works on an auto.cfg file.
scene.print_saves() : Prints the scenes list.
scene.quit() : Quits the application.
scene.time_scale : Sets the time scale used for slow motion (use with care, may create instability in the physics simulation).

stats.reset() : Resets the performance statistics.
stats.visible : Shows the performance statistics.

video.apply() : Applies the current settings.
video.keep() : Saves the current settings. This method must be called after video.apply().
video.fullscreen : Sets the application to run in fullscreen. Call video.apply() to apply it.
video.print_current_res() : Prints the current screen resolution.
video.print_supported_res() : Prints the supported screen resolutions.
video.quality : Gets the current video quality.
video.set_quality(int quality) : Sets the video quality (0 = low, 1 = medium; 2 = high; 3 = very high).
video.set_res(int width, int height) : Sets the screen resolution. Call video.apply() to apply it.
video.bloom : Turns the bloom effect on or off.
video.aa : Sets the antialiasing value (0, 2, 4 or 8).
video.pe_aa : Turns the post effect antialiasing on or off. This effect is applied by default on very high quality.
video.ssaao : Turns the screen space ambient occlusion on or off.
video.target_fps : Target frame (how many times the video is rendered per second). Setting to 0 makes the video render as fast as possible.
video.vsync : Sets the number of VSyncs that should pass between each frame (0 to not wait for VSync).

ui.show_camera_navigation : Shows the camera navigation gizmo.
ui.show_sensors : Shows the sensors tags.
ui.show_actuators : Shows the actuators tags.
ui.show_addresses : Shows the addresses of the sensors and actuators tags.
ui.show_welcome_window : Shows the welcome screen at startup.
ui.enabled : Shows the user interface.
```

**Figura 13:** Consola de Factory IO

**Fuente:** (FactoryIO, 2020)

Para la creación de una escena simple se debe seleccionar la cámara orbital y arrastrar los íconos requeridos dentro del entorno 3D y asegurar la altura requerida para el tipo de proyecto a realizar. Se recomienda probar manualmente antes de controlar la escena con un controlador externo. Esto permite al usuario estar seguro de que la escena funciona como se espera y evitar problemas futuros. Pero antes de eso, debe saber qué son las etiquetas y cómo se pueden usar para controlar partes. Cada elemento que existe en la escena tiene al menos una etiqueta. Una etiqueta consta de un nombre y un valor y puede contener tres tipos de datos diferentes: booleano, flotante y entero. FACTORY I/O brinda la capacidad de forzar estos valores y habilitar o deshabilitar sensores o actuadores dentro de la instalación. (Marín, 2018)



Mandar	Valor	Predeterminado	Descripción
<code>app.is_user_admin</code>	Bool	Depende del usuario actual de Windows	Comprueba si el usuario actual es un administrador del sistema.
<code>app.log_to_console</code>	Bool	Falso	Registra todos los errores y advertencias en la consola.
<code>app.print_args()</code>	vacio	-	Imprime args pasados durante el lanzamiento.
<code>app.print_system_info()</code>	vacio	-	Imprime información sobre la tarjeta gráfica, el procesador y el sistema operativo.
<code>app.run_in_background</code>	Bool	Verdadero	Establece la aplicación para que se ejecute en segundo plano cuando la ventana pierda el foco.
<code>app.online</code>	Bool	Verdadero	Permite que la aplicación acceda a Internet.
<code>app.local_feed</code>	Bool	Falso	Carga el contenido de la ventana de bienvenida desde recursos locales.
<code>app.copy_log()</code>	vacio	-	Copia el archivo de registro anterior en. <code>..\Documents\Factory IO</code>
<code>app.web_server</code>	Bool	Falso	Ejecuta un servidor web en segundo plano.
<code>app.web_server_url</code>	cuerda	<code>r'http://*:7410'</code>	Dirección URL del servidor web. Debe establecerse antes de iniciar el servidor.
<code>app.quit()</code>	vacio	-	Cierra la aplicación.
<code>app.force_quit()</code>	vacio	-	Obliga a la aplicación a cerrarse sin preguntar al usuario acerca de los posibles cambios perdidos.

**Figura 14:** Descripción de comandos para control en consola Factory IO

**Fuente:** (FactoryIO, 2020)

## 2.5. Leguajes de programación

Es la forma de comunicarse con una computadora, tableta o teléfono móvil y señalarle lo que se desea realizar. Hay variados tipos de lenguaje, principalmente de alto y bajo nivel, la diferencia es qué tan cerca o lejos está el hardware a programar, esta proximidad tiene que ver con el control sobre un dispositivo, placa o controlador. Los programas normalmente se diseñan o escriben a través de un entorno de desarrollo integrado (IDE). Un IDE generalmente tiene un editor de texto, un compilador y un depurador para el lenguaje que usa, y una plataforma correspondiente para ejecutar después. Suelen tener un constructor de interfaz gráfica (GUI). (Cilsa, 2019)

Entre los principales lenguajes de programación están los siguientes:



### 2.5.1. PYTHON

Es un lenguaje de programación de elevado ras manejado para desplegar todo arquetipo de aplicaciones. En comparación a otros lenguajes tal como Java o .NET, es un lenguaje descifrado, debido a que una aplicación escrita en Python no precisa ser compilada para establecerse, sino que puede ser escrita directamente por una computadora usando un programa llamado interpretador, por lo que no es necesario "traducirlo" a lenguaje de máquina. Es muy similar al lenguaje humano, es fácil de leer y escribir. Se trata de un lenguaje multiplataforma de código abierto, es gratuito, por lo que puede desarrollar software sin restricciones. Posee una extensa diversidad de superioridades al ocuparse con ciencia de datos, inteligencia artificial, amaestramiento automático, y Big data, entre diversos otros campos en apogeo. (Santander, 2020)

```
horas = float(input("Introduce tus horas de trabajo: "))
coste = float(input("Introduce lo que cobras por hora: "))
sueldo = horas * coste
print("Tu sueldo es", sueldo)
```

**Figura 15:** Ejemplos de programación Python  
**Fuente:** (Santander, 2020)

Es un lenguaje de programación multiplataforma que hace relativamente fácil el desarrollo de aplicaciones en cualquier sistema operativo. Varias tecnologías se adaptan muy bien a Python debido a su simplicidad y potencia de procesamiento de datos. Así que no hay duda de que el lenguaje está volviendo a los lugares de trabajo donde cada vez más empresas buscan expertos en Python. (Santander, 2020)

Este lenguaje de programación es utilizado principalmente en procesos como:

### **2.5.1.1. Análisis de datos**

Su sencillez y amplia gama de bibliotecas de proceso de data forjan que Python sea excelente para examinar y dirigir grandiosas cuantías de data en tiempo real. Este lenguaje hondamente descifrado a menudo está detrás de diferentes softwares de análisis de datos, por lo que muchas empresas ahora lo usan directa e indirectamente. Analizar grandes cantidades de datos y transformarlos en información útil de big data es una de las especialidades de Python. Este se ha instituido como el principal recurso de la simbolización para el desarrollo de herramientas que admitan el análisis, ajuste y proceso de los datos. Por esto mismo, en un universo donde el Big Data posee cada vez más peso para las sociedades, aprender Python se vuelve una antelación de mayor atravesado para aquellos que indagan penetrar en el mundo de los datos metódicos. Aunque existen otros lenguajes de simbolización que además se han hecho un pedante en el sector, lo cierto es que son muchos las demostraciones por los que Python se ha consumado por imponer en la industria del análisis de datos. Una de las primordiales ventajas es lo sencillo que consecuencia su aprendizaje. Cualquiera con unos pequeños conocimientos de programación puede instruirse de los principios de este lenguaje sin dificultad. Y a medida que los vaya asimilando irá conociendo ciertos de las ventajas que muestra, como la inconstancia y la reproducibilidad. Es decir, no simplemente permite realizar variadas tareas, sino que una fracción de código, un script escrito en Python se logra imitar en cualquier plataforma. (Gil, 2021)

### **2.5.1.2. Minería de Datos**

Es una habilidad que consiente el estudio de grandiosas bases de data con la intención de anunciar propensiones futuras. Este es un asunto complicado que Python consigue revelar mediante el barrido y colocación de datos y el uso de algoritmos de noviciado automático para facilitar el análisis de datos. (Cilsa, 2019)

### **2.5.1.3. Datos científicos**

Después del desarrollo de motores numéricos como 'Pandas' y 'NumPy', Python reemplazó a MATLAB, el lenguaje que usan los científicos para trabajar con grandes cantidades de datos. La razón es la misma que en el apartado anterior. Python, junto con numerosas bibliotecas existentes, es muy adecuado para este tipo de tareas debido a su simplicidad y potencia para manipular grandes cantidades de datos. (Guevara, 2020)

