UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA
SEDE CONCEPCIÓN – REY BALDUINO DE BÉLGICA



UNIVERSIDAD TECNICA
FEDERICO SANTA MARIA

Simulación de Almacén Logístico Automatizado con PLC Virtual y Monitoreo Remoto

Memoria de titulación para optar al Título de Ingeniero de Ejecución en Control e Instrumentación Industrial.

Autor:
Daniel Isaac Opazo Rodríguez

Profesor guía:
Ricardo Vera González

2026

CONSTANCIA DE VALIDACIÓN Y CONFIDENCIALIDAD DE MONOGRAFÍA A REPOSITORIO ACADÉMICO

1.- IDENTIFICACIÓN DEL TRABAJO ACADÉMICO

Tipo de monografía (marcar una opción): ☒ Memoria o trabajo de título ☐ Tesis de Postgrado

Título del trabajo: Simulación de Almacén Logístico Automatizado con PLC virtual y Monitoreo Remoto

Nombre del candidato(a): Daniel Isaac Oparo Rodríguez.

Carrera / Grado: Ingeniería de Ejecución en Control e Instrumentación Industrial

Campus: Concepción **Departamento:** Electrónica e Informática

2.- VALIDACIÓN DEL PROFESOR GUÍA/DIRECTOR DE TESIS

Yo, Ricardo Ignacio Vera González, en mi calidad de profesor(a) guía/director(a) del trabajo académico mencionado anteriormente **DEJO CONSTANCIA** que:

- He revisado esta versión del documento y corresponde a la versión final aprobada del trabajo.
- El trabajo cumple con los requisitos académicos y de formato establecidos por la institución.

3.- EVALUACIÓN DE CONFIDENCIALIDAD POR PROPIEDAD INDUSTRIAL (marcar una opción)

☒ El trabajo **NO contiene** información que amerite confidencialidad y puede ser publicado de inmediato en repositorio con acceso abierto.

☐ El trabajo **CONTIENE** información con potenciales implicancias de propiedad industrial o intelectual y requiere un periodo de confidencialidad (**embargo**) por **(marcar una opción)**:

☐ 6 meses ☐ 12 meses ☐ 2 años ☐ 3 años ☐ 5 años ☐ 10 años

Fundamentación de la necesidad de confidencialidad (obligatorio si se solicita embargo):

4.- FIRMAS

Profesor(a) guía o director(a) de memoria o tesis:

Fecha: 30/09/26 **Firma:**

Estudiante o Candidato(a):

Fecha: 30/09/26 **Firma:**

Este formulario debe ser insertado como página 2 de la memoria o tesis, completado y firmado por estudiante y profesor(a) antes de la entrega en portal PRISMA de Biblioteca USM.

Escaneado con CamScanner

Índice

Agradecimientos **6**

Resumen **7**

Introducción **8**

Problemática **9**

Objetivo General **10**

Objetivos Específicos **11**

Alcances del Proyecto **12**

Restricciones **13**

1. Marco Teórico **14**
1.1. Logística: Sentido de Almacenar 14
1.1.1. ¿Qué es la Logística? 14
1.1.2. Almacenamiento Logístico 14
1.1.3. Impacto del Almacenamiento de productos 15
1.2. Automatización en la logística 15
1.2.1. Almacén Convencional y Automatizado 15
1.2.2. ¿Por qué conviene automatizar un almacén? 16
1.2.3. Ejemplo de Almacén automatizado 16

2. Diseño de los procesos en el Software de Simulación Factory I/O **17**
2.1. Factory I/O 17
2.2. Elementos del software que se utilizarán para el almacén 17
2.2.1. Cintas trasportadoras (Belt Conveyor) 17
2.2.2. Transportadores de rodillos (Roller conveyor) 18
2.2.3. Sensor Difuso 18
2.2.4. Báscula Transportadora 19
2.2.5. Brazo Clasificador Pivotante 19
2.2.6. Operadores con Luces Piloto 20
2.2.7. Máquina “Pick and Place” 21
2.2.8. Stacker Crane (Transelevador) 22
2.2.9. Otros elementos 23
2.3. Diseño del Almacén 24
2.3.1. Estación de Clasificado 24
2.3.2. Estación de Paletizado 25
2.3.3. Estación de Almacenamiento y Despacho 26
2.3.4. Diseño final del Almacén 27

3. Programación en TiaPortal y Simulación en PLCSIM **28**
3.1. Plantilla de Factory IO para comunicarse con PLCSIM 28
3.2. Programación 31
3.2.1. Main [OB1] 31
3.2.2. Partir Parar [FC1] 32
3.2.3. Clasificador [FB1] 33
3.2.4. Paletizador [FB2] y [FB3] 35
3.2.5. Transelevadores [FB4] 38
3.2.6. Bloque de Variables Globales "VAR" [DB1] 40
3.3. Conexión de PLCSIM y Factory I/O 42

4. Monitoreo y Control Remoto **44**
4.1. Conexión entre Node-Red y PLCSIM Standard 44
4.1.1. Uso de NetToPLCsim 44
4.1.2. Lectura de variables con Node red 45
4.2. Servicio de Azure IoT Central 47
4.2.1. Creación de una plantilla para dispositivo 47
4.2.2. Visualización de datos en la Aplicación 48
4.3. Monitoreo y control utilizando Dashboard de Node Red con una VPN 49
4.3.1. Confección de Dashboard en Node red 49
4.3.2. Acceso remoto con una VPN 51

5. Resultados y Conclusiones **54**
5.1. Resultados 54
5.2. Conclusiones 55

6. Anexos **56**
6.1. Driver Factory I/O 56
6.2. Fragmentos de Programa en TiaPortal 58
6.2.1. PARTIR_PARAR [FC1] 59
6.2.2. PALETIZADOR_M [FB3] 60
6.2.3. TRANSELEVADORES [FB4] 65
6.3. Flujo Node Red 70

7. Referencias **73**

Índice de figuras

2.1. Belt Conveyor . . . 17
2.2. Roller Conveyor . . . 18
2.3. Sensor Difuso . . . 18
2.4. Báscula Trasportadora . . . 19
2.5. Brazo Clasificador Pivotante . . . 19
2.6. Operadores con luces piloto . . . 20
2.7. Máquina Pick and Place 1 . . . 21
2.8. Pick and place 2 . . . 21
2.9. Transelevador . . . 22
2.10. Horquilla accionada . . . 23
2.11. Estación de Clasificación . . . 24
2.12. Estacion de paletizado de cajas medianas . . . 25
2.13. Estación de almacenamiento . . . 26
2.14. Estación de almacenamiento 2 . . . 26
2.15. Diseño final de Almacén. Vista frontal . . . 27
2.16. Diseño final de Almacén. Vista lateral . . . 27
3.1. Habilitación de comunicación PUT/GET . . . 28
3.2. Bloque FC9000 . . . 29
3.3. Módulos de entradas . . . 29
3.4. Configuracion en Factory IO . . . 30
3.5. Árbol del proyecto . . . 31
3.6. Network 1 y Operadores . . . 32
3.7. FB1 LLAMADO EN MAIN . . . 33
3.8. Net. 1 CLASIFICADOR . . . 34
3.9. Net. 2 CLASIFICADOR . . . 34
3.10. Net.3 CLASIFICADOR . . . 34
3.11. Net. 4 CLASIFICADOR . . . 34
3.12. FB3 y FC2 LLAMADO EN MAIN . . . 35
3.13. Contador de Pasos (Network 1 [FB2]) . . . 36
3.14. Ejecución Paso 1 (Network 2 [FB2]) . . . 36
3.15. Ejecución Paso 2 (Network 3 [FB2]) . . . 37
3.16. Ejecución Paso 3 y 4 (Network 3 [FB2]) . . . 37
3.17. Paletizador Cajas Medianas . . . 37
3.18. Paletizador Cajas Grandes . . . 37
3.19. [FB4] Llamado a Network 5 . . . 38
3.20. [FB4] Contador de pasos . . . 39
3.21. Contador de Posición. . . . 39
3.22. Ejemplo ejecución de Paso 1, 2 y 3 . . . 40
3.23. VAR [DB1] . . . 41
3.24. Iniciar simulación . . . 42
3.25. Cargar a la instancia . . . 42
3.26. Almacen controlado por PLCSIM . . . 43
4.1. Selección de tarjeta de red . . . 44

4.2. Selección de instancia virtual de PLCSIM 45
4.3. Iniciar servidor 45
4.4. Configuración de end point Node-Red 46
4.5. Propiedades DB1. 46
4.6. Lectura de Variables con Node red 47
4.7. Creación de Datos en Azure IoT Central 47
4.8. Edición de la Plantilla 48
4.9. IoT Central desde Móvil 48
4.10. Nombre Panel General. 49
4.11. Creación de Grupo 49
4.12. Programación de grupo estado 50
4.13. Flujo de Dashboard 50
4.14. Dashboard final desde el PC 51
4.15. Configuración de Red en ZeroTier 52
4.16. Añadir red en PC 52
4.17. Añadir red en Android 52
4.18. Dispositivos miembros autorizados 53
4.19. Panel de operación desde Android 53
6.1. Entradas y salidas Factory I/O 56
6.2. Entradas y salidas Factory I/O P.2 57
6.3. HMI 58
6.4. 59
6.5. 59
6.6. Net. 4 FB4 67
6.7. Net. 5 FB4 68
6.8. Librerías instaladas en NodeRed 71

Agradecimientos

Deseo expresar mi agradecimiento, primeramente, a mi Dios y Señor Jesucristo, por absolutamente todo, por ayudarme en todas las cosas, protegerme a lo largo de toda mi vida, darme el entendimiento para salir adelante en esta carrera, proveyéndome todas las herramientas y colocando a las personas correctas en mi camino, las cuales me han ayudado en este proceso.

Agradezco a mi familia, en especial mi padre y madre, por apoyarme incondicionalmente, consolándome en momentos de derrota, y felicitándome en momentos de triunfo, educarme, enseñándome a perseverar y a no rendirme. Igualmente, mi hermana Esperanza, mi cuñado Roger y mi sobrinito David, por motivarme y alegrarme los días cuando los veo.

Por último, agradezco a don Ricardo Vera González, mi profesor guía, por su ayuda y orientación en la confección de esta memoria, por su excelencia a la hora de ejercer la docencia, con vocación para la enseñanza y siempre con disposición de ayudar a sus estudiantes.

Resumen

En la presente memoria de titulación se documenta el camino para realizar una simulación en el software llamado Factory I/O de un almacén logístico automatizado por un controlador lógico programable (PLC) virtual, del cual, se pueda monitorear variables y controlarlas a distancia. Se abarcará el proceso de diseño de la maqueta virtual, programación del dispositivo autómata virtual, métodos para lectura y escritura de variables, así como una breve explicación de las aplicaciones y softwares utilizados.

Introducción

Hoy en día, la automatización industrial ha sido la respuesta idónea ante la búsqueda de aumentar la productividad, seguridad y disminuir costos. Gracias a la integración de nuevas tecnologías se permiten realizar trabajos de manera rápida, precisa, segura y a mayor escala, lo que ha ido tomando lugar en muchas áreas, como la fabricación de diversos objetos, producción alimenticia, etc.

El almacenamiento logístico no es la excepción ante esta evolución, ya que, es parte fundamental en cualquier empresa, por lo que no conviene que sea el punto débil. Un almacén automatizado presenta muchas ventajas y soluciona muchos problemas que conlleva uno convencional, como lo veremos a continuación en el presente informe.

Sin embargo, antes de llevar a cabo cualquier modernización industrial se debe planificar con cuidado cada detalle, una de las prácticas más comunes es realizar una simulación, la cual es una imitación del desempeño de los procesos o de sistemas del mundo real. Es habitual el uso de softwares o programas de simulación 3D, los cuales son intuitivos para crear un escenario visual de un proceso. Este proyecto está centrado en el desafío de armar una simulación desde cero, automatizada, poniendo en práctica conocimientos de programacion de PLC's con la herramienta "TiaPortal" de Siemens y aún más allá, el poder monitorear y controlar a distancia.

Problemática

Un almacén logístico convencional o no automatizado depende principalmente del trabajo manual de los trabajadores, lo que significa que pueden surgir problemas, por una parte, para los trabajadores, que muchas veces deben realizar levantamientos de carga pesada si no se cuenta con montacargas, lo que puede provocar daños, accidentes, lesiones, etc. Por otra parte, si se comente un error en la manipulación de carga puede traer problemas para la empresa, ya que, se pueden dañar los productos, infraestructura, provocando pérdidas monetarias. Además, en un almacén se debe administrar bien el espacio, debe estar ordenado y transitable, lo que no siempre se logra.

Objetivo General

Diseñar y simular un sistema de almacén automatizado en el software Factory IO controlado con un PLC virtual, que sea capaz de clasificar, contar, almacenar y despachar el material que llega a un almacén de logística, además, de que se pueda monitorear el stock y los despachos a través de internet, todo esto con el motivo de aumentar la eficiencia, optimizar mejor el espacio, reducir los errores humanos que signifiquen daños a trabajadores o a los productos, y controlar el inventario.

Objetivos Específicos

I. Diseñar los procesos en el Software Factory IO.

II. Realizar la programación para un PLC virtual con sofware TIA PORTAL.

III. Integrar el PLC programado junto con el escenario virtual de Factory IO.

IV. Monitorear el proceso a distancia utilizando Node-Red.

Alcances del Proyecto

I. **Simulación de un almacén automatizado capaz de clasificar por peso dos tipos de productos (cajas), paletizarlos, almacenarlos y despacharlos. Para lograr esto se debe realizar:**

   A. **Diseño general del Proceso.**

      Se diseñará el almacén en el software de simulación 3D, con el fin de delimitar y optimizar el espacio de cada uno de los subprocesos. Utilizando lo necesario de los elementos que proporciona el software, como sensores, actuadores, cintas transportadoras, estanterías, etc.

   B. **Automatización del Proceso.**

      Se programará un PLC virtual que permitirá automatizar los subprocesos diseñados en Factory IO para que funcionen de manera correcta, autónoma y en conjunto, sin errores, siendo capaz de simular procesos logísticos (clasificar, contar, paletizar, almacenar y despachar los productos).

   C. **Simulación del Almecén.**

      Se integrará lo hecho en Factory IO con el programa para el PLC virtual que controle el proceso, se simulará la entrada de dos tipos de productos de manera aleatoria, el sistema debe clasificar, contar, paletizar, almacenar y despachar dichos productos.

II. **Monitoreo a través de internet de la cantidad de productos almacenados y despachados en el almacén.**

   A. **Envío de datos a aplicación de la Nube.**

      Los datos del PLC que se vayan generando con la simulación, como el conteo de cada producto, serán enviados a través de Node-Red a una Aplicación en la Nube de Azure, cada vez que se esté simulando el proceso, lo que permitirá el monitoreo del stock disponible desde cualquier parte del mundo mientras se tenga acceso a internet.

   B. **Monitoreo y control a través de internet (VPN)**

      Confecionar una Dashboard con Node-red para el control y monitoreo a distancia.

Restricciones

I. **Licencias de software requeridas:**

- **Tia Portal V18:** Es el software de Siemens que permite programar una amplia variedad de modelos PLCs, HMI, etc. Cuenta con diversas herramientas, diferentes leguajes de programación, ideal para automatizar un proceso. (Licencia gratuita los primeros 30 días Precio de licencia, 397,98 € es decir 446mil CLP aprox. Incluye PLCSIM).
- **PLCSIM V18:** Es el Software de Siemens que permite la simulación de un PLC virtual, ideal para probar los programas si no se cuenta con un PLC físico. (Requiere la licencia de TIA Portal).
- **Factory I/O versión 2.5.10:** Es el software de simulación industrial 3D, que permite el diseño de escenarios industriales, los cuales pueden ser controlados por PLC real o virtual. (Requiere Licencia, 31.406 CLP al mes, los primeros 30 días son gratis).
- **Node-Red:** Permite conectar y controlar fácilmente dispositivos de hardware, como sensores o actuadores en un proyecto de IoT. También permite enviar datos de hardware a servicios en la Nube. (Código abierto).

II. **Servicio o aplicación de la Nube:** Cada aplicación de Azure tiene un costo específico que depende de cuánto tiempo estén en funcionamiento. Para una cuenta normal hay un crédito de 200 UDS los primeros 30 días, en cambio, una cuenta con membresía "Azure for Students" cuenta con 100 UDS que expira en un año.