### **2.5.1.5. Blockchain**

Conocida mundialmente como la base de las criptomonedas, también funciona muy bien con Python. Es un lenguaje versátil, seguro y rápido que es muy útil para construir cadenas de bloques y permite a los desarrolladores crear cadenas de bloques simples en menos de 50 líneas de código, lo que facilita la creación de cadenas complejas. (Santander, 2020)

### **2.5.1.6. Aprendizaje Automático**

La robótica y la IA se han tornado autoinstruidas a medida que resuelven más y más data. De este modo, recogen información cada vez más distinguida que les consiente tomar buenas disposiciones, Python asimismo es muy seguro en esta área en el que el proceso de data eficiente es transcendental. (Santander, 2020)

### **2.5.2. C++**

Este un modelo de un lenguaje de programación agrupado de múltiples ejemplos que es especialmente imperativo y encaminado a objetos, comprendida la programación genérica y funcional. Un software escrito en una expresión imperativo es un grupo de instrucciones que le dicen a un computador como ejecutar una tarea de manera vertiginosa, eficaz y correcta. En un lenguaje imperativo, cada paso del software conoce su estado, definitivo por los valores de las variables manejadas, y el estado del programa se cambia de acuerdo con la instrucción. La mayoría de las ejecuciones de hardware de computadora están trazadas para ejecutar código de máquina escrito en forma imperativa. Las variables son sitios de almacenamiento e instrucciones. (UVa, 2020)

Una de las principales ventajas de programar en C++ es el alto rendimiento como una de sus principales características. Esto se debe a que puede llamar directamente al sistema operativo, es un lenguaje compilado para cualquier plataforma, tiene muchos parámetros de optimización y se integra directamente con el lenguaje ensamblador. Es un lenguaje moderno que permite la creación, asociación y manipulación de datos complejos e implementa varios

patrones de diseño. C y C ++ son ampliamente multiplataforma en el sentido de que están presentes en casi cualquier programa o sistema escrito en esos lenguajes o parte de ellos, desde navegadores web hasta el propio sistema operativo. (Robledano, 2019)

Los principales inconvenientes de C++ son que es un lenguaje muy extenso con muchas líneas de código, requiere una compilación por plataforma y es complicado de depurar debido a errores. Además, manejar bibliotecas es más complejo que otros lenguajes como Java o .Net, y la curva de aprendizaje es mucho mayor. (UVa, 2020)

Es un lenguaje de programación multiparadigma desarrollado y estandarizado por Microsoft Corporation como parte de la plataforma .NET, y más tarde por ECMA (ECMA-334) e ISO (ISO/IEC23270). Es uno de los lenguajes de programación desarrollados para Common Language Framework. Su sintaxis básica se deriva de C/C++, utilizando el modelo de objetos de la plataforma .NET similar al de Java, pero incorporando mejoras derivadas de otros lenguajes. C# es parte de la plataforma .NET, pero es una API mientras C# es otro lenguaje de programación diseñado para crear programas en esa plataforma. Ya existen compiladores implementados que proporcionan el marco Mono-DotGNU para generar programas para varias plataformas como Microsoft Windows, Unix, Android, iOS, Windows Phone, Mac OS y GNU/Linux. (Ortego, 2019)

El código fuente escrito en C# se compila en un lenguaje intermedio (IL) que cumple con la especificación CLI. El código IL y los recursos, como mapas de bits y cadenas, generalmente se almacenan en ensamblados con extensiones .dll. Un ensamblado contiene un manifiesto que proporciona información de tipo,

versión y referencia cultural. Cuando ejecuta su programa C#, el ensamblado se carga en el CLR, este realiza la compilación Just-In-Time (JIT) para convertir el código IL en instrucciones de máquina nativas. Además, CLR proporciona otros servicios relacionados con la recolección de elementos no utilizados, el manejo de excepciones y la gestión de recursos. El código que se ejecuta en CLR a veces se denomina "código administrado". (Microsoft, 2020)

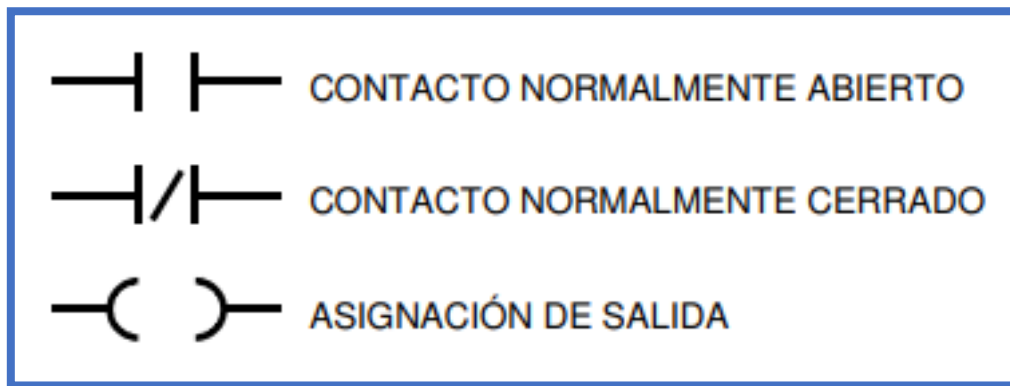
El "código no administrado" se compila en el lenguaje de máquina nativo para un sistema en particular. La interoperabilidad entre lenguajes es una característica central de .NET. El código IL generado por el compilador de C# se ajusta a la especificación de tipo común (CTS). El código IL generado a partir de C# puede interactuar con el código generado a partir de las versiones F#, Visual Basic y C++. Hay más de 20 otros idiomas admitidos por CTS. Un único ensamblado puede contener varios módulos escritos en diferentes lenguajes y los tipos pueden hacer referencia entre sí como si estuvieran escritos en el mismo lenguaje. Además de los servicios de tiempo de ejecución, .NET también incluye una amplia gama de bibliotecas para admitir diversas cargas de trabajo. Están organizados en espacios de nombres que proporcionan varias funciones útiles. Las bibliotecas incluyen todo, desde E/S de archivos, manipulación de cadenas y análisis XML hasta marcos de aplicaciones web y controles de Windows Forms. Una aplicación típica de C# hace un uso extensivo de las bibliotecas de clases .NET para manejar tareas de infraestructura comunes. (Microsoft, 2020)

#### **2.5.5. Diagrama Ladder**

Es un lenguaje gráfico derivado del lenguaje de retransmisión. Los símbolos se utilizan para representar contactos, bobinas, etc. Su principal

ventaja es que los símbolos básicos están estandarizados y cumple con las normas IEC y es utilizado por todos los fabricantes. (UNLP, 2018)

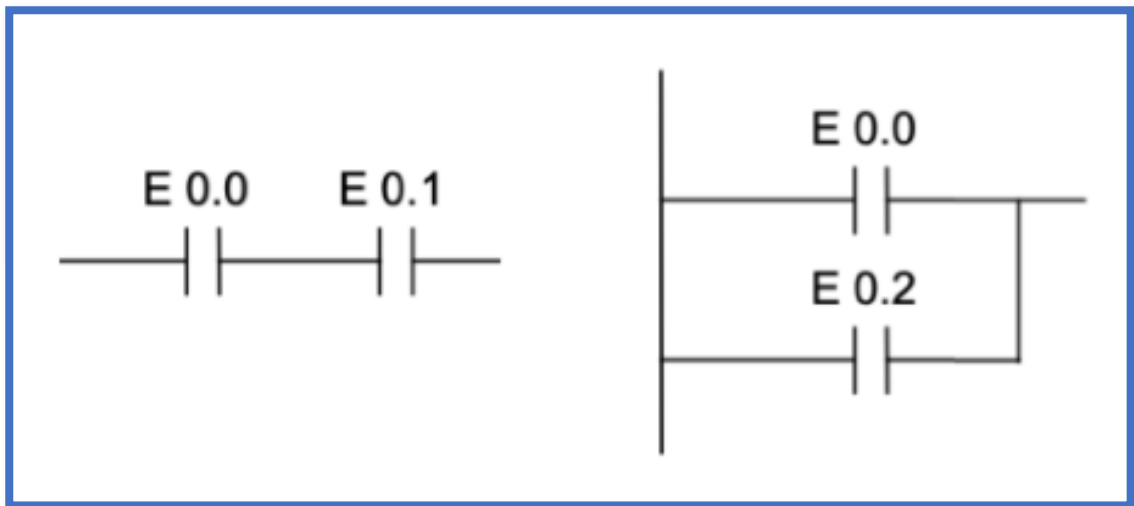
Los símbolos básicos son:



**Figura 16:** Símbolos básicos del lenguaje Ladder  
Fuente: (UNLP, 2018)

Los contactos normalmente abiertos cuentan con dos platinos y su posición natural es abierta. Al aplicar tensión en la bobina del relevador, el contacto cambia la posición de sus platinos para unirlos. Se utilizan para cerrar un circuito eléctrico cuando se energiza la bobina del relevador, mientras que los contactos normalmente cerrados mantienen sus dos platinos unidos de forma natural. Cuando se energiza la bobina del relevador se separan. Estos contactos se utilizan para desconectar o también para impedir la conexión de un circuito eléctrico al aplicar tensión a la bobina del relevador. A diferencia de los dos tipos anteriores, estos cuentan con tres platinos ya que es la combinación de un contacto abierto y un contacto cerrado. Uno de los platinos cambia de posición cada vez que se energiza la bobina. Son prácticos en aplicaciones que requieren de un relevador que cuente con una combinación de contactos abiertos y cerrados. Por ejemplo, para un circuito de control de un motor trifásico que considere la protección contra inversión de giro. (FinderNet, 2019)

En estas figuras la línea vertical de la izquierda es un conductor con tensión y las líneas verticales que aparecen a la derecha Tierra. Los diagramas suelen ilustrar diferentes tipos de comportamientos eléctricos. Se puede utilizar para sintetizar sistemas de control y realizar la programación de PLC utilizando las herramientas de software adecuadas. Se pueden implementar funciones lógicas de forma sencilla. Por ejemplo, en la figura siguiente (Fig.11) se implementa una función AND y una OR. (Brunete, 2018)



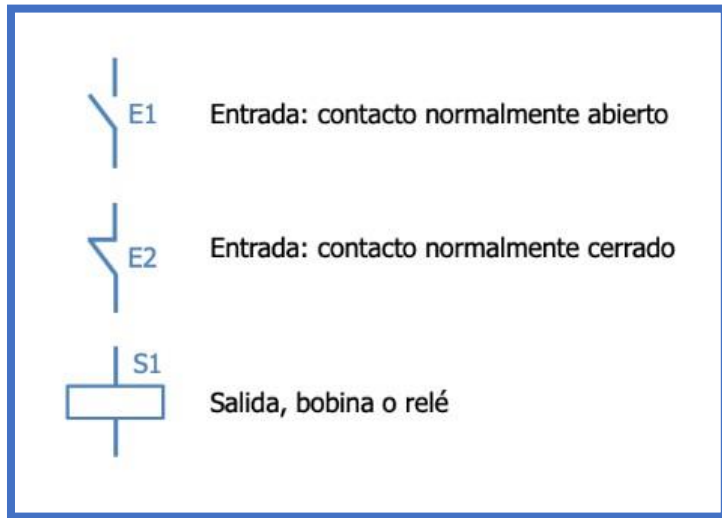
**Figura 17:** Implementación de funciones básicas: AND (izquierda) y OR (derecha)

**Fuente:** (Brunete, 2018)

### 2.5.6. Esquema de contactos

Es una representación que corresponde directamente a los esquemas eléctricos que deben crearse para construir la automatización. Regulado por DIN40713-6. La Fig.12 muestra la representación de los elementos más típicos de este esquema. (UNLP, 2018)

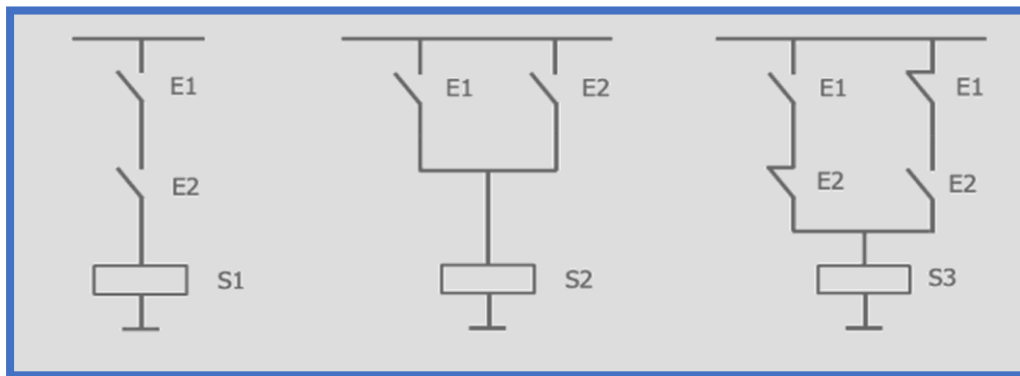




**Figura 18:** Elementos básicos del esquema de contactos

**Fuente:** (Brunete, 2018)

La implementación de funciones lógicas básicas (AND, OR, XOR) se muestra en la Fig.13:



**Figura 19:** Funciones lógicas con diagrama de contactos: AND, OR, XOR

**Fuente:** (UNLP, 2018)

Al conectar los contactos normalmente abiertos en serie, se puede implementar la puerta AND.

INPUT (I0.0)	INPUT (I0.1)	OUTPUT (Q0.0)
0	0	0
0	1	0
1	0	0
1	1	1

**Tabla 1:** Resultados de implementación de compuerta AND en Esquema de contactos.

**Fuente:** (MasterPLC, 2018)

Cuando los contactos A y B de las entradas se aplica un 1 (tíue), entonces la salida OULPUL' pasa a nivel alto, mientras que al conectai los contactos normalmente abiertos en paralelo, se puede implementar una compueta OR. (Huítado, 2017)

INPUT (I0.0)	INPUT (I0.1)	OUTPUT (Q0.0)
0	0	0
0	1	1
1	0	1
1	1	1