III. **Limitaciones del Software Factory I/O:**

- Cuando se utiliza con PLCSIM como controlador, la configuración de Factory IO tiene como límite 64 Bytes tanto para las salidas como las entradas. Es decir que se reduce el número de entradas y salidas si son datos reales, ya que éstas son de tipo DWORD (Que ocupan 4 bytes).
- Genera un número limitado de elementos en una simulación (Cajas, palets, piezas), una vez alcanzado ese límite no se generan más.
- Requiere una plantilla específica de programa de TIA Portal para conectarse con un PLC virtual de PLCSIM.
- Las cajas tienden a desordenarse si están en movimiento.
- Rara vez se generan movimientos aleatorios de algunas máquinas o cajas.

1. Marco Teórico

1.1. Logística: Sentido de Almacenar

1.1.1. ¿Qué es la Logística?

La logística empresarial es la diciplina encargada de planificar, gestionar y controlar el flujo de los bienes, los servicios y de la información que posee una empresa, con el propósito de que fluyan de manera eficiente hasta un destino o consumidor final. El objetivo principal de la logística es garantizar la calidad de los productos y/o servicios que se brindan, lo cual es una ventaja competitiva que mejora la confiabilidad y satisfacción al cliente, también tiene como objetivo disminuir costos, una buena planificación reduce costes de transporte y manipulación, lo que también ayuda a mejorar la rentabilidad de la compañía, y agilizar el funcionamiento de la empresa, lo que va a permitir que responda de manera rápida ante las demandas del mercado.

1.1.2. Almacenamiento Logístico

Una de las etapas más importantes de la logística empresarial es el almacenamiento de los productos, es el eje central del proceso y no sólo se trata de recibir el producto y guardarlo, sino que debe ser almacenado estratégicamente en el lugar correcto para mantener la calidad durante el tiempo guardado, y debe estar disponible para ser localizado, posteriormente prepararlo para el despacho, para así responder ante la demanda. El lugar destinado para esto es el almacén o también llamado bodega, que debe cumplir cinco principales funciones.

- **Recepción de mercaderías:** se refiere a la entrada de los productos desde los proveedores, aquí se comprueba si la cantidad, calidad, características de los materiales coinciden con la información de la guía de entrega.
- **Almacenamiento:** se trata de situar los productos en un lugar estratégico de la bodega, se deben agrupar siguiendo ciertos criterios de clasificación (tamaño, peso, naturaleza del producto), con el fin de poder localizarlos de manera sencilla y poder acceder a ellos de la manera más rápida y segura posible.
- **Conservación y mantenimiento:** se refiere a la conservación en perfecto estado de los productos mientras estén almacenados, esto va a depender de las características de la mercancía, si es frágil, peligroso, existen legislaciones vigentes sobre la higiene y seguridad del almacén y normas sobre el cuidado de cada tipo de producto.
- **Gestión y control de existencias:** se trata de controlar la cantidad de productos que se debe almacenar y la cantidad de productos que se deben solicitar a los proveedores. Esto según la demanda en el mercado, lo ideal es que los productos no sean almacenados durante mucho tiempo, debido a que se pierde espacio en el almacén, se degradan los productos y se pierde productividad.
- **Despacho de los productos:** es la etapa final del almacenamiento que consiste en preparar el producto para la salida del almacén, para esto se debe localizar, extraer de manera segura y por ultimo transportar a la zona de despacho.

1.1.3. Impacto del Almacenamiento de productos

La práctica de almacenar productos, si se realiza de la manera correcta, tiene un impacto positivo en la productividad, en los costos y en la satisfacción al cliente, por ejemplo, si un almacén es capaz de recibir, guardar, y despachar productos de manera rápida, eficiente y sin errores en cada uno estos procesos, será más ágil a la hora de responder ante la demanda de los clientes con lo cual disminuye los tiempos de espera, la calidad de los productos se mantiene, por lo cual no habrá pérdidas de material por daños, lo que provocará una reducción de costos, una mejor satisfacción, confiabilidad de los clientes hacia la empresa. Por otro lado, si un almacén tiene mal funcionamiento, está desorganizado, no se ocupa bien el espacio, tendrá un impacto negativo en los aspectos mencionados anteriormente.

1.2. Automatización en la logística

1.2.1. Almacén Convencional y Automatizado

Existen diferentes tipos de almacenes y distintas formas de clasificarlos, una de éstas es según el grado de automatización, aquí encontramos los almacenes convencionales y automatizados. Los almacenes convencionales son aquellos que cuentan con medios sencillos para el almacenamiento y transporte de carga, como montacargas, transpaletas, o de manera manual, depende principalmente de los bodegueros, de su grado de capacitación y de las herramientas que dispongan. La capacidad de almacenamiento es menor, ya que se requiere más espacio entre los pasillos para manipular las cargas, y el apilamiento generalmente no supera las 3 alturas.

En cambio, los almacenes automatizados se caracterizan por realizar los procesos de clasificación, almacenado y movimientos de cargas de manera automática, sin la intervención de personal, a través de maquinaria como cintas transportadoras, paletizadoras, robots, transelevadores, etc. Su principal objetivo son aumentar la eficiencia y rapidez de las operaciones, reducir los errores humanos, aumentar la seguridad y aprovechar el espacio al máximo.

1.2.2. ¿Por qué conviene automatizar un almacén?

La inversión inicial de la automatización de una bodega es costosa, sin embargo, es necesaria para las empresas que tienen un altísimo flujo de productos y mercaderías, estas compañías están obligadas a responder de manera rápida y precisa, y la solución de automatizar produce mejoras en muchos aspectos.

Uno de los beneficios es la mejora en el rendimiento general, ya que se aplica en toda la cadena logística (recepción, almacenaje y despacho de mercancías), permite disminuir significativamente los tiempos muertos, inclusive a cero, ya que, las máquinas, robots e instrumentación pueden operar 24/7. Otra ventaja es que elimina los errores en todos los procesos debido a la disminución de la intervención humana, al optar por la automatización se acepta una política de cero errores en el almacenado, picking (preparación de pedidos), procesos de inventarios, y en todos los procesos logísticos. La optimización del espacio físico también es una ventaja, elementos como los transelevadores, permiten movilizar cargas pesadas de mercadería y ubicarlas en los racks de almacenamientos que son estanterías diseñadas para almacenar mercadería en altura, gracias a esto permite una mayor capacidad de almacenamiento, aumento en la seguridad, debido a que se elimina el desorden y la desorganización.

Por último, gracias a la automatización se mejora la calidad de trabajo para los empleados, ya que se elimina el esfuerzo físico, gracias a que el trasporte de carga lo realizan las máquinas, hay mayor seguridad y trabajo más ergonómico y especializado. Un almacén automatizado, es una garantía de reducción de accidentes y enfermedades profesionales.

1.2.3. Ejemplo de Almacén automatizado

British Sugar: (Gran Bretaña e Irlanda) es una industria productora de azucar con más de un siglo de antigüedad y ha optado por automatizar su almacén de producción. Esto fue una solución para agilizar el almacenaje de sus productos terminados, ya que, tuvo un aumento significativo de producción y demanda, por lo cual, el almacenaje convencional terminaba siendo un cuello de botella en el proceso.

La compañía británica optó por confiar en Mecalux, la cual es una empresa dedicada a brindar soluciones tecnológicas en el área de logística, la cual diseñó un nuevo almacén automatizado, basado en el Sistema Pallet Shuttle (Sistema de almacenaje que utiliza carros motorizados, como transelevadores, para transportar pallets, depositarlos o extraerlos en estanterías metálicas de gran altura). Los beneficios que obtuvo British Sugar son, un almacén automatizado conectado directamente con las líneas de producción, agilidad y seguridad en la entrada y salida de 3000 pallets diarios aproximadamente. Lo que incrementó significativamente su capacidad de almacenamiento y agilidad a la hora de despachar productos.

2. Diseño de los procesos en el Software de Simulación Factory I/O

2.1. Factory I/O

Factory I/O es un simulador 3D que permite crear escenarios industriales comunes y realistas que admite controlarlos en tiempo real, se usa principalmente para el aprendizaje de tecnologías de automatización, principalmente para poner en práctica la programación de un PLC (Controladores Lógicos Programables), un escenario puede ser controlado tanto por un PLC real o virtual. Es compatible con diferentes marcas y modelos de controladores, como, Siemens, Schneider, Allen-Bradley, etc. El simulador cuenta con muchos elementos y máquinas comunes dentro de un entorno industrial como por ejemplos, cintas transportadoras (Para cargas livianas), transportadores de rodillos (Para cargas pesadas), diferentes tipos de sensores, tanques de líquidos, transelevador, etc.

2.2. Elementos del software que se utilizarán para el almacén

2.2.1. Cintas trasportadoras (Belt Conveyor)

Descripción: se utilizan para el transporte de cargas ligeras, como cajas individuales, el software dispone de cintas de 2, 4 y 6 metros, también incluye cintas curvas que presentan el mismo funcionamiento.

Funcionamiento: hay tres modos de configuración para estas cintas, digital, digital (+/-) y analógica. La configuración digital permite mover la cinta en una velocidad constante de 0,6[m/s] en un solo sentido a través de un actuador de tipo booleano, el modo digital (+/-) tiene dos actuadores booleanos, uno para cada sentido, y el modo análogo es un actuador de tipo de dato real el cual representa voltios, entre +10v y -10v, en 0v está quieto, el sentido lo determina el signo y la velocidad la amplitud del valor de voltaje.

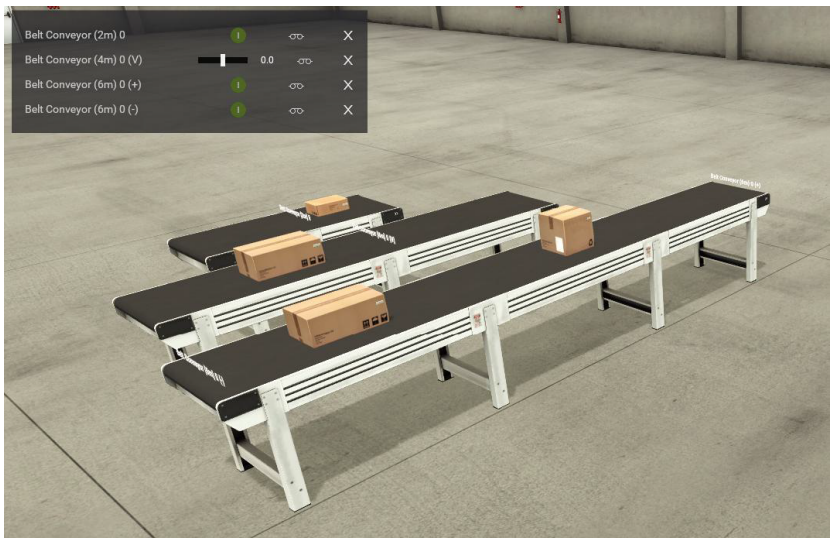


Figura 2.1: Belt Conveyor

2.2.2. Transportadores de rodillos (Roller conveyor)

Descripción: se utilizan para transportar carga pesada, como cajas agrupadas en palets, al igual que las cintas las hay de 2, 4 y 6 metros, y cuentan con el mismo funcionamiento y configuraciones que las cintas transportadoras, con la velocidad de transporte de 0,45 m/s (digital) y 0,8 m/s (analógico) como única diferencia.

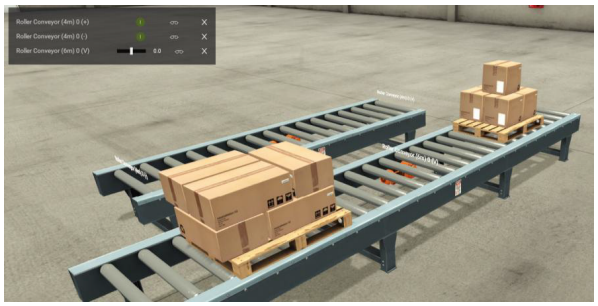


Figura 2.2: Roller Conveyor

2.2.3. Sensor Difuso

Sensor fotoeléctrico que detecta la presencia de objetos, su funcionamiento es sencillo, cuando detecta un objeto su condición es verdadera (1 lógico). A diferencia de otros sensores fotoeléctricos, el emisor y el receptor están situados en la misma carcasa, lo que simplifica su instalación.

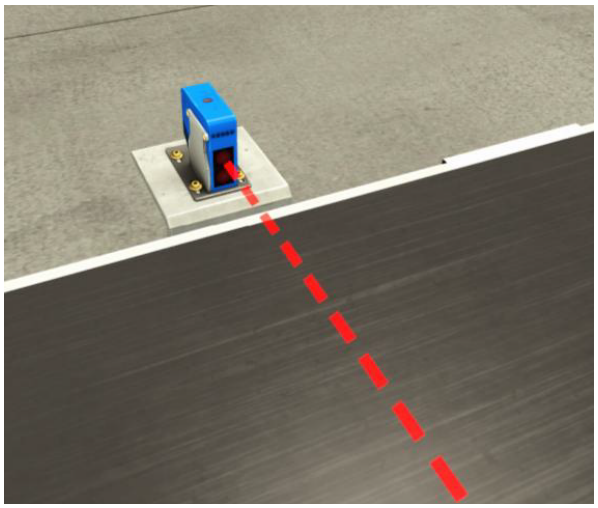


Figura 2.3: Sensor Difuso

2.2.4. Báscula Transportadora

Es una cinta trasportadora capaz de medir la masa del objeto que esté transportando. Su funcionamiento cuenta con dos actuadores booleanos que mueven la cinta en ambos sentidos, y un sensor que envía un valor análogo entre 0 a 10 voltios según la masa del objeto. La cinta debe estar quieta para una medición precisa.



Figura 2.4: Báscula Trasportadora

2.2.5. Brazo Clasificador Pivotante

Son utilizados para desviar objetos mientras son transportados en la cinta transportadora, cuenta con tres actuadores de tipo booleano, uno que efectúa el giro del brazo en 45 grados, y los otros dos determinan el sentido de la cinta del brazo. En la imagen se muestra los actuadores del giro y la banda hacia delante forzados.



Figura 2.5: Brazo Clasificador Pivotante

2.2.6. Operadores con Luces Piloto

Son pulsadores, los cuales cumplen la función de entrada, son sensores booleanos, pero también cuentan con un actuador el cual es una luz indicadora. Dependiendo el tipo de botón puede ser normalmente cerrado o abierto, los botones “Stop” y “Emergency Stop” son normalmente cerrados (Siempre están en la condición TRUE, es decir en 1 lógico).



Figura 2.6: Operadores con luces piloto

2.2.7. Máquina “Pick and Place”

Es una estación de tipo pórtico cuya función es recoger un objeto (en este caso cajas) para colocarlas en otro sitio, se mueve en tres ejes controlados por servomotores y el “gripper”(brazo o pinza) puede rotar el objeto. La pinza se activa mediante una ventosa (actuador booleano) e incluye un sensor booleano de proximidad que está activado cuando no hay caja, y otro sensor, también binario que se activa cuando detecta una caja.

El funcionamiento para el movimiento de los ejes puede controlarse mediante valores digitales y analógicos, según la configuración seleccionada. Cuando se controla en modo digital cada eje (X, Y, Z) tiene dos actuadores booleanos (+ / -), uno por dirección que mueven el brazo por pasos. El otro modo de configuración es el analógico, cada actuador de un eje es un valor real de 0v a 10v, el valor establecerá la posición del brazo.

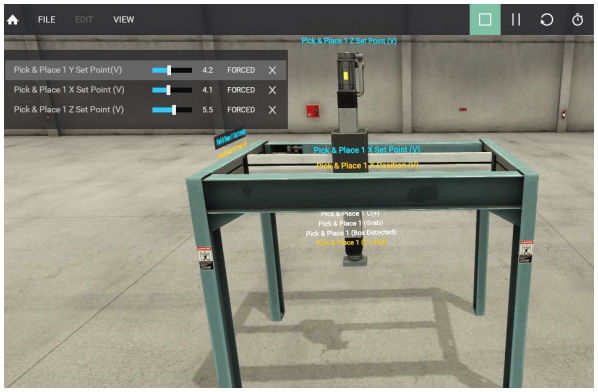


Figura 2.7: Máquina Pick and Place 1

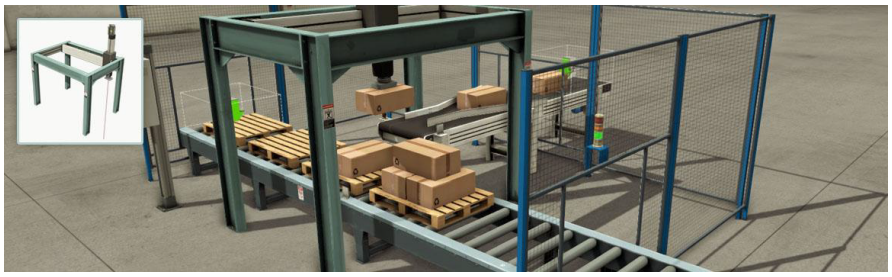


Figura 2.8: Pick and place 2

2.2.8. Stacker Crane (Transelevador)

Es la máquina cuya función es recoger cargas pesadas y almacenarlas en estanterías o racks de almacenamiento, también al revés. Se mueven de manera longitudinal y en altura, en la vida real es un pilar fundamental de los almacenes automatizados, ya que permite maximizar el uso del espacio vertical en grandes centros de distribución y almacenes, y mejorar drásticamente la velocidad y precisión de la gestión de inventario.

Datos:

- **Recorrido de la horquilla:** 1,2 metros.
- **Carrera del carro:** 10,5 metros.
- **Recorrido de la plataforma:** 6,625 metros.
- **Velocidad del carro:** 1,4 m/s.
- **Velocidad de las horquillas:** 0,5 m/s.
- **Velocidad de la plataforma:** 1,7 m/s.
- **Número máximo de posiciones:** 55.

Funcionamiento:

Se puede configurar en varios modos, pero el más conveniente y típico es el modo numérico, ya que, dispone de un actuador de tipo de dato entero (int) entre 1 y 55, cada número indica una posición en las estanterías desde la posición 1 hasta la 54, siendo el 55 la posición inicial o de de reposo. En la siguiente imagen se aprecia la máquina en la condición de reposo.

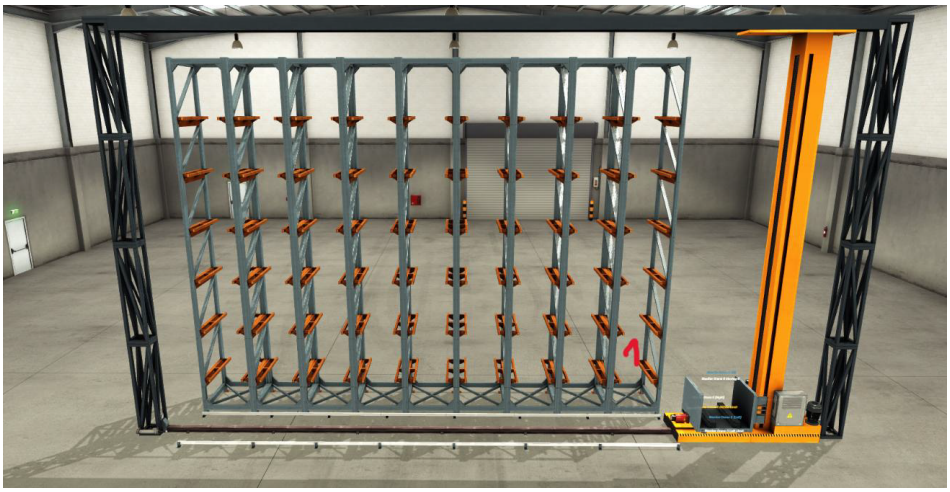


Figura 2.9: Transelevador

Cuenta con horquillas que se mueven hacia delante y atrás con el propósito de recoger o depositar un palet, el movimiento es a través de dos actuadores booleanos, uno que mueve hacia adelante y otro que hace el movimiento hacia atrás. También un actuador binario llamado “lift” se encarga de levantar o bajar levemente las horquillas para tomar o depositar el pallet.



Figura 2.10: Horquilla accionada

También la máquina cuenta con cinco sensores de tipo booleano, los cuales son:

- **Moviéndose en eje Z (Altura):** , condición TRUE indica que se está moviendo.
- **Moviéndose en X (Longitudinal):** condición TRUE indica que se está moviendo.
- **Límite medio de horquillas:** un 1 lógico indica que están en el medio.
- **Límite izquierdo de horquillas:** la condición TRUE indica que las horquillas están posicionadas en el lado izquierdo.
- **Límite derecho de horquillas:** la condición TRUE indica que las horquillas están posicionadas en el lado derecho.

2.2.9. Otros elementos

Se utilizaron también:

- **Alarmas:** Luz llamativa que indica cuando está funcionando el transelevador.
- **Desviadores:** Pequeños muros que sirven para guiar las cajas en las cintas transportadoras.
- **Emisores de objetos:** Hacen aparecer elementos, como cajas, pallets u otros. Se puede controlar con el PLC o dejar estático.
- **Removedores de objetos:** Eliminan los objetos que pasen por aquí.

2.3. Diseño del Almacén

Utilizando todos elementos descritos anteriormente se dio paso al diseño del almacén, es decir se determinó el espacio físico de cada uno se los subprocesos los cuales son la entrada y clasificación de los productos, que son dos tipos de cajas debido a las limitaciones del software (Cajas grandes y medianas, clasificadas por peso), el paletizado de éstas, y almacenamiento en las estanterías agrupadas y ubicadas en determinado lugar según su categoría.

2.3.1. Estación de Clasificado

El clasificador de cajas está compuesto principalmente por un emisor de cajas controlado por el PLC para que emita solamente cajas medianas y grandes de manera aleatoria, cintas transportadoras, una de éstas es una báscula la cual realizará el pesaje de cada caja y según su masa se desviará con el brazo clasificador pivotante a la cinta correspondiente para que se envíe al siguiente proceso.

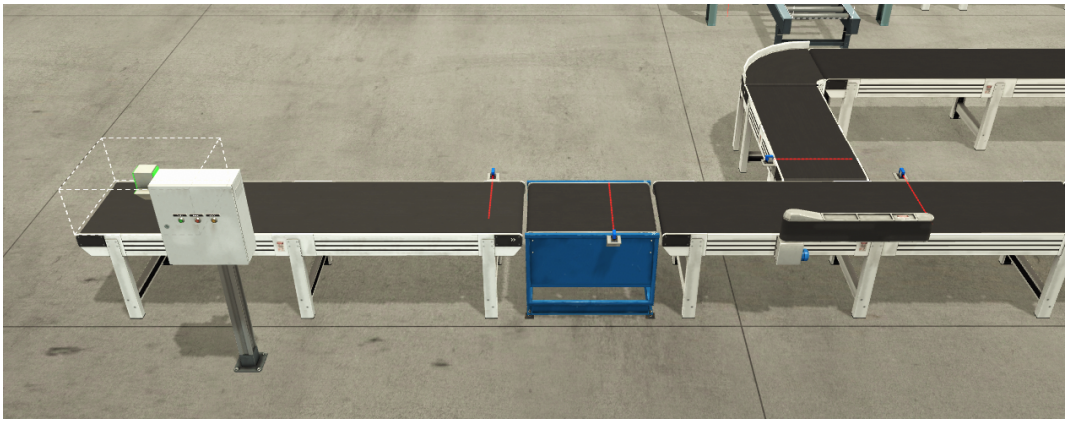


Figura 2.11: Estación de Clasificación

2.3.2. Estación de Paletizado

Para que las cajas sean almacenadas por el transelevador es necesario agruparlas de manera ordenada en un palé, es por esto que cada tipo de caja clasificada llegue a una estación con la máquina Pick & Place la cual tomará las cajas una por una y las colocará en un palé, una vez armado el palé será enviado a la siguiente estación a través de un transportador de rodillos de carga pesada.



Figura 2.12: Estacion de paletizado de cajas medianas

2.3.3. Estación de Almacenamiento y Despacho

El almacenamiento y despacho de los productos lo realizará la máquina “Stacker Crane” (Transelevador), una vez que este reciba un pallet armado lo almacenará en un espacio disponible de las estanterías. Por otro lado, para despachar, tomará algunos de los palés y lo depositará en los transportadores de rodillos destinados para la salida de los productos del almacén.

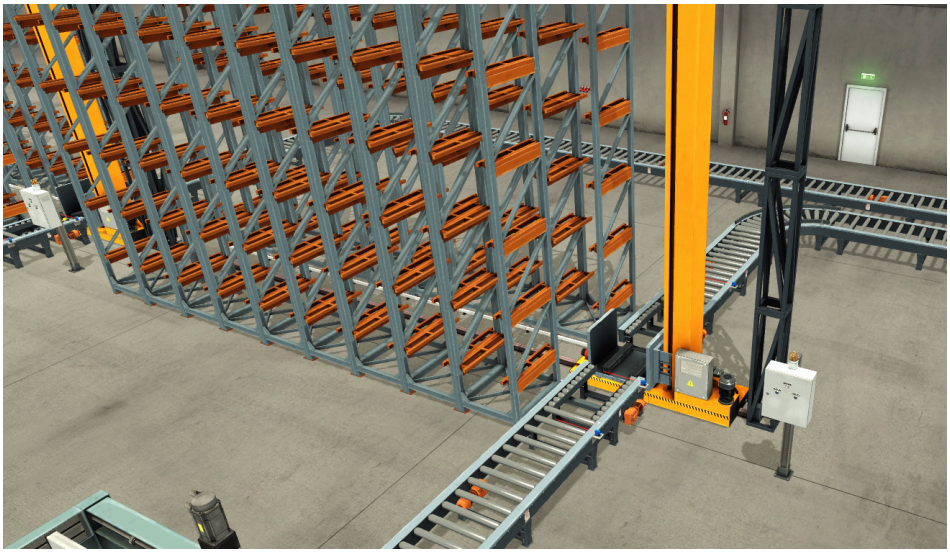


Figura 2.13: Estación de almacenamiento



Figura 2.14: Estación de almacenamiento 2

2.3.4. Diseño final del Almacén

Finalmente, como se observa en la figura 2.15 la distribución de las estaciones de trabajo del almacén quedó así, la estación de clasificado que se encargará de separar dos tipos de cajas hacia una estación de paletizado por cada una, y también un transelevador con sus respectivas estanterías por cada producto. Se puede observar que sobra espacio en el almacén, en un principio se tenía por objetivo que sean cuatro tipos de productos, pero se tuvo que descartar la idea debido a las limitaciones del software (Factory I/O), una de ellas es que a la hora de realizar una simulación se genera un número limitado de elementos (Cajas y Palets) y llegado a ese límite no se generan más, por lo que no tiene sentido destinar más espacio en estanterías para otras cajas si no se iban a aprovechar ni un 25 % de cada una. Otra limitación de Factory IO es la cantidad de Bytes destinados a las salidas entradas para un controlador simulado en PLCSIM, por esta razón tampoco se puede añadir más máquinas Pick & Place, ya que, tienen tres sensores y tres actuadores que usan datos de tipo Real (DWord).

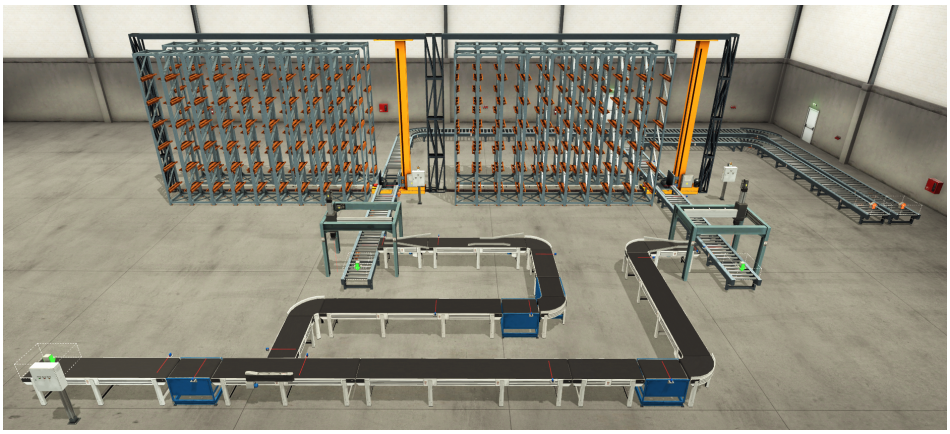


Figura 2.15: Diseño final de Almacén. Vista frontal

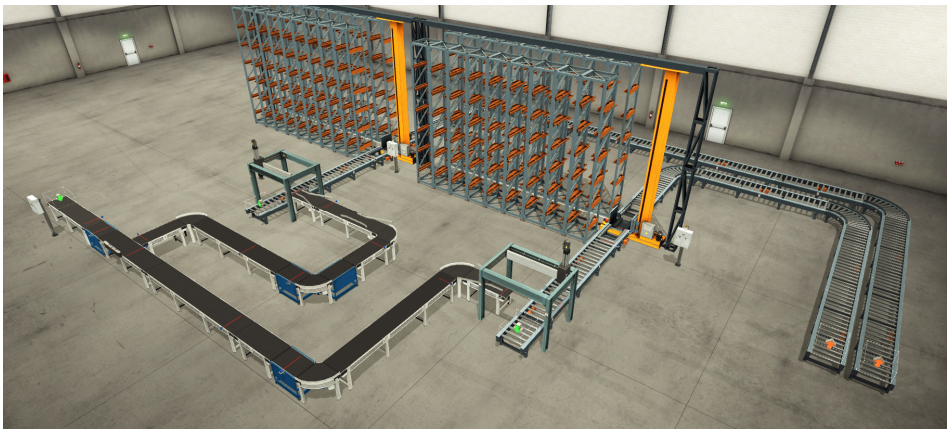


Figura 2.16: Diseño final de Almacén. Vista lateral

3. Programación en TiaPortal y Simulación en PLCSIM

La programación de los PLC's ha sido muy importante en los entornos industriales, ya que, gracias a esto es posible controlar máquinas, robots, líneas de producción y diferentes procesos para que funcionen de manera precisa, eficiente y autónoma. Una de las herramientas más populares para esta práctica es el software llamado TíaPortal (Totally Integrated Automation Portal) que es el sistema del fabricante Siemens que proporciona diversas herramientas para la configuración y programación de sus controladores. Este capítulo trata sobre la programación de el PLC simulado para que sea capaz de controlar el escenario diseñado en el capítulo anterior, se explicará el funcionamiento de los bloques más importantes del programa, así como también cómo se hace la conexión entre PLCSIM versión estándar y Factory I/O.

3.1. Plantilla de Factory IO para comunicarse con PLCSIM

Un PLC real de Siemens tanto de la familia S7-1200 y S7-1500 se puede comunicar directamente con Factory IO, siempre y cuando el PC donde está en marcha el simulador 3D esté conectado a través de ethernet con una dirección IP que se encuentre en la misma red que el controlador, y dentro de la configuración de Factory se debe especificar la familia del PLC (S7-1500 o S7-1200), la interfaz del PC donde se encuentra conectado a la red del controlador y la dirección IP de este. Esta comunicación se hace a través del protocolo ISO on TCP el cual permite que estándares industriales funcionen sobre TCP/IP.

Sin embargo, en este caso no se dispone de un PLC real, sino de uno virtual simulado en PLCSIM estándar, esta instancia no tiene una interfaz ethernet por lo que no es visible para el Simulador 3D, para lograr que Factory IO se comunique con la instancia simulada se requiere una plantilla de programa específica que se puede encontrar en sitio web oficial de Factory IO. Este programa viene configurado específicamente para lograr la deseada comunicación.

Entre los parámetros configurados más importantes está la habilitación de comunicación PUT/GET del PLC que permite que un agente externo en este caso Factory IO pueda escribir en sus entradas (por ejemplo, si en el software se detecta una caja a través de un sensor que está vinculado a una entrada virtual del PLC pueda cambiar el estado de esa entrada) y leer las salidas para que puedan actuar en la simulación. Esta casilla también debe estar marcada si se usa un PLC real.