**Tabla 2:** Resultados de implementación de compueta OR en Esquema de contactos.

**Fuente:** (MasterPLC, 2018)

Cuando en cualquier de las entradas A y B se aplica un 1 (tíue), entonces la salida OULPUL' pasa a nivel alto. Una puerta lógica es el bloque de construcción para el circuito digital. Toma una entrada binaria (0, 1) y como resultado da una salida binaria basada en la entrada provista. Las compuetas lógicas que se utilizan en el PLC son lo básico que usted debe aprender si desea mejorar sus habilidades eléctricas y electrónicas. En esta publicación, aprenderá a escribir la programación en PLC utilizando Logic Gates. Para la programación de controladores lógicos programables (PLC) necesitamos un software del PLC, El PLC es un dispositivo de control y el software nos sirve para escribir la programación usando el lenguaje de programación LADDER. (Huítado, 2017)

## 2.7. Sensores

Son dispositivos utilizados para detectar variables estequiométricas, como resultado, convierte una cantidad física en una variable eléctrica para finalmente

medirla, por ejemplo, la temperatura la capacidad para convertir la entrada (I) de temperatura en un cambio de resistencia para establecer el calor térmico en una métrica fija. Los métodos de monitorización y control solicitan sensores para medir cuantías físicas tales como perspectiva o posición, distancia, aceleración, deformación, temperatura, fuerza y vibración. (UVa, 2020)

Existen diferentes tipos de sensores, con funcionalidades diferentes como:

### 2.7.1. Sensores de proximidad

Es un transductor que detecta objetos o señales que están contiguo al elemento de detección. Existen diferentes tipos de sensores de proximidad según el principio físico utilizado. Los más comunes son los interruptores de posición, sensores fotoeléctricos como capacitivos, inductivos e infrarrojos. Son la elección predilecta para la generalidad de las aplicaciones que demandan fiabilidad, en la localización sin contacto, de objetos metálicos en las máquinas o equipos de automatización. (PeppeRL, 2019)



**Figura 20:** Sensores de proximidad Pepperl+Fuchs

**Fuente:** (PeppeRL, 2019)

## **2.8. Actuadores**

Son dispositivos utilizados para convertir energía en movimiento o aplicar fuerza. Un dispositivo toma energía de una fuente particular (energía producida por aire, fluido o electricidad) y la transforma en el movimiento deseado. Los dos tipos básicos de movimiento requeridos son lineales y rotatorios, pero el movimiento oscilatorio también es común. Un actuador lineal convierte la energía en movimiento lineal, empujando o tirando. Los actuadores rotativos, por otro lado, convierten la energía en movimiento vibratorio y se usan típicamente en una variedad de accesorios, como válvulas de mariposa y válvulas de bola. Los actuadores se utilizan comúnmente en aplicaciones industriales y de fabricación. Dispositivos como válvulas, motores, interruptores y bombas dependen en gran medida de ellos. Hay diferentes versiones para cada tipo de actuador y vienen en diferentes tamaños, diseños y modos de operación dependiendo de la aplicación. (Especificar, 2019)

### **2.8.1. Actuador neumático**

Transforma la energía neumática, es decir aire comprimido en energía mecánica, normalmente en movimiento que puede ser lineal o rotativo. El aire comprimido se puede suministrar sin componentes eléctricos, lo que reduce el riesgo de explosiones e incendios en los actuadores neumáticos. Estos actuadores también se utilizan comúnmente en sistemas de automatización industrial. La provisión de aire comprimido forma que los actuadores neumáticos procedan de forma muy rápida, lo que los hace muy adecuados para aplicaciones que requieren alta velocidad. (DirectIndustry, 2020)



**Figura 21:** Actuator neumático marca  
Cy. Pag  
**Fuente:** (DirectIndustry, 2020)

Los actuadores neumáticos son los dispositivos más vendidos debido a su asequibilidad, larga vida útil y facilidad de instalación. Sin embargo, es importante asegurarse de que el aire comprimido sea de buena calidad. De lo contrario, el actuator podría dañarse. En general, se supone que la presión de suministro para los actuadores neumáticos es de 6 bar. La fuerza de empuje y tracción depende del diámetro interior del actuator. Su velocidad es tan alta como 0,2 a 0,3 m/s. Asimismo, el empuje teórico está entre 20 N y 50.000 N a presiones entre 4 y 8 bar. Sin embargo, son de escasa fuerza en comparación a los actuadores hidráulicos. Otra notada e importante primacía de los actuadores neumáticos es su peso liviano y su macicez. (Flix, 2021)

### 2.8.2. Actuador hidráulico

Tienen fluido líquido, o aceite, como fuente de energía. Al ser un fluido incompresible, los actuadores hidráulicos pueden soportar fuerzas más altas que los actuadores neumáticos. Por lo tanto, es ideal para cargas pesadas. Además, el control de velocidad es más simple que los actuadores neumáticos y es bastante para el frecuente flujo de aceite que se integra a la cámara del cilindro. Los actuadores hidráulicos poseen una prórroga en su vida útil. (DirectIndustry, 2020)



**Figura 22:** Actuador hidráulico marca Eckart  
**Fuente:** (DirectIndustry, 2020)

Estos dispositivos tienen una velocidad de respuesta más lenta que los actuadores neumáticos. Además, pueden ocurrir fugas de aceite, reduciendo el rendimiento, y la circulación de fluidos en las tuberías genera pérdidas de presión. Además, la alta presión (50-700 bar) aumenta el riesgo de accidentes. La inflamabilidad del aceite también puede aumentar el riesgo de incendio. Finalmente, es una tecnología costosa debido al alto costo de los componentes y mantenimiento preventivo. A la hora de preferir un actuador hidráulico, se encarga tomar muy en cuenta su sustento, fundamentalmente a las uniones que resguardan las cámaras de los cilindros. (Flix, 2021)

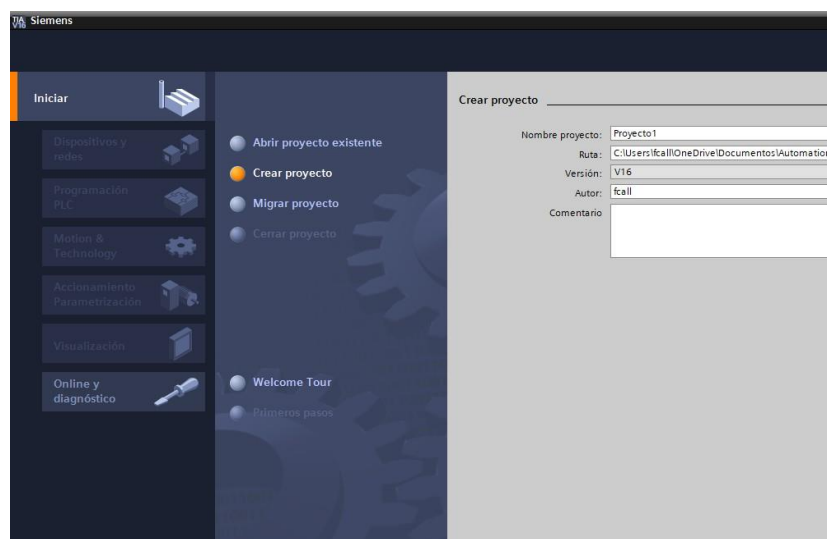
## CAPÍTULO 3: CONSTRUCCIÓN Y SIMULACIÓN

### 3. Consideraciones Iniciales

Para seleccionar el carácter y modos de clasificación se visitó una empresa Courier provincial con el propósito de definir y describir términos de procesos actuales. En esta indagación se consideró factores como lugar de entrega, tamaño del paquete, tipo de mercadería y conteo de bultos que son las principales necesidades actuales que requieren los trabajadores encargados en la clasificación y repartición de paquetes. Esto es tomado en cuenta para la realización de la construcción y simulación de este proyecto.

#### 3.1. Conexión de TIA Portal con FactoryIO

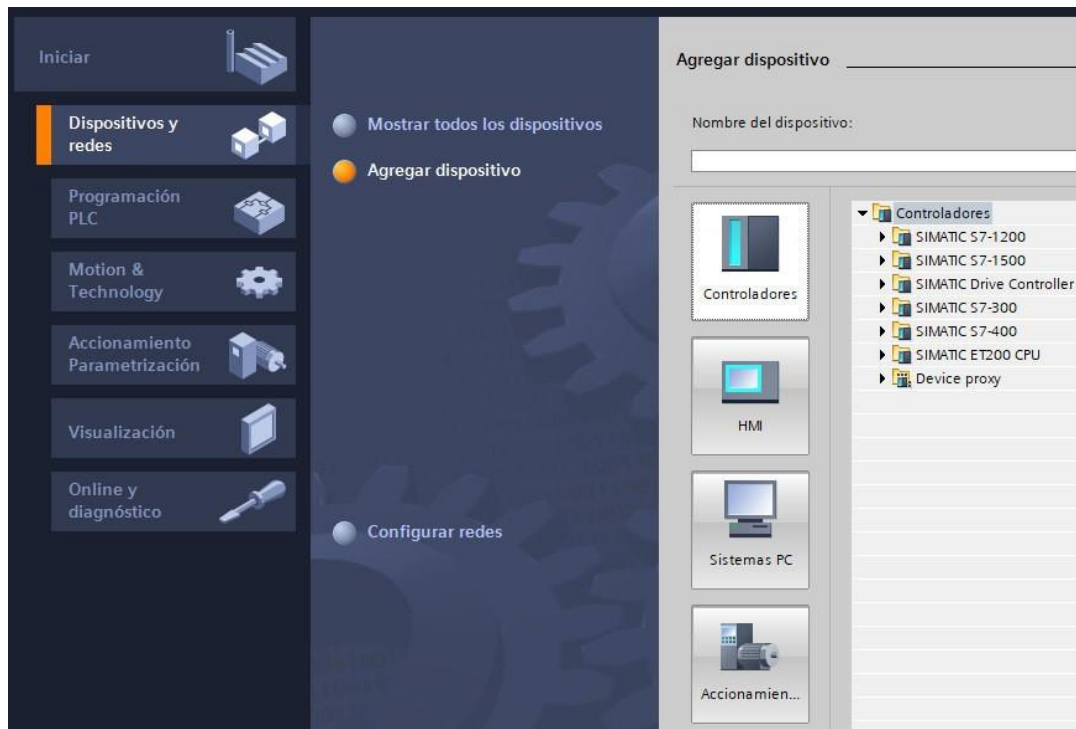
El software FactoryIO y las características del PLC virtual utilizado en TIA Portal deben configurarse correctamente para garantizar que la simulación de las distintas estaciones coincida con el programa cargado en el Controlador Lógico Programable.



**Figura17:** Interfaz inicial TIA Portal V16

**Fuente:** Autor

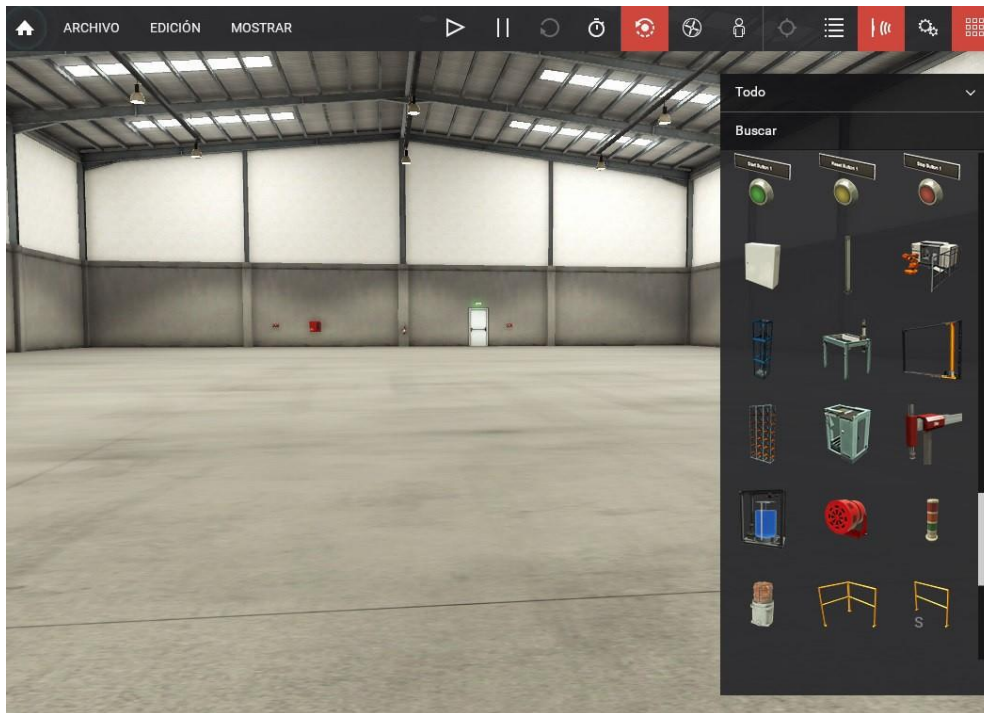
Al crear un proyecto se configuran los denominados “primeros pasos”, estos consisten en el ajuste de los dispositivos y redes, en este caso agregamos un PLC SIMATIC S7-1200. En este paso también se podrían agregar paneles o pantallas HMI, pero al ser una simulación en un entorno virtual no es necesario añadirlos.



**Figura 23:** Lista de controladores disponibles en TIA Portal V16  
**Fuente:** Autor

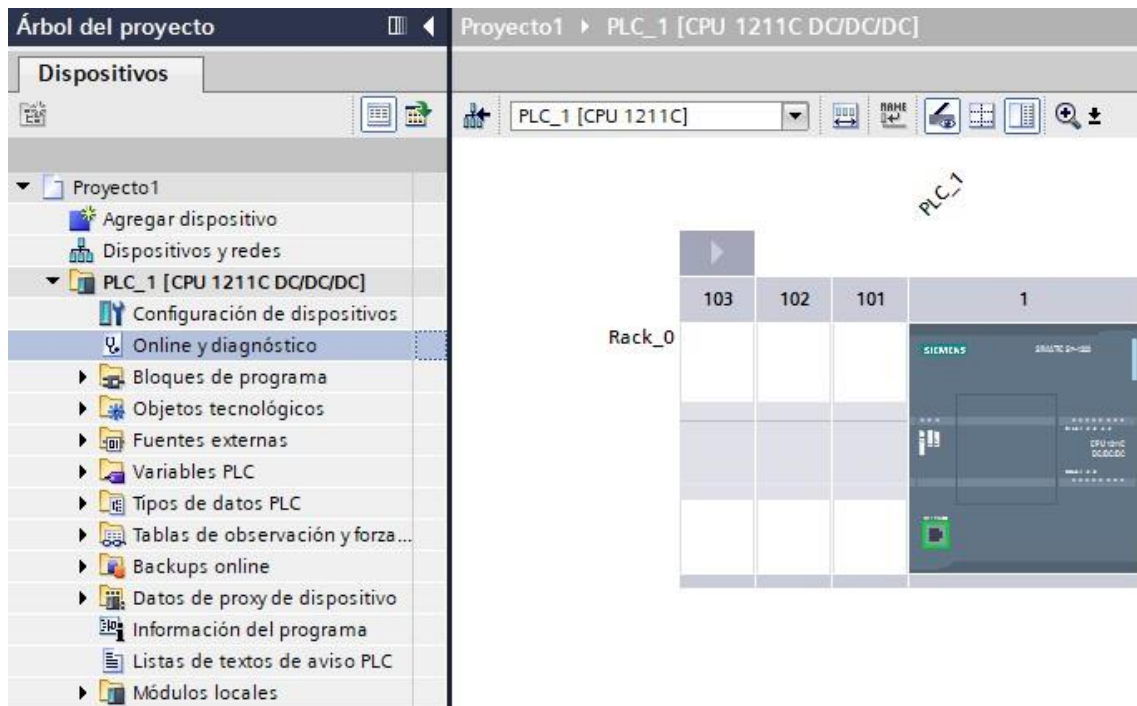
La configuración inicial del dispositivo en TIA Portal deberá ser tomada en cuenta para el segundo software (FactoryIO) en el que haremos el diseño gráfico simulando un entorno industrial virtual.