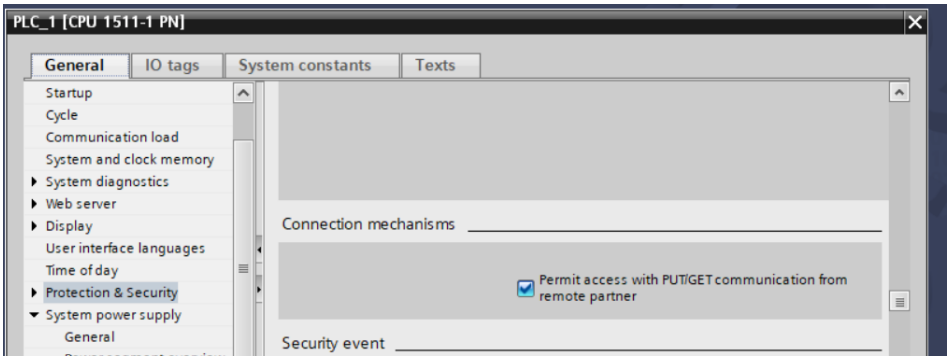


Figura 3.1: Habilitación de comunicación PUT/GET

La plantilla para un PLC S7-1500 y 1200 incluye un bloque de función (FC9000), el cual debe estar llamada en una network del Main del programa. Esta función es la que permite que Factory IO pueda detectar y conectarse con el dispositivo virtual de PLCSIM estándar. Su función es sincronizar la memoria extrayendo los datos de las salidas Q del PLC virtual y enviarlas al simulador 3D y también recibe datos de los sensores para forzar las entradas I de la instancia, todo esto en cada ciclo SCAN del PLC. Un PLC real no requiere este bloque.

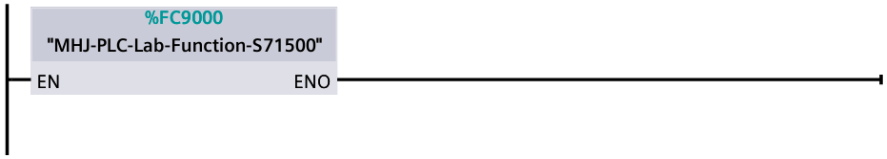


Figura 3.2: Bloque FC9000

Finalmente, en la plantilla para un S7-1500 están configurados unos módulos de entradas analógicas y otro de entradas digitales, el propósito de estos módulos es "crear" entradas para que la función FC9000 escriba y fuerce estas entradas según los datos de los sensores que recibe desde Factory IO, si no se colocan estos módulos notificará un error al cargar el programa al PLC simulado y además el bloque FC9000 no podrá escribir sobre una entrada que no tiene un hardware asociado, por ende, no habrá sincronización con las entradas del PLC virtual y los sensores de Factory I/O.

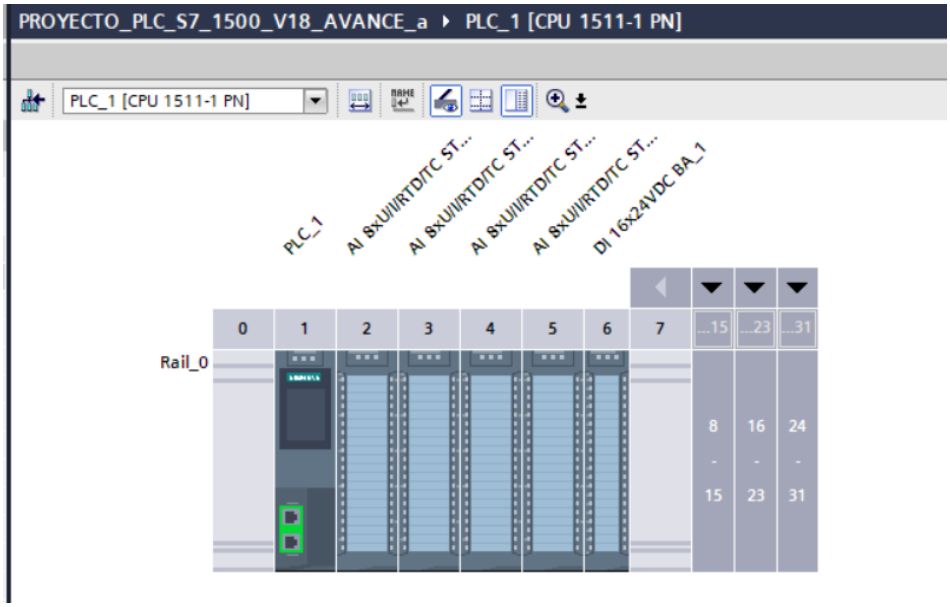


Figura 3.3: Módulos de entradas

Sobre esta plantilla se realizó toda la programación para que funcione el almacén automatizado diseñado en Factory I/O. En la configuración del simulador 3D se debe especificar que el controlador será PLCSIM, también se debe indicar la cantidad de direcciones de entradas/salidas del PLC que se van a usar en Factory, esto será según el programa y la cantidad de sensores y actuadores del escenario.

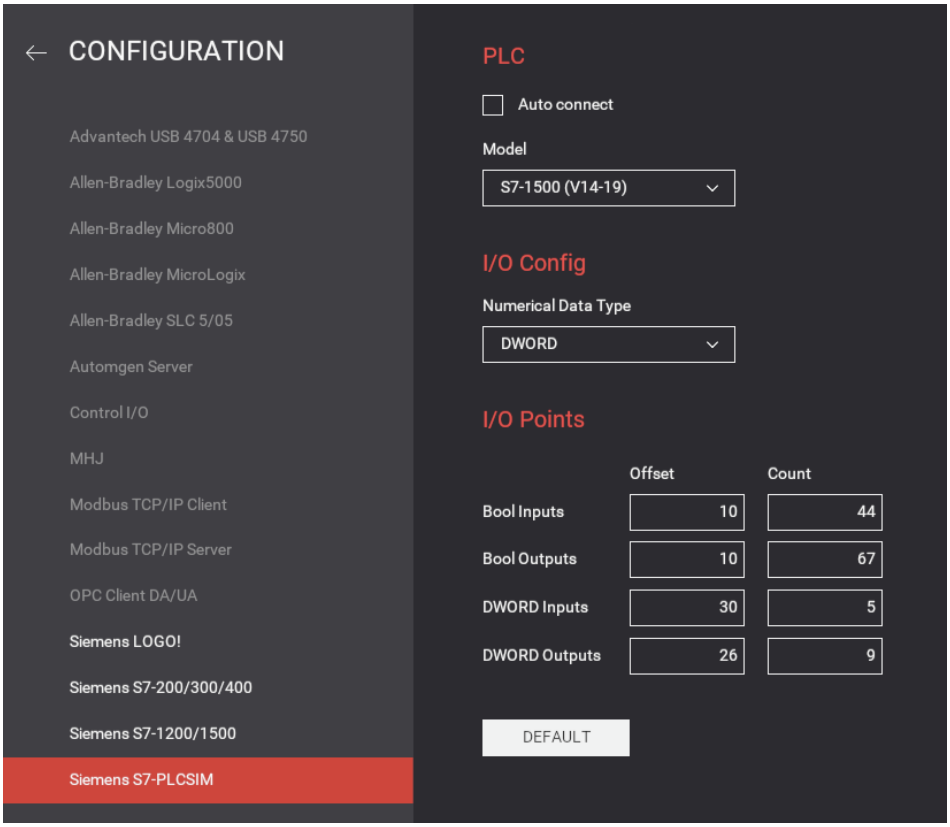


Figura 3.4: Configuracion en Factory IO

3.2. Programación

Se optó por simular un PLC S7-1500, porque, se sabe que es superior que un S7-1200 en muchos aspectos, por ejemplo, tiene mayor velocidad de procesamiento, es decir, tiene un ciclo scan más corto, presenta mayor capacidad de almacenamiento, y tiene una alta escalabilidad, ya que, admite hasta 32 módulos de expansión de entrada y salida. Todo esto porque está diseñado para sistemas más complejos y programas más grandes. En un principio se diseñó un almacén mucho más grande que iba a requerir muchas entradas y salidas tanto booleanas como reales (Dword), sin embargo, en el proceso se descubrieron las limitaciones de Factory IO, de igual manera se mantuvo la simulación de un S7-1500. A continuación, se describe el algunas de las funciones más importantes.

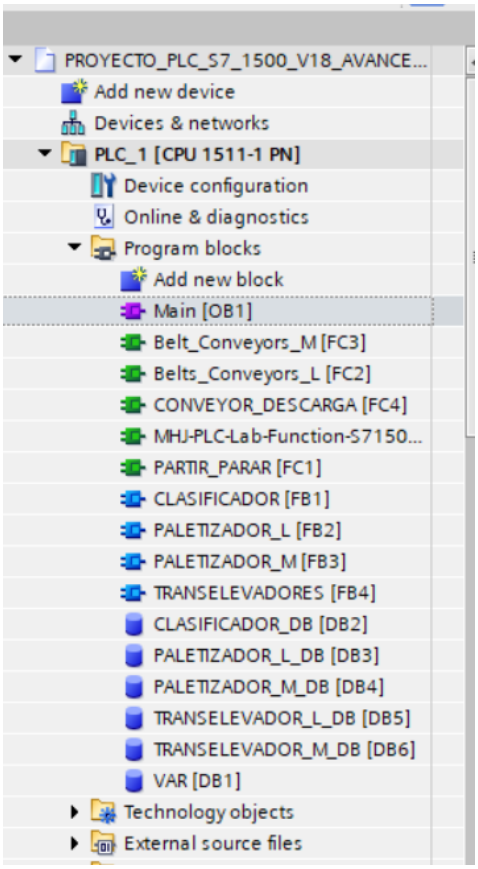


Figura 3.5: Árbol del proyecto

3.2.1. Main [OB1]

Es el bloque principal en el cual están organizadas todas las funciones y bloques de funciones que se programaron para que el PLC los ejecute de manera continua y cíclica. Consta de 8 networks.

3.2.2. Partir Parar [FC1]

Es la primera función llamada en el Network 1 del Main, su función es como su nombre lo indica, iniciar y detener todo el proceso, sus entradas I10.0 y I10.1 son los pulsadores de la maqueta virtual y las salidas booleanas las cuales corresponden a las luces piloto, y cuenta con una salida de dato entero asociada al actuador que emite las cajas y controla qué tipos de caja se van a emitir, si se llega al límite de cajas almacenadas de un tipo se inicia el despacho automático, por lo tanto se deja de emitir ese producto en la entrada, si se inician ambos despachos automáticos no se emite nada. También cuenta con otras dos variables de partir y parar (%DB1.DBX16.2 y %DB1.DBX16.3) que son las variables que se modifican en el panel confeccionado en Node-Red.

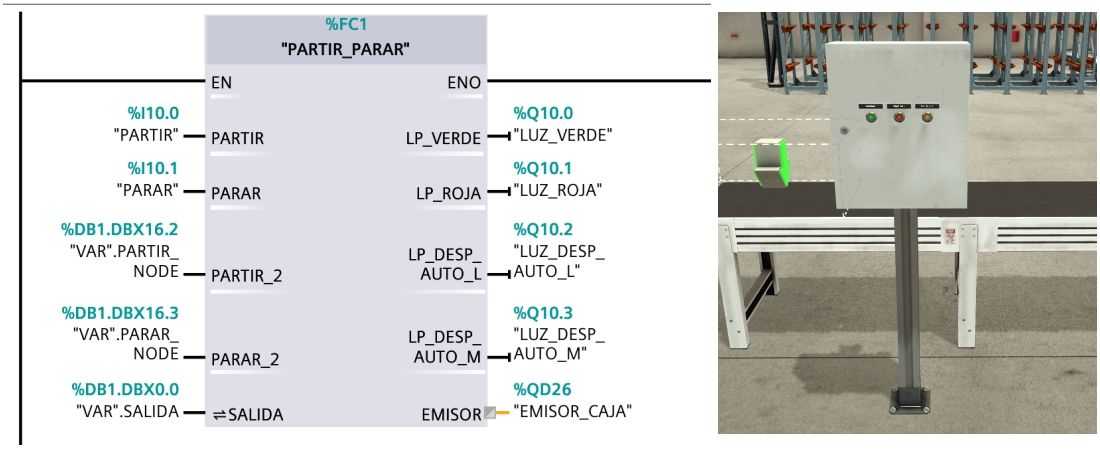


Figura 3.6: Network 1 y Operadores

3.2.3. Clasificador [FB1]

En la network 2 del main se encuentra este bloque de función, que está programado para mover las primeras cintas transportadoras y desviar las cajas según su peso, en sus entradas están asignados las direcciones de sensores difusos y tiene una entrada %ID30 que corresponde a un valor real entre 0v y 10v que entrega el conveyor báscula, este valor representa el peso del objeto. En las salidas están asignadas los actuadores de las cintas transportadoras y los brazos clasificadores pivotantes de cada caja.

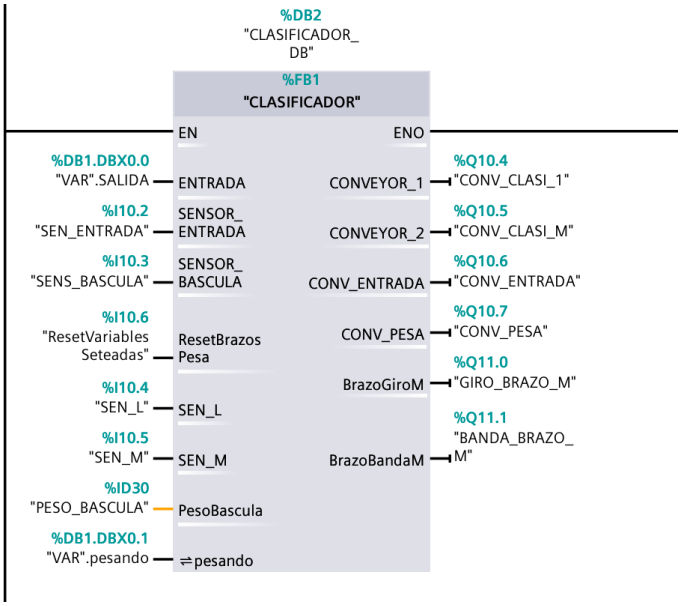


Figura 3.7: FB1 LLAMADO EN MAIN

Cuando una caja llega al transportador báscula es detectada por un sensor difuso el cual fija en estado TRUE la variable local “pesando” y hace detener el transportador por un tiempo de 750 milisegundos (figura 3.9), cumplido el tiempo, según lo que midió la báscula va a enclavar (“setear”) la variable local correspondiente al tipo de caja y a la vez va a volver a activar el conveyor de la pesa para que la caja avance (figura 3.8), la variable que indica el tipo de caja activa los brazos pivotantes para que clasifique (figura 3.10), el conveyor previo a la báscula, llamado “CONV_ENTRADA” se detiene mientras esté la variable “pesando”fijada en TRUE y una caja esté en el “SENSOR_ENTRADA” hasta que haya clasificado la primera caja. Finalmente, un sensor de salida por cada destino que toma una caja clasificada hace resetear o desfijar las variables “pesando” y la que indica el tipo de caja, dejando de estar activado el brazo, y permite mover el “CONV_ENTRADA” y entre la siguiente caja a la báscula (figura 3.11).