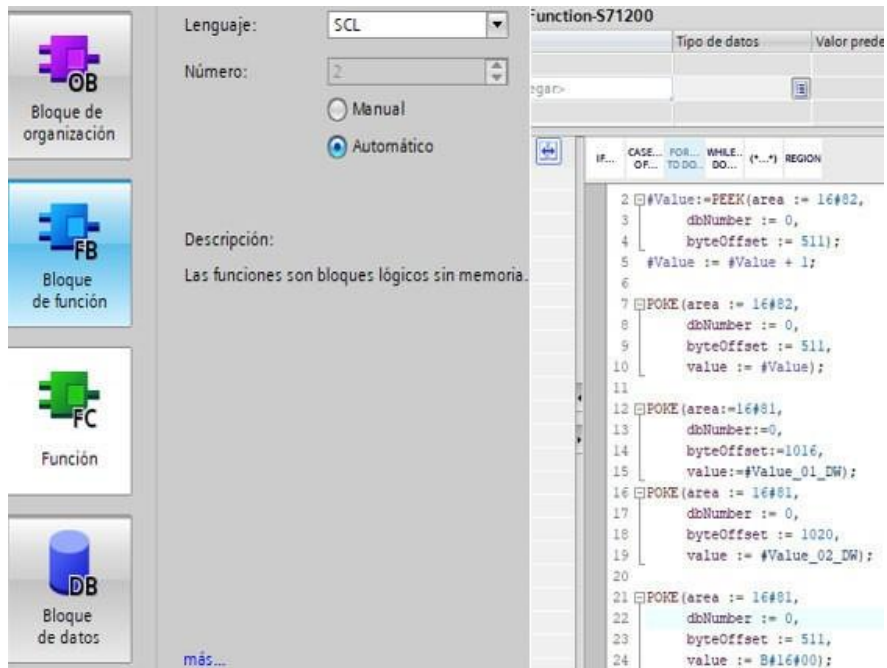
**Figura 24:** *Interfaz inicial FactoryIO*  
**Fuente:** Autor

Para fijar una conexión entre TIA Portal y FactoryIO se debe ubicar el árbol de proyecto del software de Siemens, luego de reconocer el PLC que se configura anteriormente se podrá desplazar una lista que contiene los bloques de programa, objetos tecnológicos, fuentes externas, variables PLC, tipos de datos PLC, tablas de observación, Backups y módulos locales.



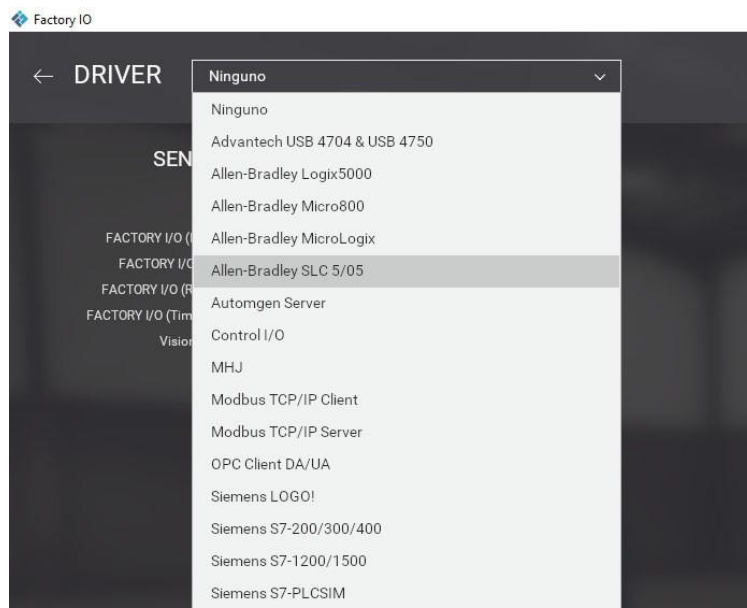
**Figura 25:**Árbol del proyecto en TIA Portal V16  
Fuente: Autor

Luego crear un Bloque de función de programa (Se distingue del resto de bloques de programa por ser de color verde como se muestra en la Fig.29), Es necesario establecer SCL como lenguaje de programación predeterminado de este bloque. A continuación, usando C++ se escriben las líneas de programación imprescindibles para realizar la conexión con Factory IO.

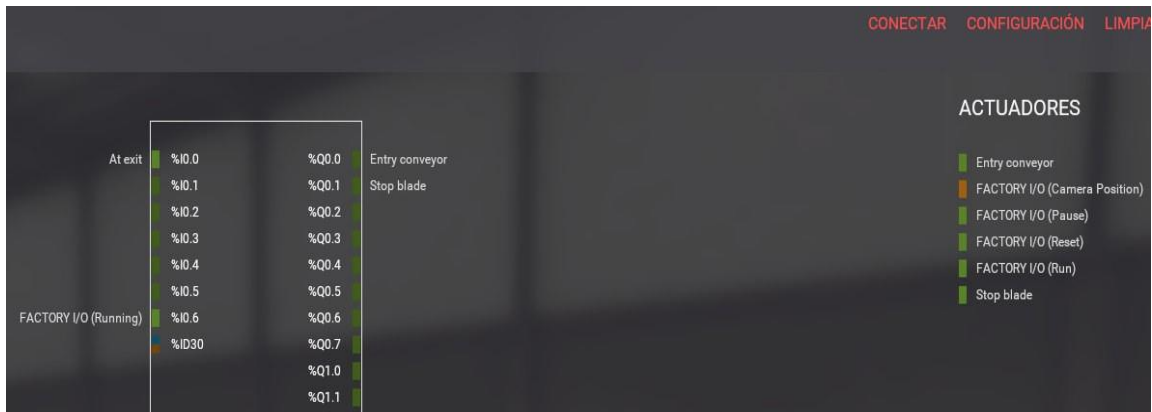


**Figura 26:** Bloque de función de programa con C++  
Fuente: Autor

Por consiguiente, el Bloque de función se queda guardado y enlistado en el PLC. Luego en el software FactoryIO presionando la tecla “F4” se pueden ver la lista de PLC’s compatibles con la versión del programa, en este caso se elige “Siemens S7-PLCSIM”, el mismo que fue declarado en el TIA Portal, finalmente se presiona conectar y se establece una conexión en espera de una compilación de bloques de programa con las variables expuestas.



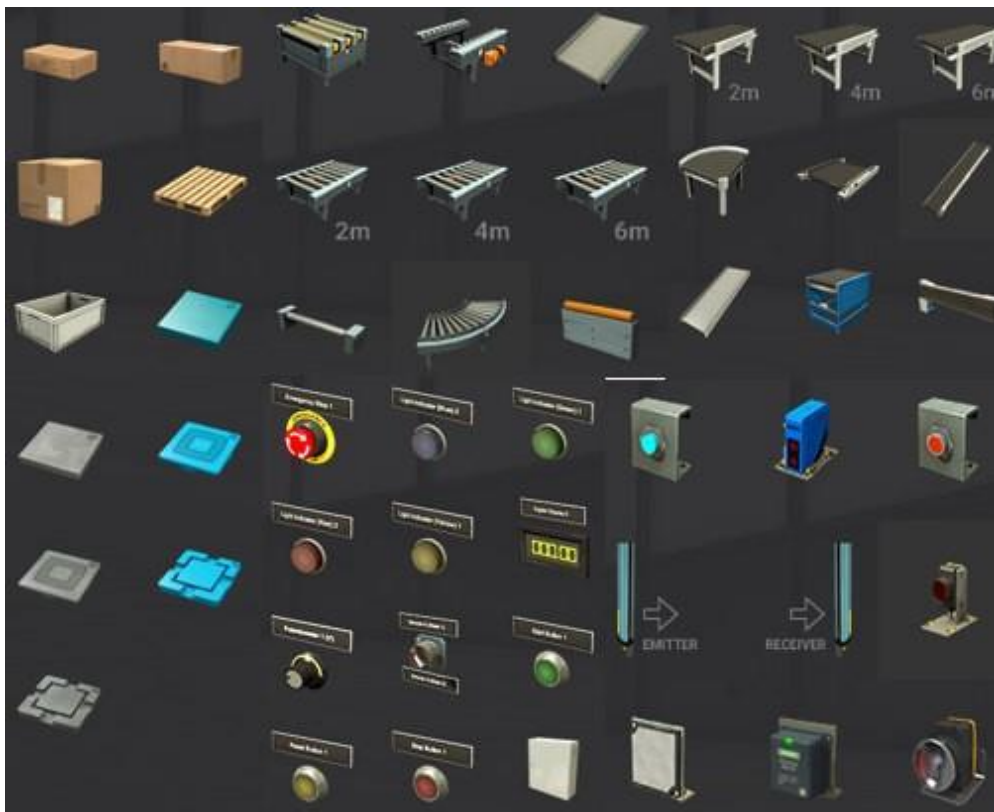
**Figura 27:** Lista de Drivers compatibles en FactoryIO  
Fuente: Autor



**Figura 28:** Variables expuestas por Siemens S7-PLCSIM  
**Fuente:** Autor

### 3.2. Diseño gráfico de clasificadora de paquetes

Como se puede visualizar en el lado derecho de la interfaz FactoryIO en la Fig.32 están enlistados los elementos disponibles a graficar como: Elementos pesados, ligeros, sensores, operadores, estaciones, dispositivos de advertencia y pasarelas.



**Figura 29:** Elementos disponibles para diseñar en FactoryIO  
**Fuente:** Autor

Conforme se añaden los elementos, juntando y ensamblando la clasificadora se van exponiendo las variables ubicándose automáticamente en las entradas(I) y salidas(Q) del PLCSIM del apartado drivers.



**Figura 30:** Inicio de diseño de clasificadora con variables  
Fuente: Autor

En un escenario típico de la vida real se propuso un punto inicial, es decir la banda transportadora (A), donde el camión llega de un Courier matriz a una oficina sucursal de provincia a dejar un cargamento de paquetes, la banda transportadora encamina los paquetes hacia la siguiente fase (B) donde estará un sensor de visión que se va a encargar de reconocer cada bulto, para finalmente ser dirigido a la última fase donde los actuadores se van a encargar de redirigir cada paquete.

La forma de clasificación queda a criterio del Courier, ya sea este por tamaños, direcciones, etc. Adicionalmente se colocó un panel de operador donde estarán las botoneras que servirán como mando de la clasificadora y contadores





### 3.3. Diseño de programación en TIA Portal

Para el correcto funcionamiento de la clasificadora y que los elementos ensamblados en FactoryIO tengan movilidad propia se debe declarar las mismas variables otorgadas. Dentro del árbol de proyecto y el PLC configurado hay una carpeta para declarar variables, en esta se crean tablas de entradas, salidas, marcas y estados. Siempre respetando el orden y nombre que concedió FactoryIO a las variables.