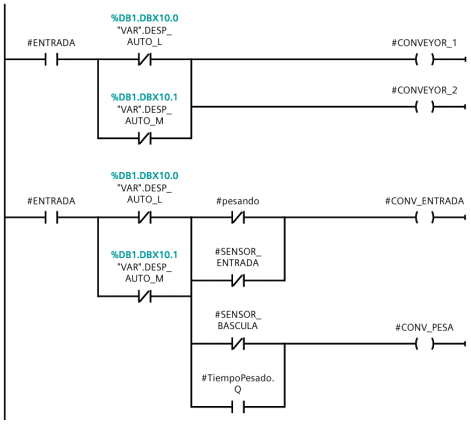


Figura 3.8: Net. 1 CLASIFICADOR

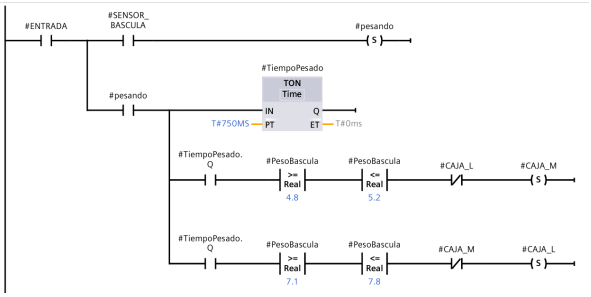


Figura 3.9: Net. 2 CLASIFICADOR

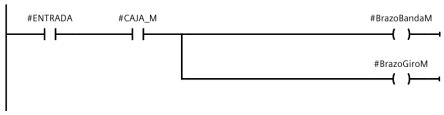


Figura 3.10: Net.3 CLASIFICADOR

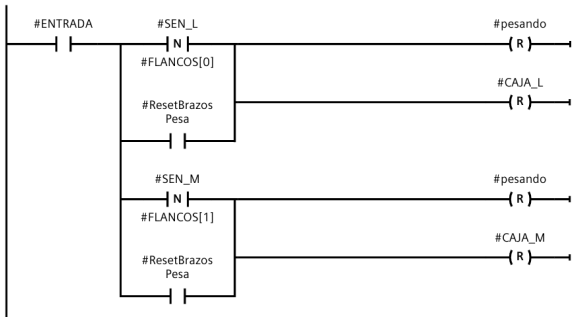


Figura 3.11: Net. 4 CLASIFICADOR

3.2.4. Paletizador [FB2] y [FB3]

Estos bloques de funciones están ubicados en el Network 3 y 4 cada uno está encargado de controlar todo el sistema el paletizado de las cajas clasificadas es decir se ocupa control de las cintas, en conjunto con la máquina pick and place, y los transportadores de rodillos con el pallet. Como entradas tiene la variable “SALIDA” que es el encendido del partir parar, los sensores difusos (%i11.3 y %i11.1 en este caso), el sensor del brazo de la máquina de tipo booleano, y los sensores de posición de los brazos los cuales son datos de tipo DWORD (son las posiciones en los ejes X, Z, representados en valores decimales de 0 a 10v), y como salidas tiene los actuadores que establecen la posición del brazo (X, Y, Z, salidas QD), la ventosa del brazo y el transportador de rodillo ambos de tipo booleano.

Estos bloques funcionan en conjunto con el bloque FC2 y FC3 que son los que controlan el movimiento de las cintas que llevan la caja hacia el paletizado que se detienen en casos especiales como la acumulación de cajas, o algún despacho en marcha, esto por seguridad.

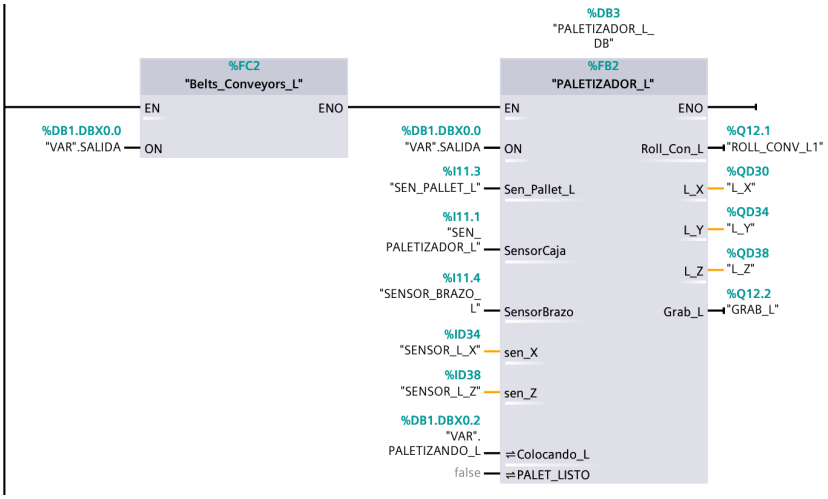


Figura 3.12: FB3 y FC2 LLAMADO EN MAIN

El funcionamiento de este bloque se basa en un contador de pasos, cada cuenta ejecuta un paso específico que una vez terminado aumenta en uno el número del contador para ejecutar el paso siguiente, así hasta que se arme un palet con cuatro cajas, una vez ejecutado el último paso se resetea este contador volviendo a 0 que es la condición inicial.

La secuencia comienza cuando la entrada “Sensor_Pallet_L” detecta que un pallet está en posición y detiene el “Roll_Conv_L” que es el transportador de rodillos que transporta el palet. Y cuando se detecta una caja en posición y hay un pallet en posición se demora 100 milisegundos en fijar la variable “Colocando_L”, haciendo que se abra el intelock y desactive el temporizador, esto es en caso de que llegue una caja mientras se ya está ubicando otra, de esta manera cuando se termine de colocar la caja, se pueda continuar el proceso con la siguiente.

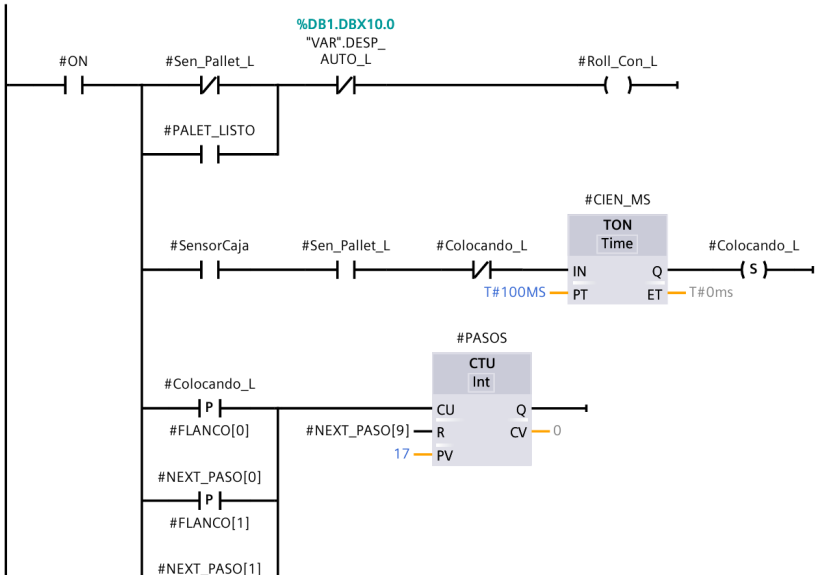


Figura 3.13: Contador de Pasos (Network 1 [FB2])

Cuando se fija “Colocando_L” en estado TRUE aumenta 1 en el contador de pasos solo en el flanco ascendente, esto es para que una vez pasado un ciclo scan del PLC no se mantenga el contacto cerrado y se impida seguir contando (el contador funciona con pulsos) y ejecuta el paso correspondiente a la secuencia, si es el primer paso entonces se mueve el brazo a la posición donde está la caja, es detectada por el sensor del brazo y la toma (activación de #Grab_L en figura 3.14).

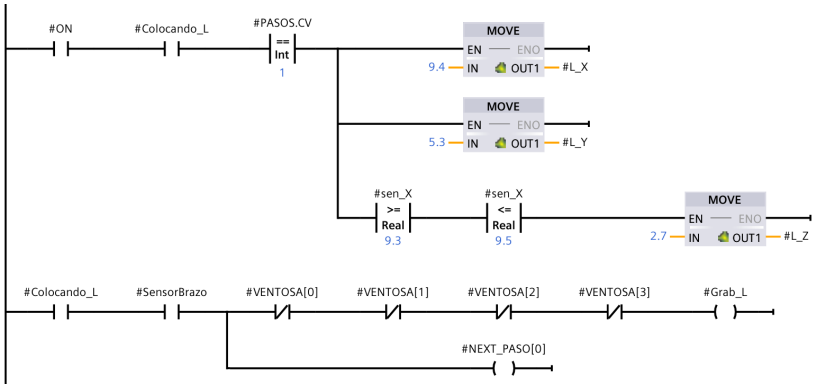


Figura 3.14: Ejecución Paso 1 (Network 2 [FB2])

Recordemos que la máquina pick & place funciona con actuadores (setpoints, “L_X” , “L_Y”, “L_Z”, que son las de este bloque) y sensores (Posición actual del brazo “sen_Z” y “sen_X” que son entradas) por cada uno de los 3 ejes de movimiento (X, Y, Z), son datos de tipo DWord que representan números reales entre 0.0v a 10.0v que indican la posición del brazo en ese eje. Cuando el sensor del brazo detecta una caja se activa la ventosa asignada con la salida booleana “Grab_L” en este caso. Y se ejecuta el siguiente paso, donde se lleva la caja a una posición del palet, luego se desactiva la ventosa y se desfija la variable “Colocando_L” (Figura 3.16).

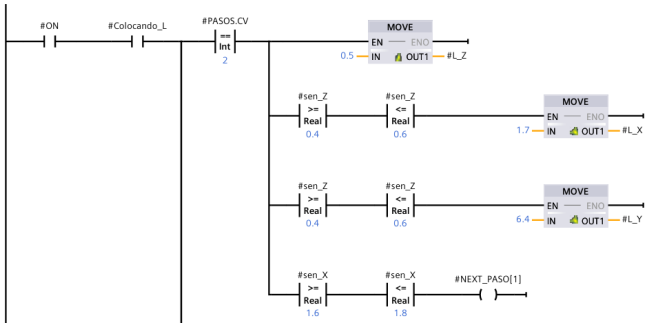


Figura 3.15: Ejecución Paso 2 (Network 3 [FB2])

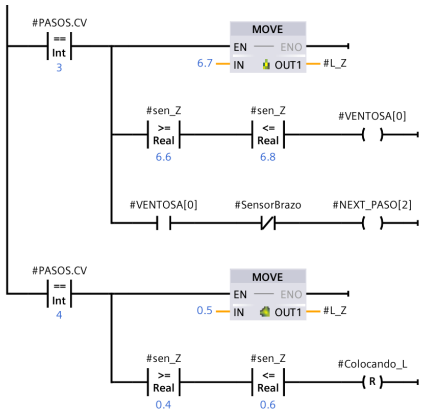


Figura 3.16: Ejecución Paso 3 y 4 (Network 3 [FB2])

Anteriormente se describió el bloque de función FB2 correspondiente al paletizado de cajas grandes, por otra parte, el FB3 encargado de controlar el paletizado de cajas medianas funciona de igual manera, pero, con ligeras diferencias ya que las cajas son de dimensiones diferentes, por lo que las posiciones del palet son distintas, además las cajas medianas son más largas, por lo que se requiere activar el giro de las cajas.

En resumen, cada paso del contador ejecuta una instrucción, por ejemplo, establecer la posición del brazo, y según los sensores de posición y del brazo cuando lleguen a dicha posición se dará por terminado el paso y aumenta el contador de pasos, lo que ejecutará la siguiente instrucción, así paletizará cajas de manera cíclica de manera automática.

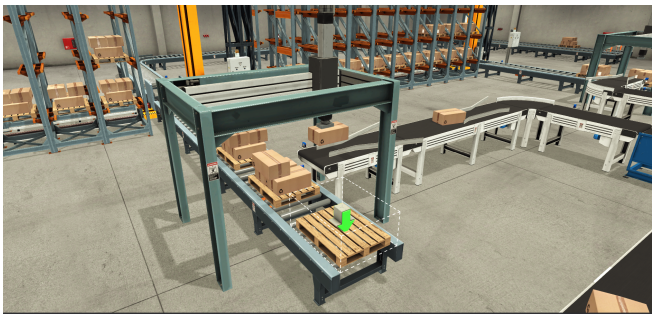


Figura 3.17: Paletizador Cajas Medianas



Figura 3.18: Paletizador Cajas Grandes

3.2.5. Transelevadores [FB4]

Es la función encargada de controlar los dos Transelevadores, es un único bloque de función llamado en las Networks 5 y 6 del MAIN [OB1], ya que cada uno tiene el mismo funcionamiento. Está programado para que cuando esté iniciado el sistema, sea capaz recibir un pallet con cajas y depositarlo en la posición correspondiente y si recibe otro lo almacena en la siguiente posición, y así con todos los pallets que llegan. Para despachar, se programó para que vaya a buscar la última caja que se almacenó y la deposita en los transportadores de rodillos de despacho que dirigen la carga hacia la salida del almacén. En el simulador 3D (Factory I/O) se designó un botón para extraer un solo palet, otro que activa el despacho automático hasta que se vacíen las estanterías y otro botón para parar el despacho, si se llega al límite de 45 palets almacenado de activa el despacho automático.

Por último, se programó para que se pueda despachar un pedido específico de pallets de cada caja, esto se puede hacer a través de una HMI simulada o a través de la dashboard confeccionada en Node Red.

En las entradas locales son todas booleanas, están los sensores difusos y los sensores que corresponden al transelevador, de los cuales tres indican la posición de las horquillas, dos indican si se están moviendo en eje X y/o Z, otros tres a botones, (uno para despachar un palet, otro para iniciar el despacho automático manualmente y otro para para pararlo). Como salidas booleanas tiene transportadores de rodillo, actuadores de la máquina (Horquillas, levantamiento y una alarma que indica que está despachando o almacenando) y como salidas numéricas, tiene la posición del transelevador, la cuenta del almacenamiento y despacho.

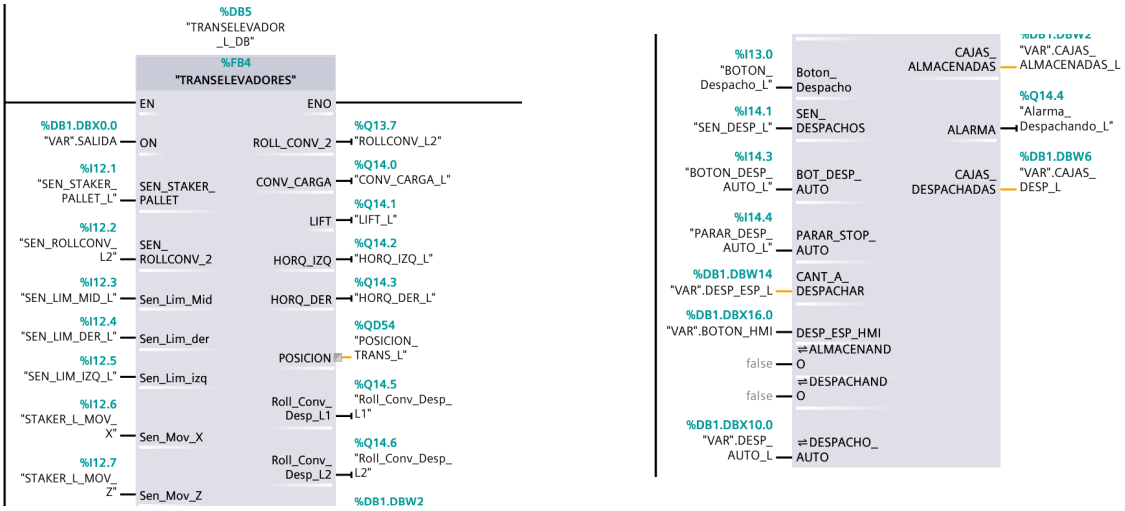


Figura 3.19: [FB4] Llamado a Network 5

El funcionamiento interno del bloque es utilizando el mismo principio que los bloques de paletizadores, es decir se utiliza un contador de pasos que hace que en cada paso se ejecuten ciertas instrucciones de manera ordenada. También cuenta con un contador el cual el "CV" (Current Vaule o Valor Actual) determina la posición en la cual colocar el palet. Este contador aumenta una posición cuando se fija en TRUE la variable "ALMACENANDO" y disminuye cuando se activa "DESPACHANDO".