#### 3.3.1. Tabla de Entradas

*Tabla 3: Entradas declaradas en TIA Portal*

Nombre	Tipo de datos	Dirección
At exit	Bool	%I0.0
Start	Bool	%I0.1
Reset	Bool	%I0.2
Stop	Bool	%I0.3
Emergency Stop	Bool	%I0.4
Manual	Bool	%I0.5
Visión sensor	DWord	%ID30

Fuente: **Autor**

### 3.3.2. Tabla de Salidas

**Tabla 4:** Salidas declaradas en TIA Portal

Nombre	Tipo de datos	Dirección
Entry conveyor	Bool	%Q0.0
Stop blade	Bool	%Q0.1
Exit conveyor	Bool	%Q0.2
Sorten 1 turn	Bool	%Q0.3
Sorten 1 belt	Bool	%Q0.4
Sorten 2 turn	Bool	%Q0.5
Sorten 2 belt	Bool	%Q0.6
Sorten 3 turn	Bool	%Q0.7
Sorten 3 belt	Bool	%Q1.0
Start light	Bool	%Q1.1
Reset light	Bool	%Q1.2
Stop light	Bool	%Q1.3
Counter 1	DWord	%QD30
Counter 2	DWord	%QD34
Counter 3	DWord	%QD38

Fuente: **Autor**



### 3.3.3. Tabla de Estados

*Tabla 5: Estados declarados en TIA Portal*

Nombre	Tipo de datos	Dirección
Estado 0	Bool	%M0.0
Estado 1	Bool	%M0.1
Estado 2	Bool	%M0.2
Estado 3	Bool	%M0.3
Estado 4	Bool	%M0.4
Estado 5	Bool	%M0.5
Estado 6	Bool	%M0.6
Estado 7	Bool	%M0.7
Estado 8	Bool	%M1.0
Estado 9	Bool	%M1.1

Fuente: **Autor**

### 3.3.4. Tabla de Marcas

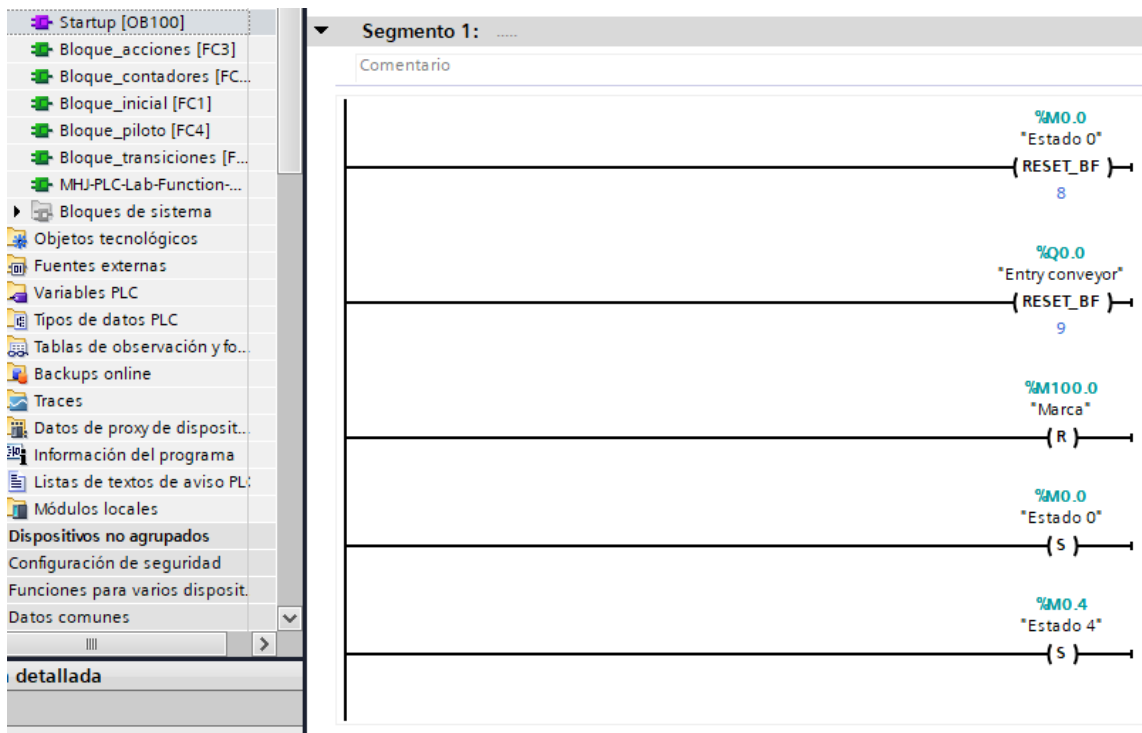
*Tabla 6: Marcas declaradas en TIA Portal*

Nombre	Tipo de datos	Dirección
Marca	Bool	%M100.0
Encendido	Bool	%M100.1

Fuente: **Autor**

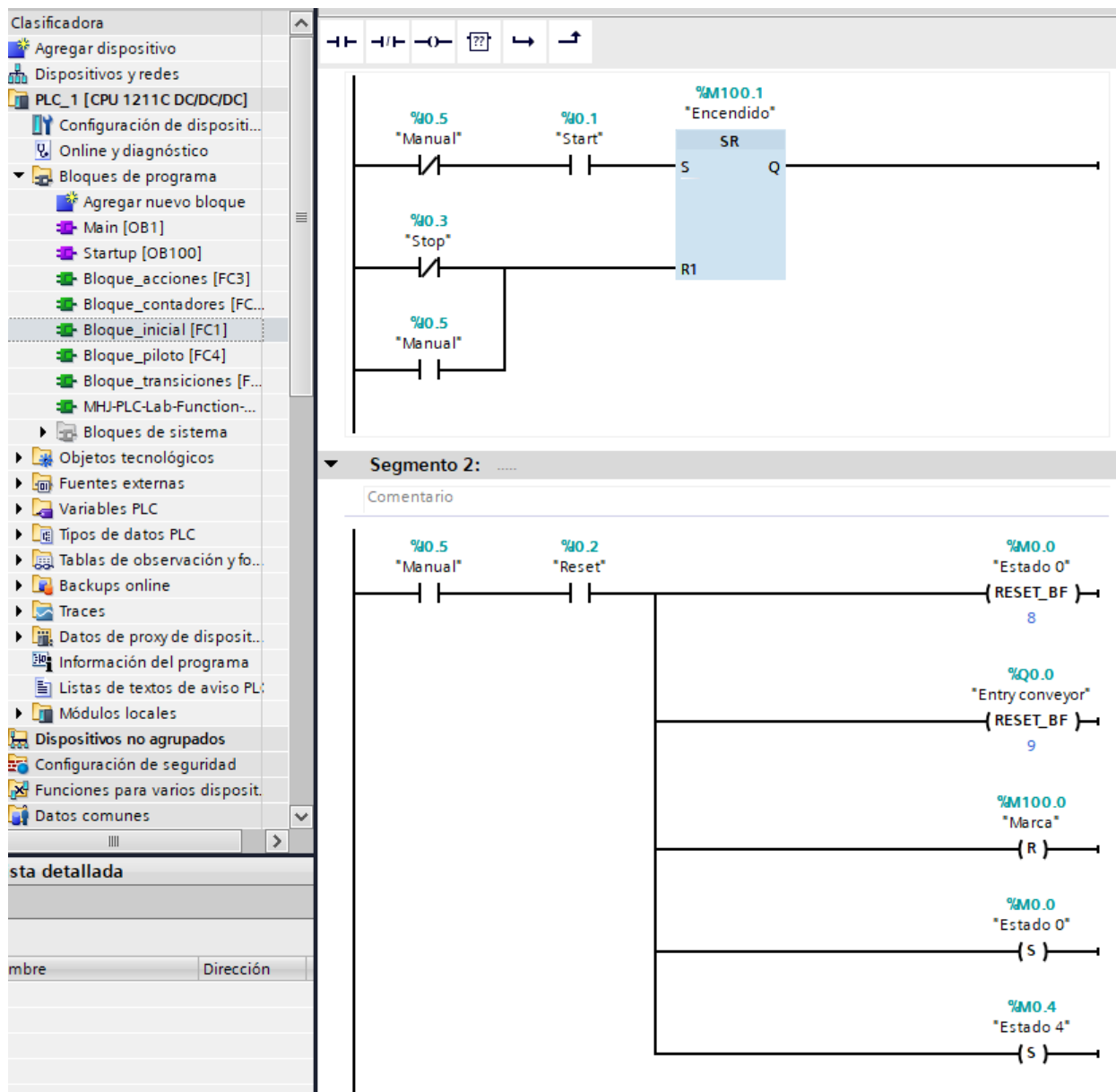
### 3.3.5. Bloques de programación

Se realiza un bloque organización de programa, se trata del StartUp OB100, módulo de arranque que se ejecuta en un solo ciclo antes de iniciar el primer ciclo de programa, se asignan salidas con Set y Reset en el segmento 1, respetando la otorgación de las variables del PLCSIM FactoryIO.



**Figura 33:** Bloque de organización StartUp  
**Fuente:** Autor

A partir de lenguaje KOP se configura el bloque Inicial, consiste en dos segmentos de programa ajustando el Start, Stop, Manual y Reset de la clasificadora, cada una está representada por “%Q.N”

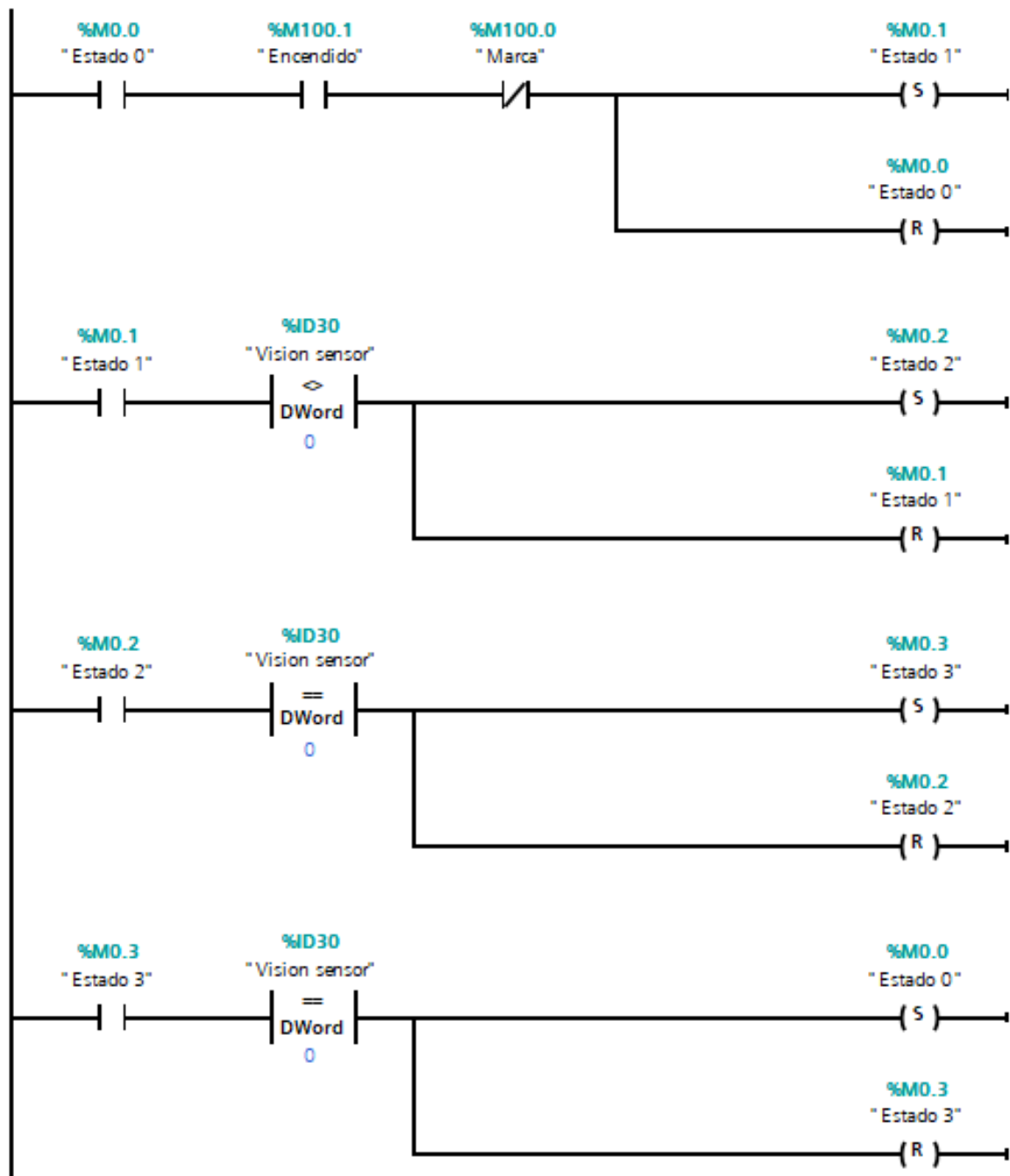


**Figura 34:** Segmentos de programación en Bloque Inicial  
**Fuente:** Autor

El siguiente bloque para programar es el de transiciones, esta parte se encarga de la orden dada por el PLC a partir de las lecturas tomadas por el sensor, es posible usando comparadores, sets y resets. El Courier interviene en este paso ya que el operador decide cómo se va a realizar la clasificación de paquetes. Este bloque se define en 2 segmentos con una programación Ladder algo extensa.