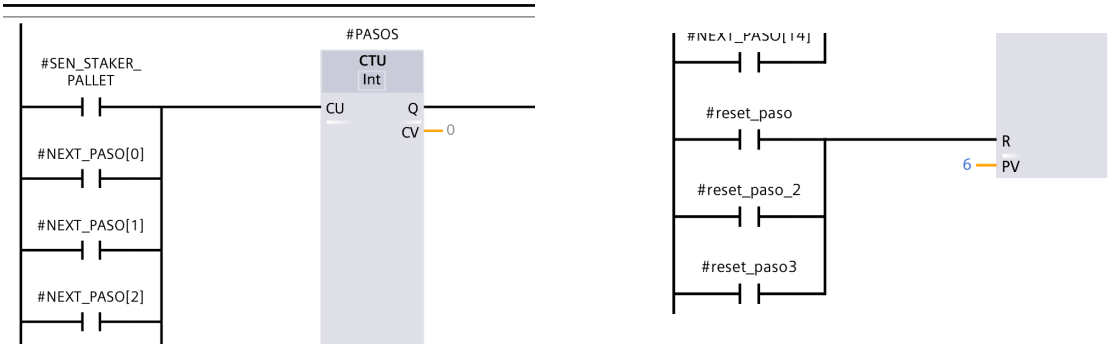


Figura 3.20: [FB4] Contador de pasos

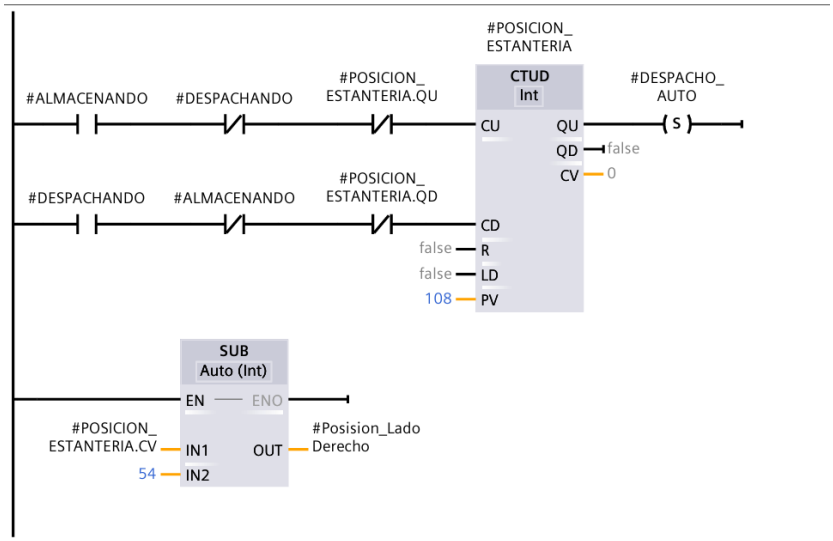


Figura 3.21: Contador de Posición.

Cuando se está despachando se va a disminuir una posición de la estantería por lo que al valor CV se le tiene que sumar 1 para que vaya a la posición de la estantería correcta.

En la figura 3.22 se muestra un fragmento del código para clarificar el funcionamiento. Cuando llegue un palet por los rodillos transportadores a la posición inicial, es detectado por un sensor y fija en TRUE la variable "ALMACENANDO" entonces aumenta en 1 el contador de la posición de la estantería y en 1 el contador de pasos, entonces en el primer paso se fijan la variable de la horquilla para que se mueva hacia la izquierda (donde está el pallet), una vez llegado al límite (sensor de límite izquierdo del transelevador) acciona "LIFT", que

levanta el palet y cuando el levantamiento cese el sensor de movimiento de Z se pondrá en FALSE que enviará un pulso de flanco de bajada que activa el siguiente paso aumentando el contador de pasos a 2. Luego se desfija la variable de la horquilla y una vez el sensor que indica que las horquillas estén en el medio se mueve el transelevador a la posición actual que tiene el contador "POSICION_ESTANTERIA.CV". Cuando se comienza a mover en el "Sen_Mov_X" activará el siguiente paso.

Y así es como funciona, cada paso ejecutado aumenta un número en el contador y ejecuta el siguiente hasta que termina el almacenamiento o despacho y vuelve el contador de pasos a 0.

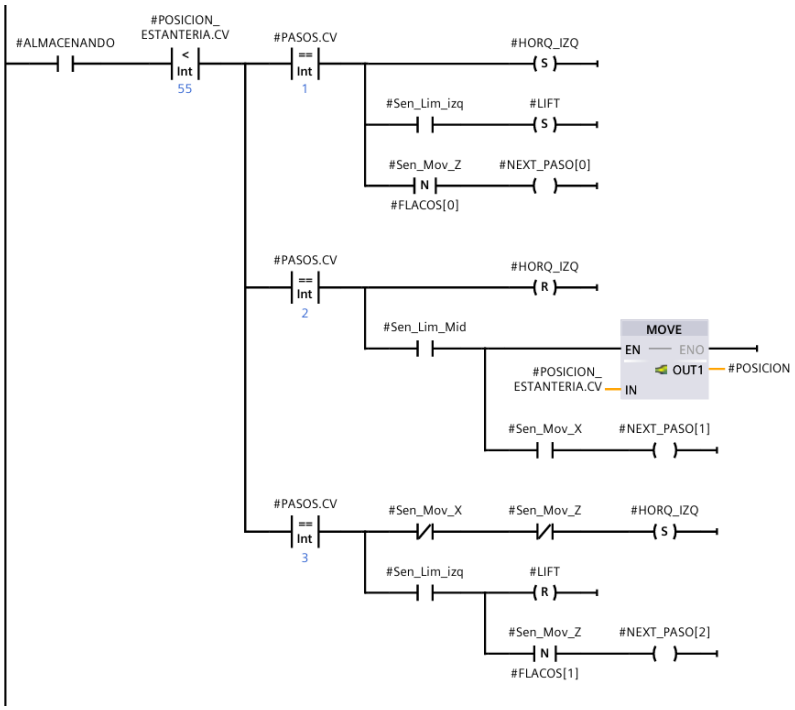


Figura 3.22: Ejemplo ejecución de Paso 1, 2 y 3

3.2.6. Bloque de Variables Globales "VAR" [DB1]

En este bloque de datos globales se guardan las variables que actúan en el programa, por ejemplo, la variable "SALIDA" que es la que se activa en la función FC1(Partir Parar inicial) y activa los demás bloques, o también para guardar algunas variables importantes como por ejemplo el conteo de las cajas almacenadas y despachadas de cada tipo, las variables que se modifican en la HMI y panel de Node Red, entre otras.

VAR

		Name	Data type	Offset	Start value
1		Static			
2		SALIDA	Bool	0.0	false
3		pesando	Bool	0.1	false
4		PALETIZANDO_L	Bool	0.2	false
5		PALETIZANDO_M	Bool	0.3	false
6		CAJAS_ALMACENADA...	Int	2.0	0
7		CAJAS_ALMACENADA...	Int	4.0	0
8		CAJAS_DESP_L	Int	6.0	0
9		CAJAS_DESP_M	Int	8.0	0
10		DESP_AUTO_L	Bool	10.0	false
11		DESP_AUTO_M	Bool	10.1	false
12		DESP_ESP_M	Int	12.0	0
13		DESP_ESP_L	Int	14.0	0
14		BOTON_HMI	Bool	16.0	false
15		SWITCH	Bool	16.1	false
16		PARTIR_NODE	Bool	16.2	false
17		PARAR_NODE	Bool	16.3	false

Figura 3.23: VAR [DB1]

3.3. Conexión de PLCSIM y Factory I/O

Para que este programa actúe sobre el escenario diseñado en Factory IO se procedió a simular el programa de manera normal, la cual TiaPortal tiene por defecto, es decir con PLCSIM, es necesario asegurarse que el bloque FC9000 se encuentre llamado en una línea del Main, de otra manera el simulador no detectará ni se conectará al PLC virtual.

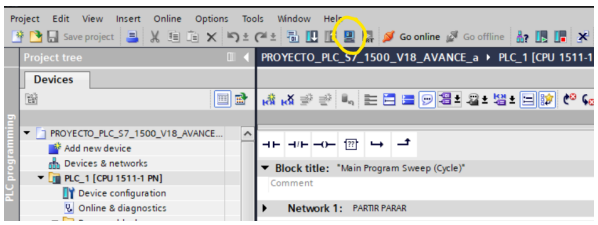


Figura 3.24: Iniciar simulación

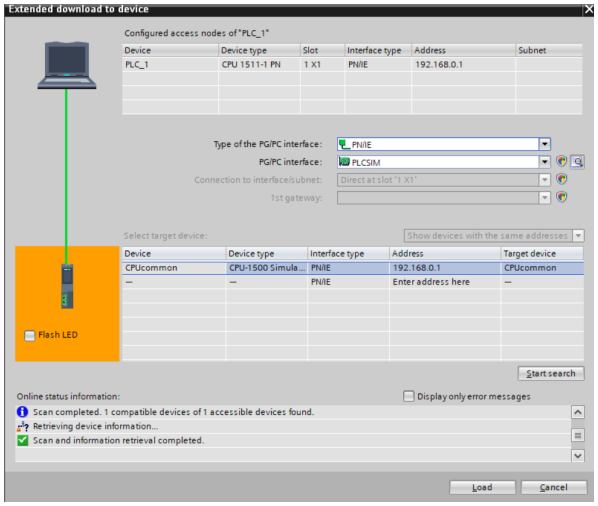


Figura 3.25: Cargar a la instancia

Una vez cargado el programa a la instancia, se debe configurar Factory IO para que sea controlado por PLCSIM, se debe especificar la cantidad de entradas y salidas que se van a utilizar y luego asignar los sensores y actuadores a las direcciones correspondientes según nuestro programa.

Finalmente, de debe conectar y simular, el escenario de Factory IO funcionará de acuerdo al programa que esté descargado en el PLC virtual.

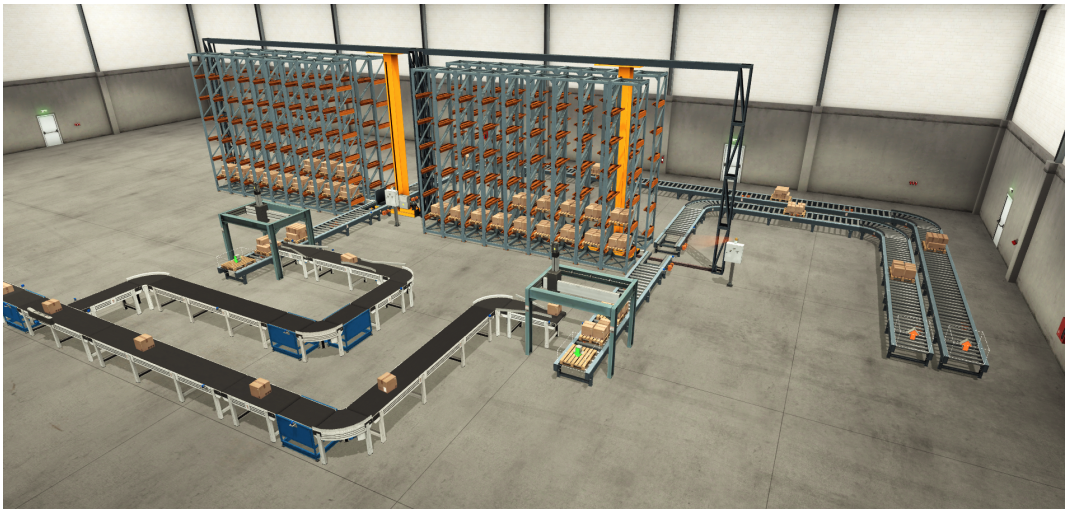


Figura 3.26: Almacen controlado por PLCSIM

4. Monitoreo y Control Remoto

El monitoreo de un proceso industrial a través de internet se ha convertido en algo de vital importancia en la industria 4.0, ya que, gracias a esto es posible tener más visibilidad y control de los procesos desde cualquier parte del mundo, también se pueden configurar alertas o notificaciones que indique que una variable crítica tenga un comportamiento anómalo y así se agilice la toma de decisiones pudiendo incluso controlar equipos de manera remota.

En este capítulo se documenta el proceso para poder monitorear y controlar algunas de las variables del PLC virtual. Para esto se utilizó la herramienta Node-red, la cual es usada para recopilar, visualizar y enviar datos de hardware en tiempo real, se utilizó de dos maneras, una para enviar los datos del PLC virtual hacia una aplicación en la nube de Azure, esta aplicación se llama “IoT Central”, que está enfocada a brindar soluciones empresariales de monitoreo de dispositivos de manera remota. Y la otra manera es creando una dashboard (con la librería “node-red-dashboard”) o panel de operador que puede visualizarse estando en la misma red o en la misma VPN (Red privada virtual) del ordenador que alberga el servidor de Node-Red.

4.1. Conexión entre Node-Red y PLCSIM Standard

4.1.1. Uso de NetToPLCsim

Existe un problema con el simulador del PLC y es por la misma razón que se requiere una plantilla para conectarse con Factory I/O, la cual es, que no está vinculada a una tarjeta de red física, por lo tanto, no es visible para el protocolo TCP, el PLC está encerrado en la simulación. La solución es utilizar un pequeño software llamado “NetToPLCsim”, cuya función es conectarse a PLCSIM y permitirle que use el adaptador de red de el PC que alberga el PLC virtual (Puede ser el adaptador de ethernet o WiFi) para que use la dirección IP y pueda comportarse como un dispositivo real.

Primero en el campo de Network IP Address se selecciona la tarjeta de red. Luego en el campo de Plcsim IP addres se selecciona la instancia del PLCSIM que generalmente está con la dirección 192.168.0.1

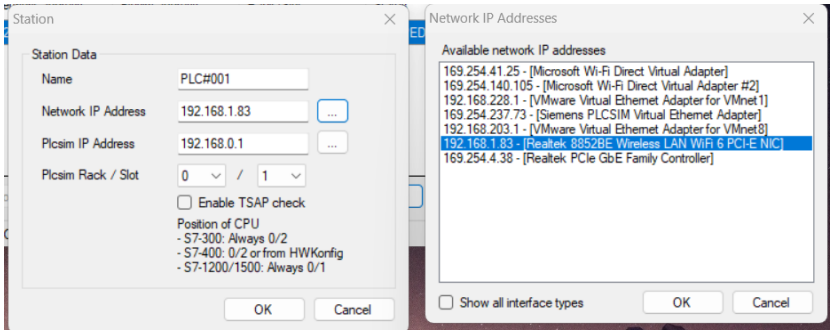


Figura 4.1: Selección de tarjeta de red

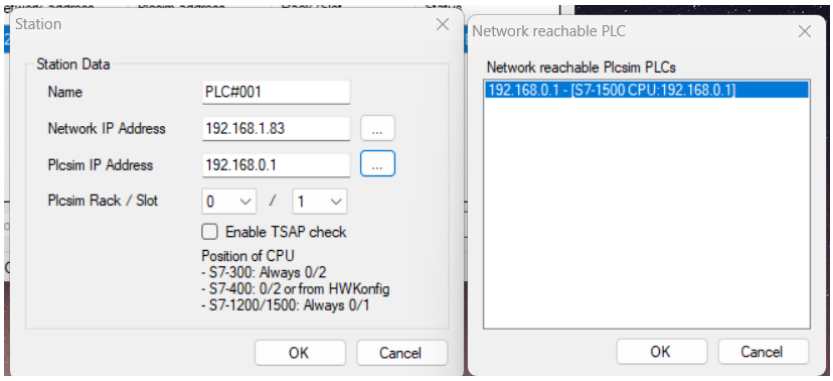


Figura 4.2: Selección de instancia virtual de PLCSIM

Luego de realizar esto se inicia el servidor, ahora el PLC virtual puede ser visto a través de esta app y puede conectarse con Node-Red. (Node Red en realidad está conectado a NetToPLCsim, el cual recibe los datos de la simulación de PLCSIM). Gracias a esto Node red puede leer y/o escribir las variables del PLC virtual como si fuera uno físico en tiempo real.

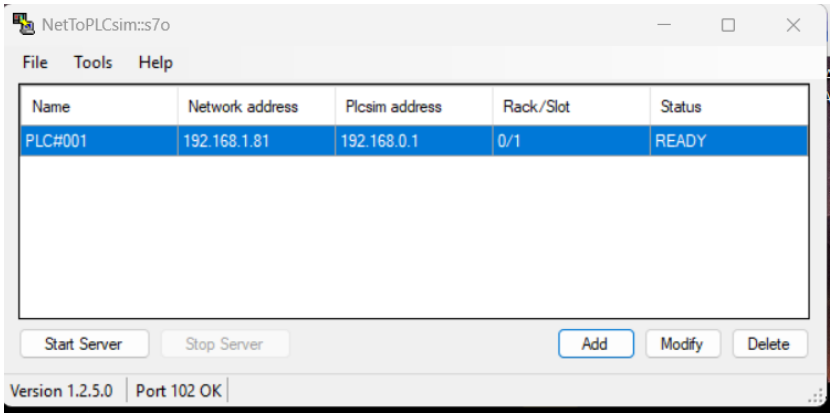


Figura 4.3: Iniciar servidor

4.1.2. Lectura de variables con Node red

Node-Red es una herramienta de programación visual de bajo código, que se usa para comunicar fácilmente hardware con APIs, servicios de internet, etc. Es muy común que se utilice en aplicaciones de IoT y monitorización industrial. Node Red funciona con bloques llamados nodos que tienen funciones específicas y se unen con otros nodos para crear un programa, el cual se denomina “Flujo”. Se utilizaron nodos de la librería llamada “node-red-contrib-s7”, los cuales son para leer o escribir variables de un PLC de marca Siemens. El nodo “s7 in” es para leer una memoria del PLC, “s7 out” para escribir en una memoria.