## Segmento 1: .....

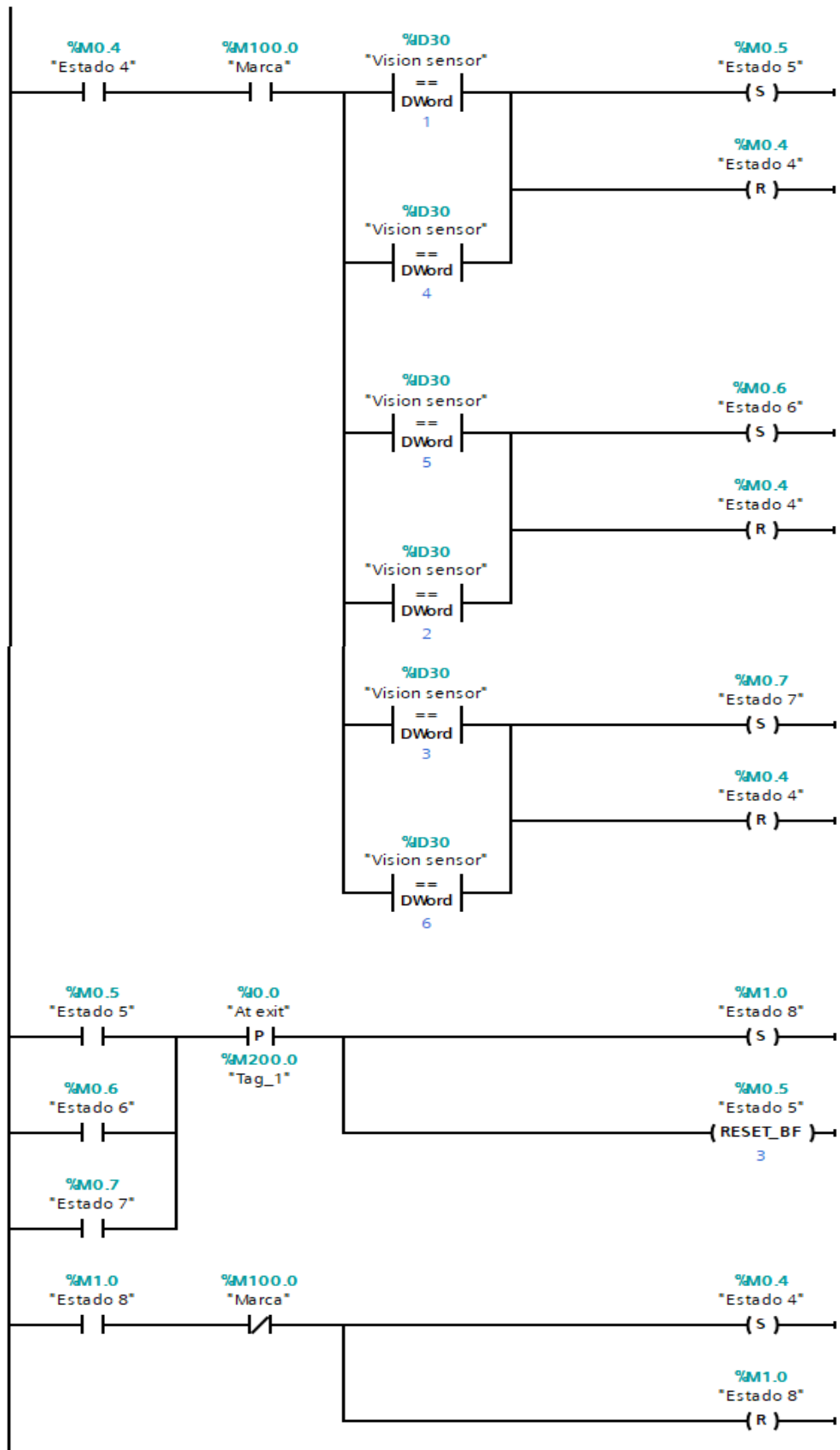
Comentario



**Figura 35:** Segmento 1 del bloque de transiciones  
Fuente: Autor

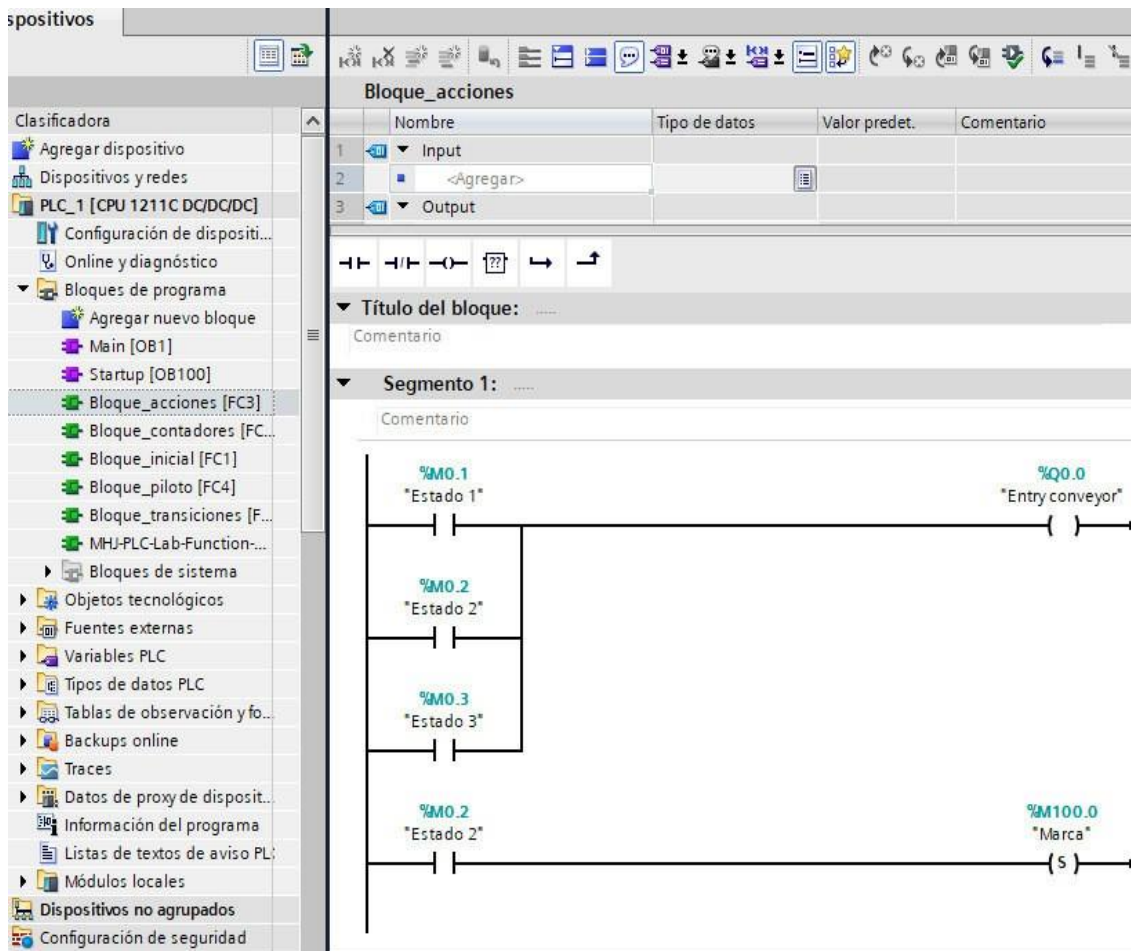
▼ Segmento 2: .....

Comentario



**Figura 36:** Segmento 2 del bloque de transiciones  
Fuente: Autor

El bloque de acciones es indispensable ya que se encarga del funcionamiento de cada actuador, estos van a mover y redirigir cada paquete hacia el final de carrera.



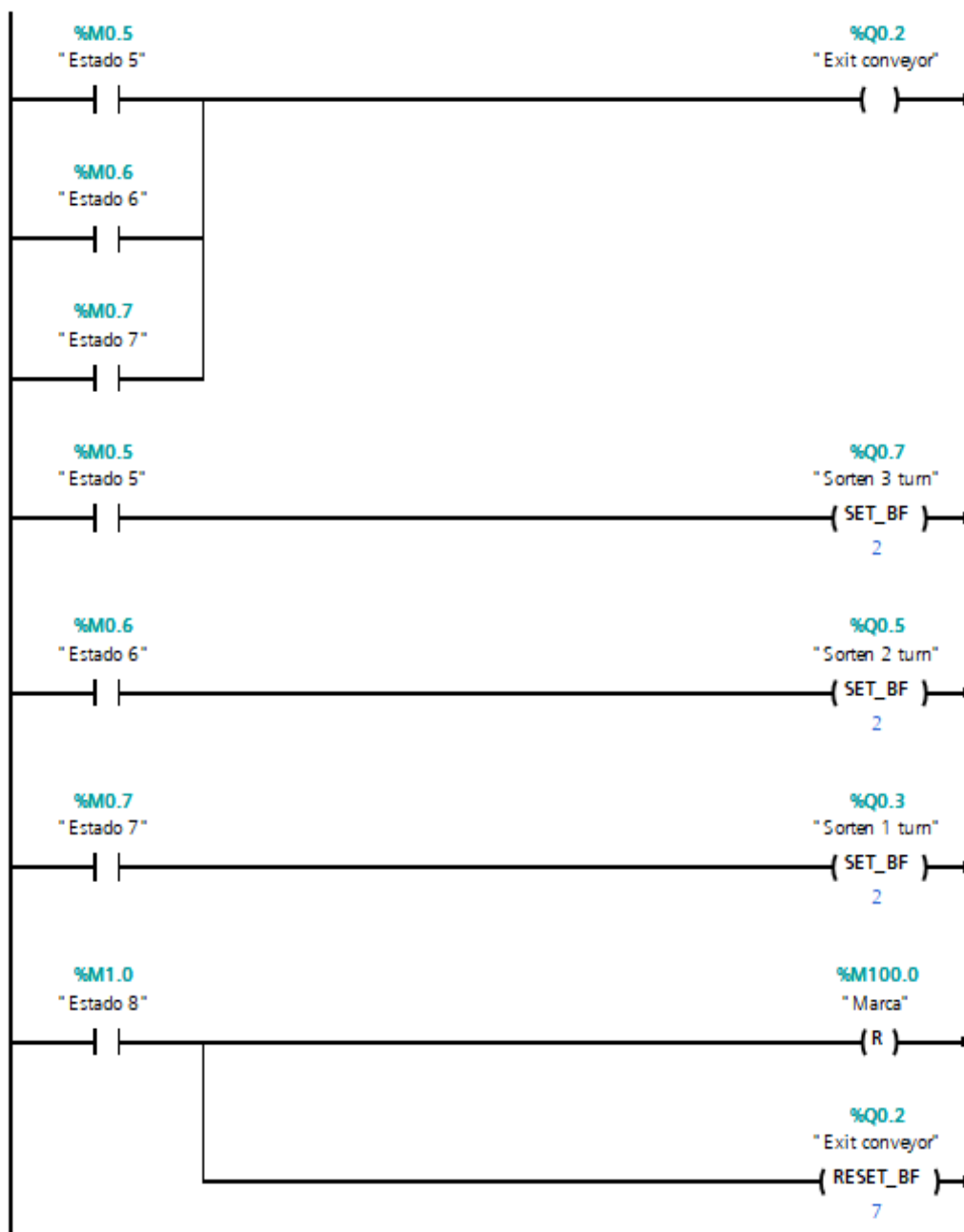
**Figura 37:** Segmento 1 del bloque de acciones

**Fuente:** Autor

Se define en dos segmentos utilizando cada estado y salida proporcionado por FactoryIO.

▼ Segmento 2: .....

Comentario



**Figura 38:** Segmento 2 del bloque de acciones

**Fuente:** Autor

El siguiente bloque para programar es el piloto, se encarga de las luces piloto del panel de control del operador, cada uno con su respectiva función como Encendido o Start Light, se usa un contacto normalmente cerrado con la variable encendido para el Stop Light y Reset para Reset light.

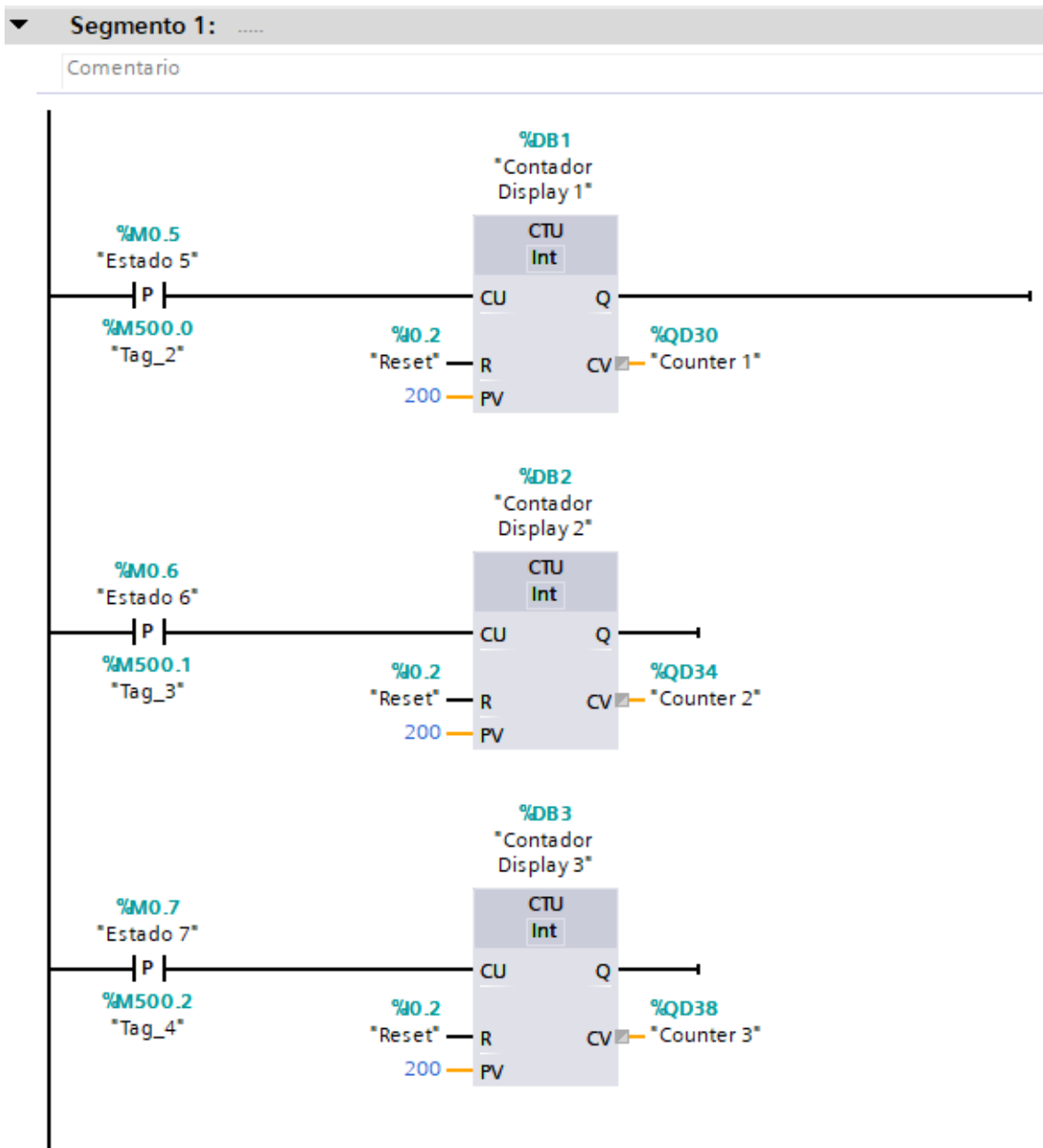


**Figura 39:** Segmento 1 del bloque piloto

**Fuente:** Autor

Finalmente, en el bloque de contadores se usan tres líneas en un segmento con un contador ascendente cada una, configurado con un Reset, el valor de los Counters en CV con contaje actual y el valor con el que se activa la salida (PV) en 200s. Con ese segmento los displays contadores quedan con total funcionalidad.

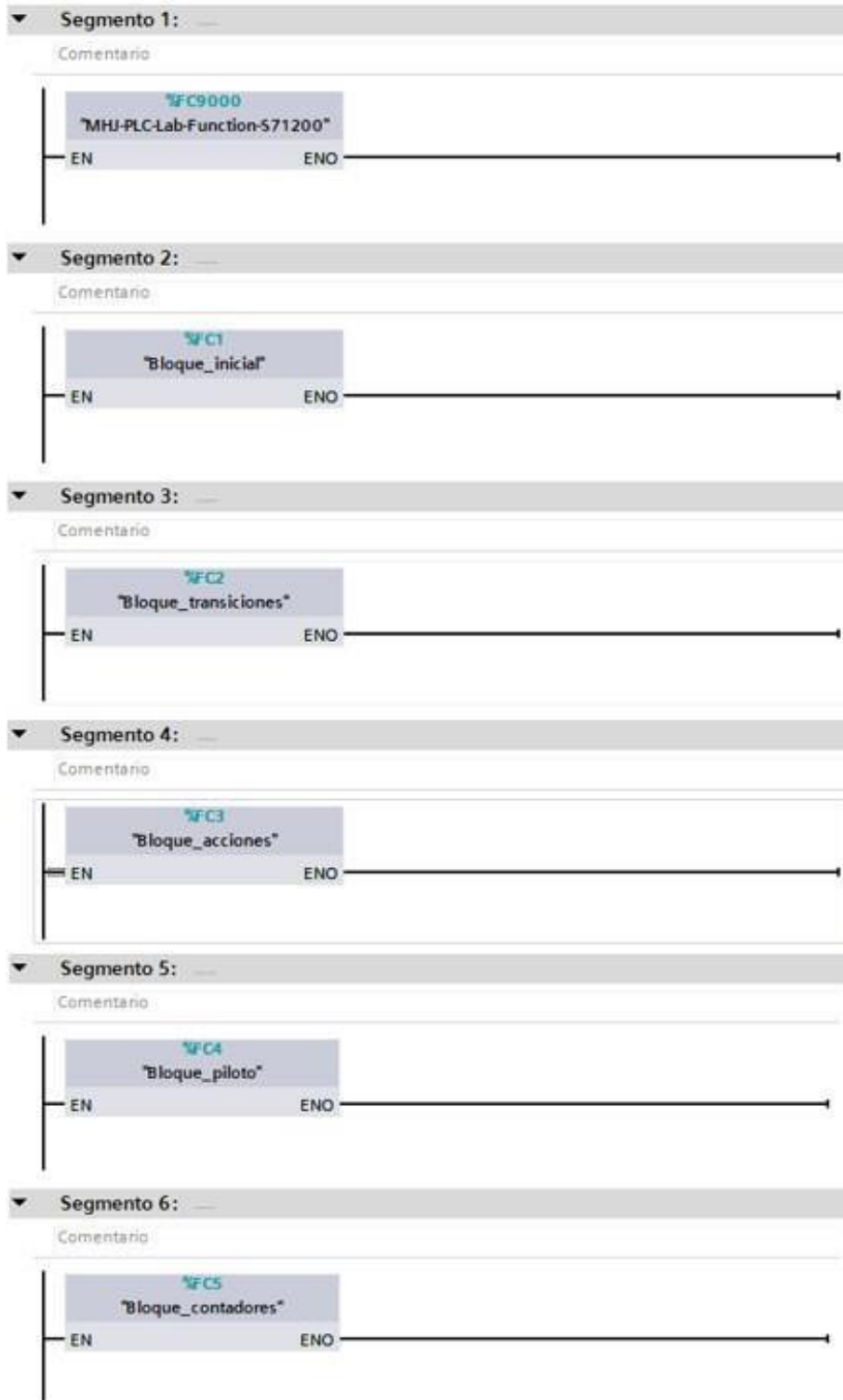




**Figura 40:** Segmento 1 del bloque contadores

**Fuente:** Autor

Antes de realizar la compilación total del programa se debe realizar un bloque organizacional de programa más citando cada bloque de función creado, es decir: El bloque de programación C++ para la conexión con FactoryIO, bloque inicial, de transiciones, de acciones, piloto y contadores. Este bloque organizacional llevaría el nombre "Main".



**Figura 41:** Bloque organizador Main

Fuente: Autor

## **CAPÍTULO 4: CONCLUSIONES Y RECOMENDACIONES**

En este capítulo se detallan las conclusiones obtenidas del método diseñado para la disminución en tiempos de entrega en agencias Couriers sucursales de provincias.

### **4.1. Conclusiones**

Se realizó la determinación de los procesos que median el trabajo como la importación y exportación de bultos o documentos, con ello se consigue conocer y tener una vista clara de las actividades y roles desempeñadas por los colaboradores u operativos de una empresa Courier provincial. Posterior a la determinación se diagnosticaron insuficiencias en los procesos afines a la operación de manejo y clasificación de los bultos que se manejan a nivel interprovincial. Mal uso del tiempo por tareas mal realizadas, tareas sin valor agregado, etc., y uno de los problemas que la mayoría de las empresas Courier enfrentan con más frecuencia que otras, en ciertos casos es el impacto físico sobre el cual se desarrollan los procesos operativos.

Con lo expuesto se elaboró en este proyecto una propuesta de mejora, la operación de manejo y distribución de paquetería buscando la optimización a través de la automatización del proceso, se prevé que con el uso de una clasificadora automática como la actualmente modelada y diseñada puede reducir tiempos de entrega de paquetería en un 80%. Aplicando esta mejora no solo habrá un claro desarrollo profesional, también se brindará mayor eficacia mejorando la operatividad y funcionalidad de una empresa Courier provincial.

## 4.2. Recomendaciones

- Se recomienda la adaptación del proyecto a las necesidades de la empresa Courier, esto debido a los cambios del espacio físico disponible en las bodegas y oficinas sucursales.
- La eficiencia de todo el proceso depende mucho del sensor aplicado para la detección de paquetes, se recomienda el uso de un sensor de alta gama.
- Evitar ambientes húmedos y salinos en el espacio donde se vaya a implementar la clasificadora.
- En caso de cambios en la programación o finales de carrera, asegurarse de el correcto funcionamiento en la simulación antes de compilarlo y subirlo al PLC

## Referencias

- Aula21. (14 de Abril de 2018). *Curso Aula 21*. Obtenido de <https://www.cursosaula21.com/que-es-un-automata-programable-o-plc-y-como-funciona/#:~:text=¿Qué%20es%20un%20autómata%20programable,de%20los%20dispositivos%20de%20salida.>
- Binda, N. U. (07 de Octubre de 2013). Obtenido de [https://scholar.google.com/ec/scholar\\_url?url=https://revistas.ucr.ac.cr/index.php/economicas/article/download/12730/11978/&hl=es&sa=X&ei=T7\\_-YsqYDoLGsQLJuoKgAQ&scisig=AAGBfm0MwZRSEuEqs-9NUIVKw4oYadUTXQ&oi=scholar](https://scholar.google.com/ec/scholar_url?url=https://revistas.ucr.ac.cr/index.php/economicas/article/download/12730/11978/&hl=es&sa=X&ei=T7_-YsqYDoLGsQLJuoKgAQ&scisig=AAGBfm0MwZRSEuEqs-9NUIVKw4oYadUTXQ&oi=scholar)
- Brunete, A. (22 de Septiembre de 2018). Obtenido de [https://bookdown.org/alberto\\_brunete/intro\\_automatica/diagrama-de-escalera.html](https://bookdown.org/alberto_brunete/intro_automatica/diagrama-de-escalera.html)
- Celi, E. (29 de Junio de 2021). *Primicias Ec*. Obtenido de <https://www.primicias.ec/noticias/economia/nueva-empresa-postal-entrega-paquetes/>
- Centeno, P. (27 de Septiembre de 2018). Obtenido de [https://oa.upm.es/49911/1/PFC\\_PEDRO\\_CENTENO\\_POMARETA.pdf](https://oa.upm.es/49911/1/PFC_PEDRO_CENTENO_POMARETA.pdf)
- Cilsa. (02 de Marzo de 2019). Obtenido de <https://desarrollarinclusion.cilsa.org/tecnologia-inclusiva/que-es-un-lenguaje-de-programacion/>
- DirectIndustry. (19 de Abril de 2020). Obtenido de [https://guide.directindustry.com/es/que-actuador-elegir/#:~:text=Los%20actuadores%20se%20dividen%20en,a%20presión\)%20y%20actuadores%20eléctricos.](https://guide.directindustry.com/es/que-actuador-elegir/#:~:text=Los%20actuadores%20se%20dividen%20en,a%20presión)%20y%20actuadores%20eléctricos.)
- Dynapar. (31 de Marzo de 2019). Obtenido de <https://www.dynaparencoders.com.br/blog/es/ethernet-ip-caracteristicas-que-y-diferencias-topologia-y-mas/>
- ElectricalChile. (11 de Octubre de 2020). Obtenido de <https://www.electricalchile.cl/plclogosiemens10.php>
- EnagoAcademy. (22 de Julio de 2021). Obtenido de <https://www.enago.com/es/academy/choose-best-research-methodology/#:~:text=Las%20metodolog%C3%ADas%20de%20investigaci%C3%B3n%20pueden,que%20interesan%20a%20la%20investigaci%C3%B3n.>
- Especificar. (16 de Febrero de 2019). Obtenido de <https://especificarmag.com.mx/todo-sobre-los-actuadores-html/>

- FactoryIO. (22 de Septiembre de 2020). *FactoryIO*. Obtenido de <https://docs.factoryio.com>
- FinderNet. (02 de Octubre de 2019). Obtenido de <https://www.findernet.com/es/mexico/news/tipos-de-contactos/>
- Flix. (31 de Julio de 2021). Obtenido de <https://www.flix-instrumentacion.com/blog/los-distintos-tipos-de-actuadores-de-control>
- Gandhi, M. (05 de Febrero de 2019). Obtenido de <https://www.autycom.com/protocolos-de-comunicacion-industrial/>
- Gil, F. (22 de Abril de 2021). Obtenido de [https://www.sicma21.com/que-son-las-redes-de-comunicacion-industrial/#Como\\_se\\_realizan\\_las\\_redes\\_de\\_comunicacion\\_industrial](https://www.sicma21.com/que-son-las-redes-de-comunicacion-industrial/#Como_se_realizan_las_redes_de_comunicacion_industrial)
- GNScomponents. (05 de Mayo de 2019). Obtenido de <http://m.es.led-diode.com/info/relay-definition-and-classification-40954598.html>
- Guevara, G. (3 de Julio de 2020). *Recimundo*. Obtenido de <http://recimundo.com/index.php/es/article/view/860>
- Gútiérrez, I. (12 de Diciembre de 2019). Obtenido de [https://programacionsiemens.com/1-introduccion-la-programacion-en-step-7/#Programacion\\_en\\_Step\\_7](https://programacionsiemens.com/1-introduccion-la-programacion-en-step-7/#Programacion_en_Step_7)
- HBM. (20 de Mayo de 2018). *HBK Company*. Obtenido de <https://www.hbm.com/es/7646/que-es-un-sensor-de-presion/>
- Hurtado, J. (21 de Abril de 2017). Obtenido de [http://www.infoplcn.net/files/documentacion/comunicaciones/infoPLC\\_net\\_introduccion-a-las-redes-de-comunicacion-industrial.pdf](http://www.infoplcn.net/files/documentacion/comunicaciones/infoPLC_net_introduccion-a-las-redes-de-comunicacion-industrial.pdf)
- Hydac. (27 de Enero de 2019). Obtenido de <https://www.hydac.com/shop/es-es/sensores/sensores-de-presion>
- INCIBE. (16 de Febrero de 2018). Obtenido de <https://www.incibe-cert.es/blog/caracteristicas-y-seguridad-profinet>
- IndustriasGSL. (19 de Febrero de 2019). Obtenido de <https://industriasgsl.com/blogs/automatizacion/arquitectura-del-plc#:~:text=La%20arquitectura%20del%20PLC%2C%20est%C3%A1,y%20de%20fuente%20de%20alimentaci%C3%B3n.>
- LogicBus. (11 de Abril de 2019). Obtenido de <https://www.logicbus.com.mx/blog/modbus-rtu/>
- LogicBus. (14 de Julio de 2020). Obtenido de <https://www.logicbus.com.mx/automatizacion.php>
- Lopez, E. (11 de Diciembre de 2017). Obtenido de [https://www.researchgate.net/figure/Figura-1-Estructura-del-sistema-automatizado-Tractiva-41\\_fig1\\_328973690](https://www.researchgate.net/figure/Figura-1-Estructura-del-sistema-automatizado-Tractiva-41_fig1_328973690)

- Lugo, E. (11 de Marzo de 2020). Obtenido de <https://electronicalugo.com/clasificacion-de-reles/>
- Margulis, E. (22 de Noviembre de 2018). Obtenido de <https://support.unitronics.com/index.php?/selfhelp/view-article/Welcome-to-U90-Ladder#:~:text=U90%20Ladder%20is%20the%20software,Program%20Editors>
- Marín, J. (21 de Agosto de 2018). Obtenido de <https://idus.us.es/bitstream/handle/11441/83982/TFG-1778-MARIN.pdf?sequence=1&isAllowed=y>
- MasterPLC. (5 de Diciembre de 2018). Obtenido de <https://masterplc.com/programacion/compuertas-logicas-en-el-plc/>
- MecaFenix. (19 de Octubre de 2018). Obtenido de <https://www.ingmecafenix.com/como-funciona/que-es-y-como-funciona-el-sensor-de-nivel/>
- MecaFenix. (s.f.). *Ingeniería MecaFenix*. Obtenido de <https://www.ingmecafenix.com/como-funciona/que-es-y-como-funciona-el-sensor-de-nivel/>
- MHJTools. (05 de Abril de 2019). *MHJTools*. Obtenido de <https://www.mhj-tools.com/?page=factory-io>
- Microsoft. (27 de abril de 2020). Obtenido de <https://docs.microsoft.com/es-es/dotnet/csharp/tour-of-csharp/>
- Mozilla, D. (14 de mayo de 2019). Obtenido de [https://developer.mozilla.org/es/docs/Learn/Getting\\_started\\_with\\_the\\_web/JavaScript\\_basics](https://developer.mozilla.org/es/docs/Learn/Getting_started_with_the_web/JavaScript_basics)
- Omega. (28 de Agosto de 2019). *Omega Engineering*. Obtenido de <https://es.omega.com/prodinfo/sondas-de-nivel-medicion.html>
- Ortego, D. (28 de Marzo de 2019). Obtenido de <https://openwebinars.net/blog/que-es-c-introduccion/>
- Palau. (09 de Junio de 2018). *S&P*. Obtenido de [https://www.solerpalau.com/es-es/blog/sensor-temperatura/#:~:text=Los%20sensores%20temperatura%20son%20dispositivos,agua%20caliente%20sanitaria%20\(ACS\).](https://www.solerpalau.com/es-es/blog/sensor-temperatura/#:~:text=Los%20sensores%20temperatura%20son%20dispositivos,agua%20caliente%20sanitaria%20(ACS).)