En este proyecto se utilizaron ambos, para usarlos se debe configurar de la siguiente manera, primero se debe añadir un dispositivo, el cual es llamado “end point” en Node-Red y configurar la dirección IP del PLC, en este caso se asigna la dirección IPv4 del adaptador de red, la cual es 192.168.1.83 correspondiente al PC, porque estamos utilizando “NetToPLC-sim”. El protocolo de transporte es ISO on TCP el cual usa el puerto 102.

Además, se deben asignar las variables del PLC que se quieren visualizar, en este caso son las variables alojadas en el bloque de datos globales [DB1].

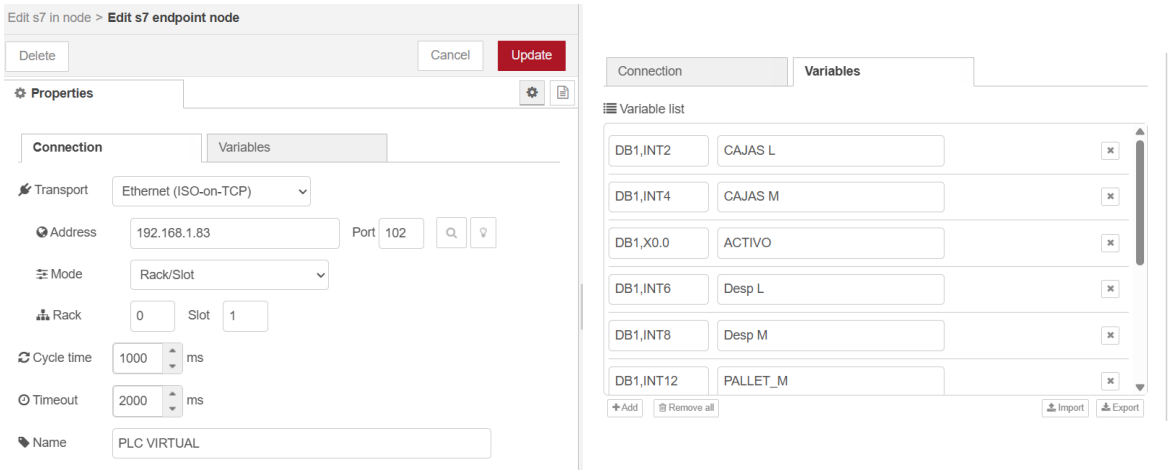


Figura 4.4: Configuración de end point Node-Red

Para que las variables de DB1 puedan ser visibles se debe configurar las propiedades del bloque, en atributos, se desmarcó la casilla de “Optimized block access” (acceso optimizado al bloque) esto hace que cada variable del DB1 tenga su propia dirección específica. Por ejemplo la primera variable booleana será DB1.DBX0.0.

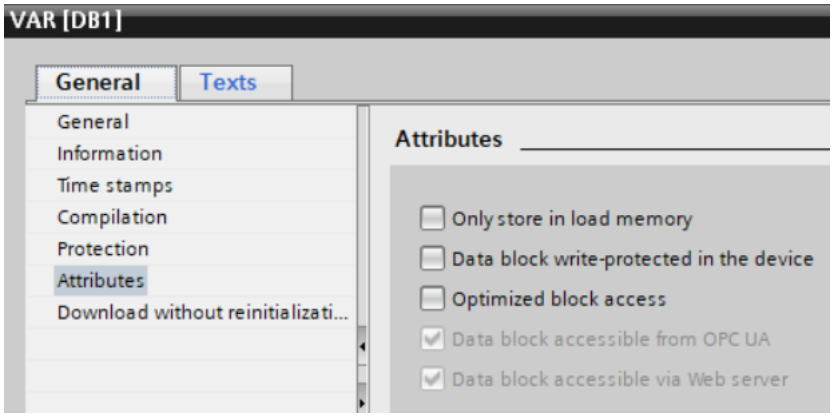


Figura 4.5: Propiedades DB1.

Una vez hecho lo anterior cada nodo s7 in debe configurarse especificando el dispositivo (end point) y la variable que debe leer.



Figura 4.6: Lectura de Variables con Node red

4.2. Servicio de Azure IoT Central

Azure IoT Central es una plataforma de Microsoft Azure que se usa principalmente para aplicaciones de domótica, que brinda una interfaz web y APIs listas para usar para conectar hardware, monitorear, crear reglas, gestionar dispositivos, confeccionar paneles de visualización, etc. En resumen, es un servicio sencillo de usar y en este proyecto se usará para monitorear las variables del PLC virtual que recibe Node Red, es decir, el conteo de productos almacenados y despachados del almacén.

4.2.1. Creación de una plantilla para dispositivo

Primero se debe agregar y configurar una plantilla en la cual se deben especificar las variables a mostrar. Cada una tiene un nombre específico el cual debe ser el mismo nombre establecido en un nodo de función de Node-Red.

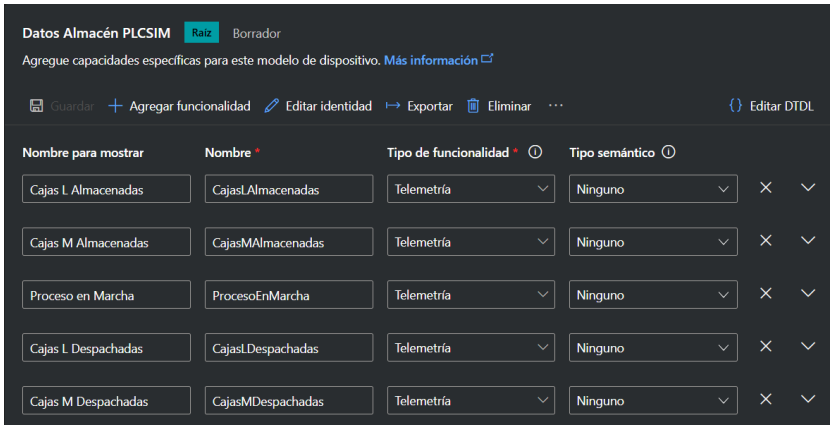


Figura 4.7: Creación de Datos en Azure IoT Central

Luego se configura la plantilla y cómo se desea que se visualicen los datos que se acaban de crear, se optó por usar objetos visuales que indican el último valor conocido.

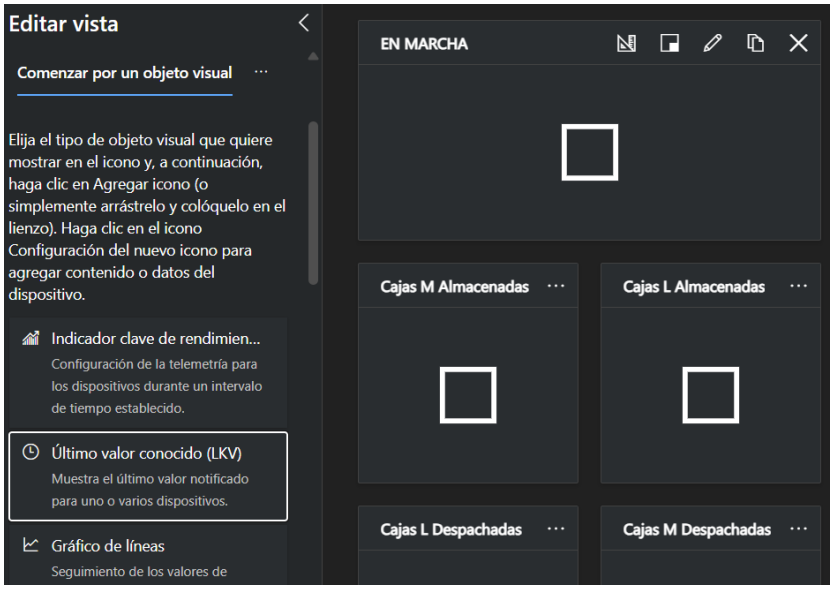


Figura 4.8: Edición de la Plantilla

4.2.2. Visualización de datos en la Aplicación

Se crea un dispositivo y se le asigna la plantilla creada y se conecta al nodo “Azure Iot Central” (que pertenece a la librería “azure-iot-central” que se debe instalar), se debe configurar el nodo colocando los mismos datos de seguridad para crear este enlace.

Y finalmente los datos se visualizaron de esta manera desde un dispositivo móvil a través de internet.

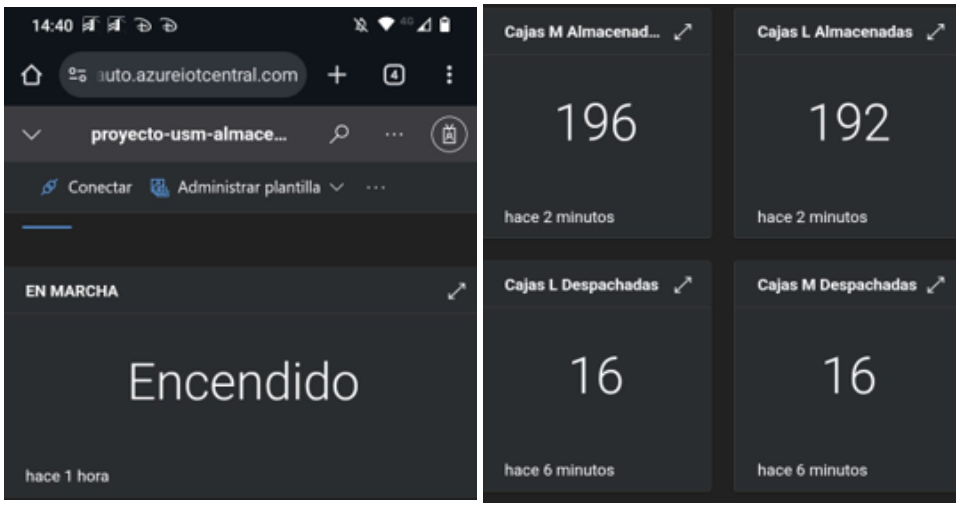


Figura 4.9: IoT Central desde Móvil

4.3. Monitoreo y control utilizando Dashboard de Node Red con una VPN

4.3.1. Confección de Dashboard en Node red

Para diseñar el panel de operación se instaló la librería “node-red-dashboard”, la cual trae consigo 21 nodos, entre los cuales hay botones, interruptores, campos de entrada y salida numéricos, de texto, gráficos, etc.

Para utilizar dichos nodos primeramente se debe que configurar la estructura del panel, primero se asigna un nombre al panel general, en este caso se le llamó “DASHBOARD ALMACEN PROYECTO”. Luego se debe crear por lo menos un grupo, aquí se crearon tres grupos llamados “ESTADOS”, “ALMACENAMIENTO” y “DESPACHOS”. El propósito de estos grupos es seccionar el panel para que se visualice de manera ordenada. Cada vez que se utiliza un nodo del panel (dashboard) se debe asignar a uno de estos grupos.

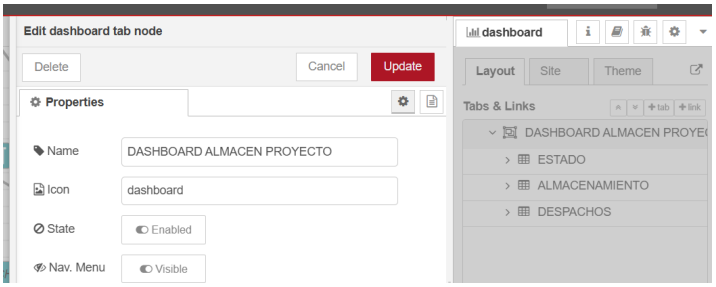


Figura 4.10: Nombre Panel General.

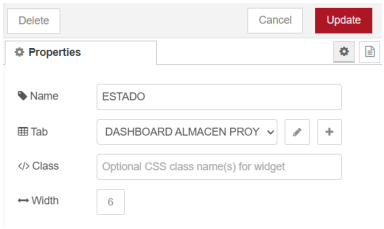


Figura 4.11: Creación de Grupo

El grupo “ESTADO” está formado por dos nodos de tipo botón que se les llamó PARTIR y PARAR, estos botones están enlazados a un nodo llamado “trigger 250ms” cuya función es enviar un pulso “true” por 250 milisegundos y luego envía un “false”, estos pulsos están conectados a nodos de tipo “s7 out” y escriben en una memoria del PLC, estas memorias son las que actúan en el bloque FC1 como pulsadores para arrancar y detener el sistema. También se incluyó un par de nodos led para que indiquen en qué condición está el sistema.

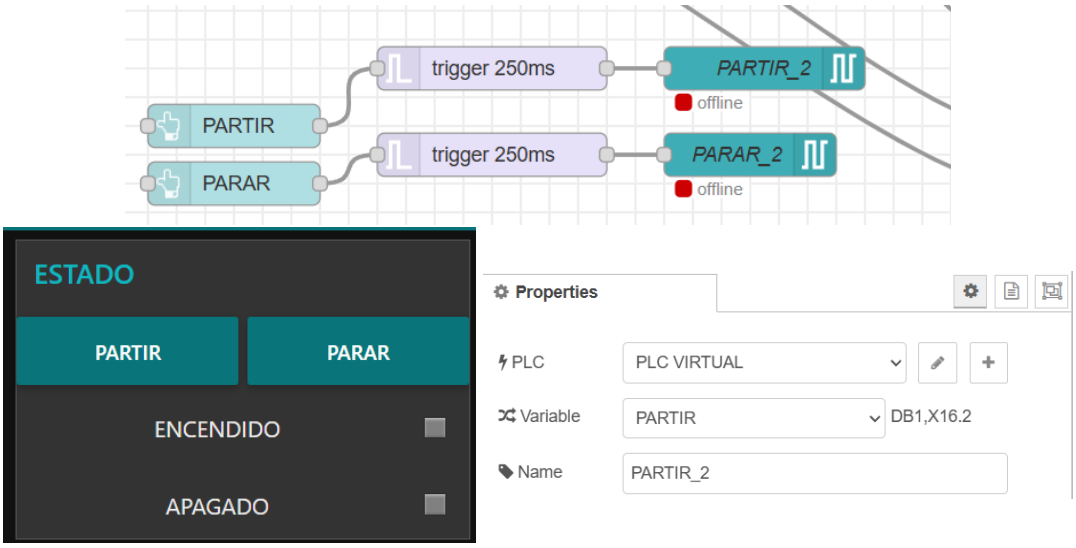


Figura 4.12: Programación de grupo estado

El grupo “ALMACENAMIENTO” está formado simplemente por dos nodos “text” que reciben la información de la memoria del PLC de las cajas almacenadas a través de los nodos “s7 in”.

Por último, el grupo “DESPACHOS” está formado por dos nodos “text” que son los indicadores de la cantidad de despachos, los cuales reciben información desde el PLC virtual, un nodo interruptor llamado “DESPACHO ON/OFF” conectado a un nodo “s7 out” que escribe en la memoria DB1.DBX16.1 del PLC, que habilita el despacho de una cantidad de pallets específica ingresada por el usuario a través de los nodos “Pallets Cajas L y M” y por último el botón “DESPACHAR” que gatilla la acción. Estos últimos cuatro nodos actúan sobre las memorias del PLC a través de los nodos “s7 out” respectivos.