- PCEInstruments. (14 de Julio de 2020). Obtenido de <https://www.pce-iberica.es/instrumentos-de-medida/sistemas/sensores-vibracion.htm>
- PepperRL. (02 de Noviembre de 2019). Obtenido de [https://www.pepperl-fuchs.com/global/es/classid\\_142.htm](https://www.pepperl-fuchs.com/global/es/classid_142.htm)
- RedHat. (29 de Enero de 2018). Obtenido de <https://www.redhat.com/es/topics/automation/whats-it-automation>

- Ripipsa. (22 de Noviembre de 2021). Obtenido de <https://ripipsacobots.com/empresa-de-automatizacion-industrial/>
- Robledano, A. (22 de Julio de 2019). Obtenido de <https://openwebinars.net/blog/que-es-cpp/>
- Robledano, A. (18 de Junio de 2019). Obtenido de <https://openwebinars.net/blog/que-es-tcpip/>
- Rojas, D. (14 de Enero de 2018). Obtenido de <https://davidrojasplc.files.wordpress.com/2009/01/17.jpg>
- Santander. (17 de Julio de 2020). Obtenido de <https://www.becas-santander.com/es/blog/python-que-es.html#:~:text=Python%20es%20un%20lenguaje%20sencillo,permite%20desarrollar%20software%20sin%20límites.>
- ServiceNow. (13 de Febrero de 2020). Obtenido de <https://www.servicenow.com/es/now-platform/what-is-automation.html>
- Siemens. (25 de Marzo de 2018). Obtenido de <https://new.siemens.com/global/en/products/automation/systems/motion-control/simotion-software/scout.html>
- Siemens. (15 de Febrero de 2018). Obtenido de <https://new.siemens.com/global/en/products/automation/systems/industrial/plc/logo/logo-software.html#:~:text=Description,LOGO!,mode%20and%20in%20network%20mode.>
- Siemens. (07 de Agosto de 2019). Obtenido de <https://support.industry.siemens.com/cs/document/65601780/tia-portal-an-overview-of-the-most-important-documents-and-links-controller?dti=0&lc=en-EC>
- Siemens. (18 de Junio de 2020). Obtenido de <https://new.siemens.com/global/en/products/automation/industry-software/automation-software/tia-portal.html#:~:text=TIA%20Portal%20lets%20you%20integrate,motion%20control%2C%20and%20power%20distribution.>
- UNLP. (16 de Agosto de 2018). Obtenido de <https://www.educacionurbana.com/apuntes/ladder.pdf>
- UVa. (11 de diciembre de 2020). *Escuela de Ingenierías Industriales* . Obtenido de [https://www2.eii.uva.es/fund\\_inf/cpp/temas/1\\_introduccion/introduccion.html#:~:text=C%2B%2B%20es%20un%20lenguaje%20compilado,a%20objetos%20del%20lenguaje%20C.&text=C%2B%2B%20significaría%20incremento%20de%20C,operador%20%2B%2B%20con%20ese%20nombre.](https://www2.eii.uva.es/fund_inf/cpp/temas/1_introduccion/introduccion.html#:~:text=C%2B%2B%20es%20un%20lenguaje%20compilado,a%20objetos%20del%20lenguaje%20C.&text=C%2B%2B%20significaría%20incremento%20de%20C,operador%20%2B%2B%20con%20ese%20nombre.)



Valencia, U. d. (11 de Octubre de 2018). Obtenido de  
<https://www.uv.es/rosado/courses/CINS/IntroWinCC.pdf>

Valencia, U. d. (10 de Octubre de 2019). Obtenido de  
[https://www.uv.es/rosado/courses/sid/Capitulo3\\_rev0.pdf](https://www.uv.es/rosado/courses/sid/Capitulo3_rev0.pdf)

## Anexos

### Anexo 1



**La Libertad, 09 de noviembre del 2022**

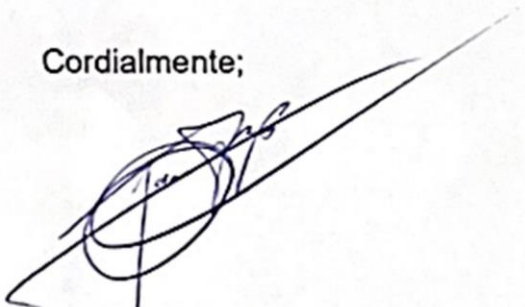
**Señor.**

Franklin Calle Asencio

Estudiante de la Universidad Católica Santiago de Guayaquil

De acuerdo con la solicitud enviada por usted le informamos que la Agencia Sucursal ENETSA La Libertad está presta a colaborar con su investigación dando Información General de procesos operativos actuales.

Cordialmente;



Fabricio Ramírez

Administrador de Agencia Sucursal

## Anexo 2

**La Libertad, 08 de noviembre del 2022**

**Señor**

**Fabrizio Ramírez**

**Administrador de Agencia Sucursal ENETSA La Libertad**

**Yo, Franklin Paul Calle Asencio, identificado con CI N°2450683848 Ante Ud. respetuosamente me presento y expongo:**

Que, habiendo culminado la carrera profesional de **Ing. Electrónica en Control y Automatismo** en la Universidad Católica Santiago de Guayaquil, solicito a Ud. permiso para realizar un trabajo de Investigación en la Agencia Sucursal ENETSA La Libertad sobre Descripción de términos operativos actuales en la repartición de paquetes para definición de necesidades en la implementación de una banda clasificadora automática para realización de tesis **"Simulación de banda clasificadora de Courier, utilizando Tia Portal de Siemens y Factory IO que controlarán actuadores y motores con PLC's, para evitar pérdidas económicas en sus clientes"** y optar el grado de Ingeniero Electrónico en Control y Automatismo.

Por lo expuesto,  
Ruego a usted acceder a mi solicitud.



---

**Franklin Calle Asencio**  
**CI:2450683848**

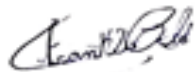
## DECLARACIÓN Y AUTORIZACIÓN

Yo, **Calle Asencio, Franklin Paul** con C.C: # 2450683848 autor del Trabajo de Titulación: **Simulación de banda clasificadora de Courier, utilizando Factory IO y Tia Portal de Siemens que controlarán actuadores y motores con PLC's, para evitar pérdidas económicas en sus clientes**, previo a la obtención del título de **INGENIERO ELÉCTRONICO EN CONTROL Y AUTOMATISMO** en la Universidad Católica de Santiago de Guayaquil.

1.- Declaro tener pleno conocimiento de la obligación que tienen las instituciones de educación superior, de conformidad con el Artículo 144 de la Ley Orgánica de Educación Superior, de entregar a la SENESCYT en formato digital una copia del referido trabajo de titulación para que sea integrado al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor.

2.- Autorizo a la SENESCYT a tener una copia del referido trabajo de titulación, con el propósito de generar un repositorio que democratice la información, respetando las políticas de propiedad intelectual vigentes.

Guayaquil, 12 de febrero del 2023



---

Nombre: Calle Asencio Franklin Paul  
C.C: 2450683848



**REPOSITORIO NACIONAL DE CIENCIA Y TECNOLOGÍA**

**FICHA DE REGISTRO DE TESIS/TRABAJO DE TITULACIÓN**

<b>TEMA Y SUBTEMA</b>	Simulación de banda clasificadora de Courier, utilizando Factory IO y Tia Portal de Siemens que controlarán actuadores y motores con PLC's, para evitar pérdidas económicas en sus clientes.		
<b>AUTOR</b>	Calle Asencio, Franklin Paul		
<b>REVISOR/TUTOR</b>	Ing. Suárez Murillo, Efraín Oswaldo		
<b>INSTITUCIÓN:</b>	Universidad Católica de Santiago de Guayaquil		
<b>FACULTAD:</b>	Facultad de Educación Técnica para el Desarrollo		
<b>CARRERA:</b>	Ingeniería Electrónica en Control y Automatismo		
<b>TÍTULO OBTENIDO:</b>	Ingeniero Electrónico en Control y Automatismo		
<b>FECHA DE PUBLICACIÓN:</b>	12 de febrero del 2023	<b>No. DE PÁGINAS:</b>	77
<b>ÁREAS TEMÁTICAS:</b>	Automatización, Comunicación industrial		
<b>PALABRAS CLAVES:</b>	Clasificadora, Software de programación, PLC, TIA PORTAL, Siemens, Factory I/O.		
<b>RESUMEN:</b>	<p>En el presente documento se realiza la simulación de una línea clasificadora de paquetes sin la utilización excesiva de recursos, se propone integrarlas en agencias sucursales de Courier provinciales, se implementará dos softwares, uno de programación y simulación de PLC virtual llamado TIA Portal de Siemens, el segundo software es dedicado a la simulación gráfica y de compatibilidad con PLC's virtuales, es llamado Factory I/O. Estos programas simulan la comunicación con óptimos protocolos con la finalidad de reducir tiempos y agilizar el proceso, todo será controlado por un tablero virtual de operador graficado en el software Factory I/O, este va a tener los controles suficientes e información necesaria para el operador. Al tener una línea de clasificación menos manual y sistemática, se podrá reducir los tiempos de entrega dentro de las sucursales de empresas Courier.</p>		
<b>ADJUNTO PDF:</b>	SI	No	
<b>CONTACTO CON AUTOR</b>	Teléfono: +593 999467226	Email: fcalle98@hotmail.com	
<b>CONTACTO CON LA INSTITUCIÓN (COORDINADOR DEL PROCESO UTE):</b>	Nombre: Bayardo Bohórquez Escobar		
	Teléfono: +593-9-95147293		
	E-mail: celso.bohorquez@cu.ucsg.edu.ec		
<b>SECCIÓN PARA USO DE LA BIBLIOTECA</b>			
<b>No . DE REGISTRO (en basea datos):</b>			
<b>No . DE CLASIFICACIÓN:</b>			
<b>DIRECCIÓN URL (tesis en la web):</b>			