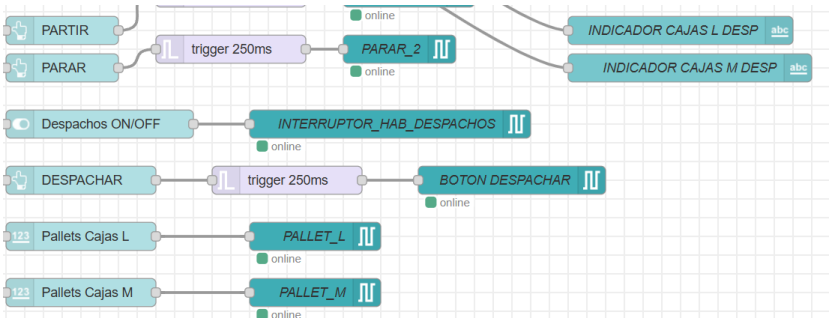


Figura 4.13: Flujo de Dashboard

Finalmente, así se visualiza el panel desde el ordenador, para acceder a este se debe escribir en un navegador web la dirección IPv4 del computador donde está en marcha el servidor de Node red seguido de :1880/ui, en la figura se utilizó la dirección de loopback (127.0.0.1) porque es el mismo PC. Para acceder desde otro dispositivo es necesario estar en la misma red de área local.



Figura 4.14: Dashboard final desde el PC

4.3.2. Acceso remoto con una VPN

Una VPN (Virtual Private Network) nos permite expandir nuestra red de área local de forma segura a través de internet, de esta forma podemos comunicarnos entre dispositivos como si estuvieran en la misma oficina, pero estando a kilómetros de distancia conectados a internet.

Se utilizó la aplicación ZeroTier, que una excelente herramienta para estos casos, entre sus principales características es que tiene conectividad global, es decir permite crear redes virtuales que abarcan todo el mundo, también es de alta seguridad, ya que, su sistema de seguridad es de tipo cifrado de extremo a extremo de 256 bits, garantizado que los datos estén protegidos contra el acceso no autorizado, además es fácil de configurar, entre otras características.

Pasos para crear red con ZeroTier:

I. Desde su página web se crea una cuenta y se selecciona el plan gratuito que permite crear una sola red con 10 dispositivos.

II. Crear y configurar nuestra red:

   Es importante rescatar la ID de la red (a581878f7d2ef219), ya que, esta es para agregar la red en otros dispositivos. La dirección ipv4 de red se asigna automáticamente, en este caso tocó la 10.149.79.0 /24.



Figura 4.15: Configuración de Red en ZeroTier

III. Descargar aplicación en dispositivos miembros:

En dispositivos móviles de sistema operativo Android ZeroTier está disponible en PlayStore, en Ordenadores desde la página web oficial, luego de eso se debe añadir la red ingresando la ID “a581878f7d2ef219”.

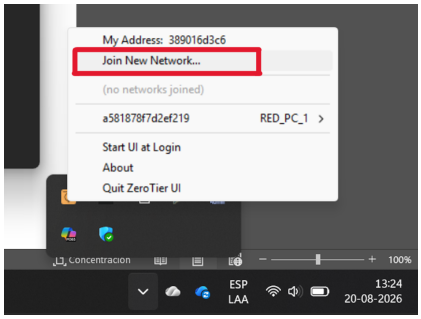


Figura 4.16: Añadir red en PC

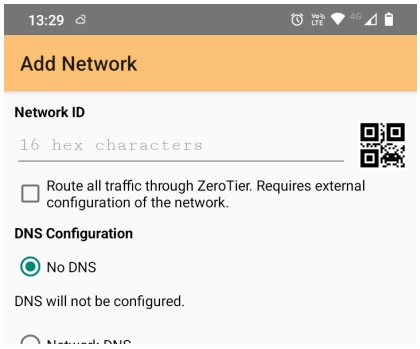


Figura 4.17: Añadir red en Android

IV. Gestionar los miembros:

Luego desde la interfaz de configuración de la red serán visibles los dispositivos que quieren ser parte de la red esperando que se de autorización, aquí se puede especificar el nombre del dispositivo y la dirección IPv4 se asigna automáticamente.

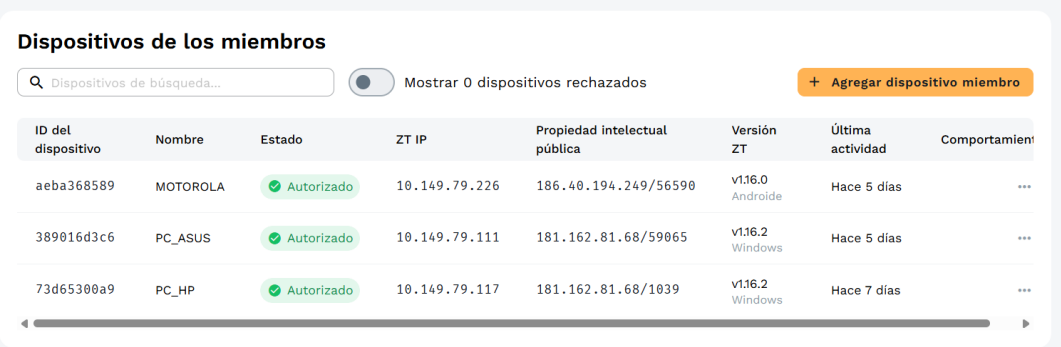


Figura 4.18: Dispositivos miembros autorizados

V. Ahora en cada dispositivo con acceso se debe conectar a esta red a través de la aplicación. Una vez hecho eso los dispositivos estarán conectados como si estuvieran en la misma red de área local.

Para acceder al panel de node red desde el dispositivo móvil que pertenece a esta VPN es lo mismo que si fuera un pc, se debe escribir en el navegador la dirección IPv4 del PC_ASUS es decir 10.149.79.111:1880/ui. Si observamos la imagen el celular está con acceso a internet con los datos móviles con señal 4G.

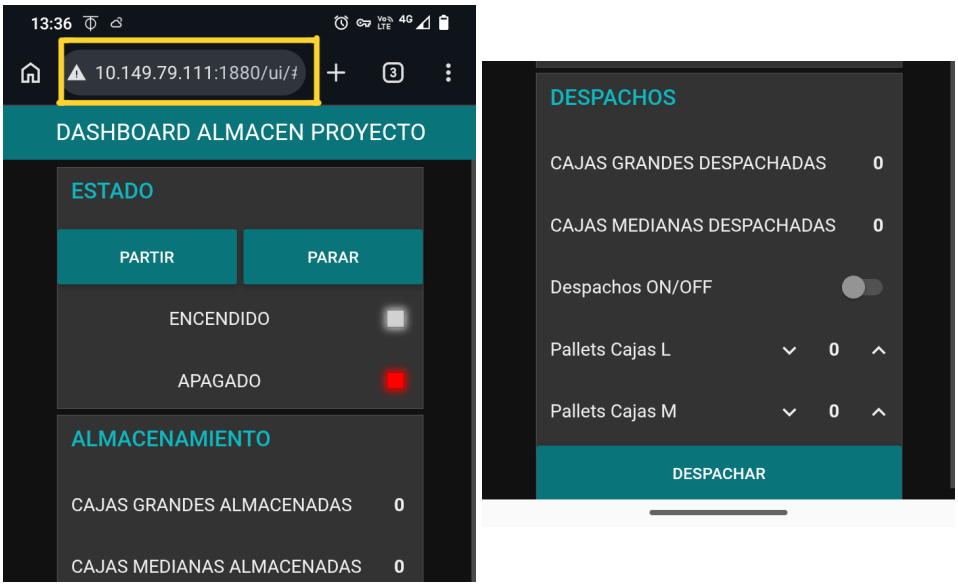


Figura 4.19: Panel de operación desde Android

De esta manera no solo se pudo monitorear, sino controlar el almacén automatizado a distancia, activándolo, deteniéndolo, o despachar un pedido específico.

5. Resultados y Conclusiones

5.1. Resultados

El fruto del proyecto en general es una simulación de un almacén automatizado por un PLC virtual del cual se pueden monitorear variables como el conteo de productos almacenados, despechados y ordenar un pedido de despacho concreto.

Logros específicos:

- **Escenario de Almacén en Factory I/O:** Se logró diseñar un almacén pensado para realizar tres procesos, la clasificación por peso de dos tipos de productos (cajas medianas y grandes), el paletizado de cada tipo, el almacenaje y despacho de éstas. Se logró delimitar el espacio de cada estación y la conexión de éstas mediante cintas transportadoras y de rodillos, también se ubicaron sensores difusos en ciertos lugares para usar con el programa (ya sea para detener las cintas, activar un proceso, etc.), al igual que botones, luces, entre otros.
- **Programa para PLC virtual:** Se programó en software TiaPortal v18 un proyecto para un PLC virtual capaz de automatizar el escenario diseñado, de tal manera que se permite iniciar y detener el sistema desde la maqueta virtual y desde un panel externo como lo es el de Node Red, también clasifica las cajas por su peso y luego las paletiza en grupos de 4 cajas, y por último permite el funcionamiento de los transelevadores que al recibir cada plataforma con cajas los almacena en un lugar en las estanterías, y también los despacha. Físicamente el almacén tiene capacidad total de 108 palés por cada tipo de caja, pero el software genera un número limitado de productos, por lo que en la programación se fijó como límite 45 palés por tipo de caja, una vez llegado a este límite se activa el despacho automático para ese grupo de cajas y dejan de entrar, si ambos llegan a ese límite se detiene la entrada de productos y se comienza a despachar.
- **HMI virtual local:** se diseñó una HMI con la cual se puede monitorear localmente la cuenta de los productos, además se puede ordenar un despacho específico.
- **Sistema de Monitoreo con Aplicación de Azure:** Se logró programar un "flujo" en Node-red, que permite enviar las variables del PLC virtual a la aplicación "Azure IoT Central" con la cual se diseñó un panel simple para monitorear el conteo de las cajas almacenadas y despachadas por tipo, al igual que el estado del sistema (iniciado o apagado).
- **Dashboard en Node Red:** Se logró confeccionar un Panel de Operación y monitoreo con Node-Red, desde cual se puede iniciar y apagar el almacén, ordenar una salida específica de productos, y finalmente monitorear los datos de cajas almacenadas y despachadas.
- **VPN:** Con la aplicación ZeroTier se creó una red privada virtual con el objetivo de acceder al panel de Node Red desde un dispositivo miembro de la VPN que esté conectado a internet desde otra red. Así se logró el acceso remoto.

5.2. Conclusiones

Observando los resultados, podemos decir que el proyecto cumplió los objetivos planteados del inicio de este informe, de cada meta específica se concluyó lo siguiente:

- **Diseño de Almacén:** Un buen diseño permitió el correcto funcionamiento del sistema automatizado, ya que, establecer de manera ordenada los procesos, elementos, aprovechando bien el espacio, permite mayor facilidad a la hora de programar y ejecutar el escenario.
- **Programación de PLC virtual e Integración:** Permitió la automatización total desde que se activa la simulación. La programación estructurada de Siemens permitió escribir un código ordenado separando las funciones que automatizan cada estación, lo que hace más fácil la modificación en caso de querer mejorar el programa.
- **Monitoreo y Control a distancia:** Concedió conocer la producción en tiempo real del almacén lo que permite medir el rendimiento y junto con el control a distancia permite agilizar la toma de decisiones y dar respuesta a alguna necesidad detectada en la supervisión.

Finalmente, se logró el cumplimiento del objetivo general del proyecto, que es simular un almacén automatizado capaz de realizar las tareas logísticas (clasificación, paletizado, almacenado y despacho) de manera autónoma y monitorear el conteo de productos, demostrando que se puede disminuir la intervención humana en procesos que pueden ser peligrosos para la salud como el levantamiento de carga, o la manipulación de maquinaria, en la cual un error puede ser fatal, también se aumenta la productividad, entre otras ventajas.

En entornos reales la automatización de un almacén es una inversión muy costosa, pero si todas las industrias y empresas se están modernizando o automatizando, los almacenes convencionales terminan siendo un cuello de botella en toda la cadena de producción.

6. Anexos

6.1. Driver Factory I/O



Figura 6.1: Entradas y salidas Factory I/O



Figura 6.2: Entradas y salidas Factory I/O P.2

6.2. Fragmentos de Programa en TiaPortal

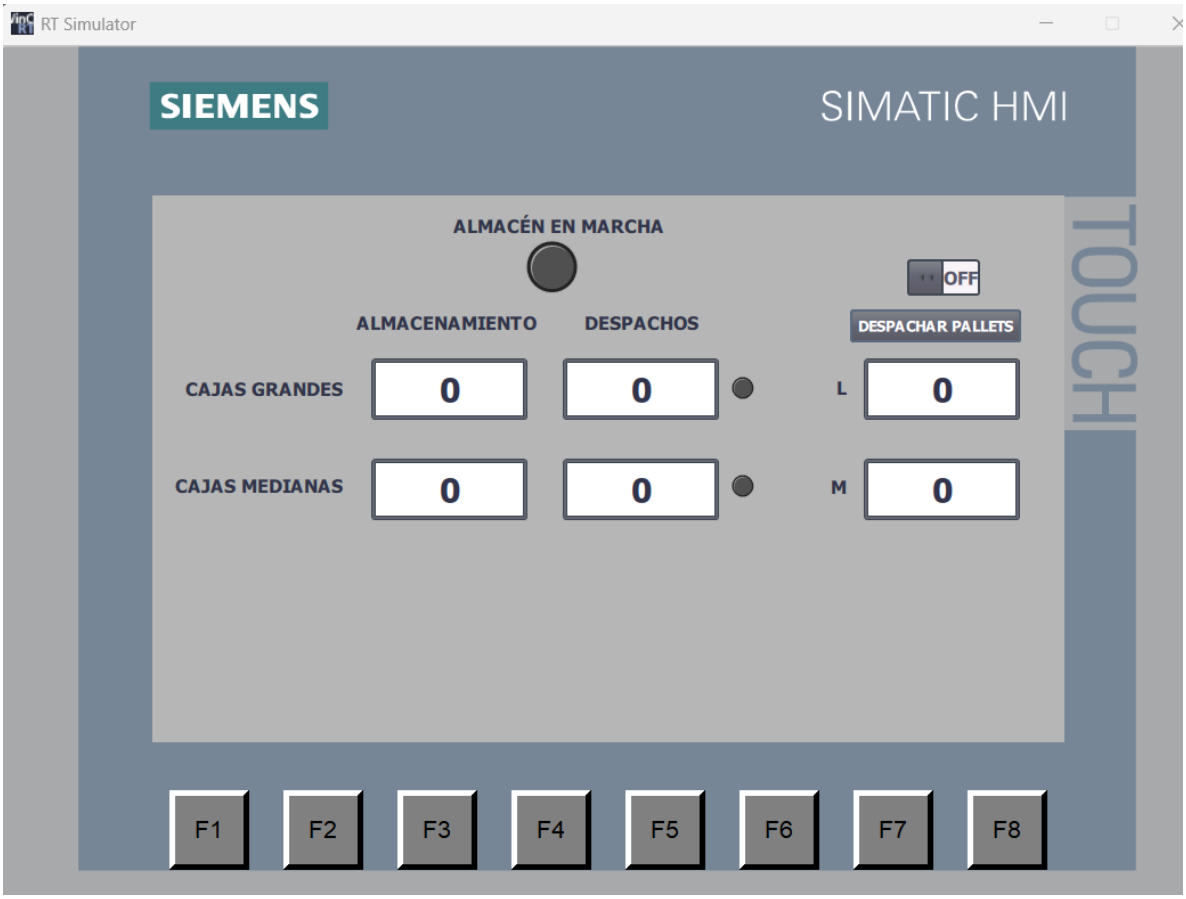


Figura 6.3: HMI

6.2.1. PARTIR_PARAR [FC1]

PARTIR_PARAR [FC1]

PARTIR_PARAR Properties							
General							
Name	PARTIR_PARAR	**Number**	1	**Type**	FC	**Language**	SCL
Numbering	Automatic						
Information							
Title		**Author**		**Comment**		**Family**	
Version	0.1	**User-defined ID**					

Name	Data type	Default value	Comment
Input			
PARTIR	Bool		
PARAR	Bool		
PARTIR_2	Bool		
PARAR_2	Bool		
Output			
LP_VERDE	Bool		
LP_ROJA	Bool		
LP_DESP_AUTO_L	Bool		
LP_DESP_AUTO_M	Bool		
EMISOR	Int		
InOut			
SALIDA	Bool		
Temp			
Constant			
Return			
PARTIR_PARAR	Void		

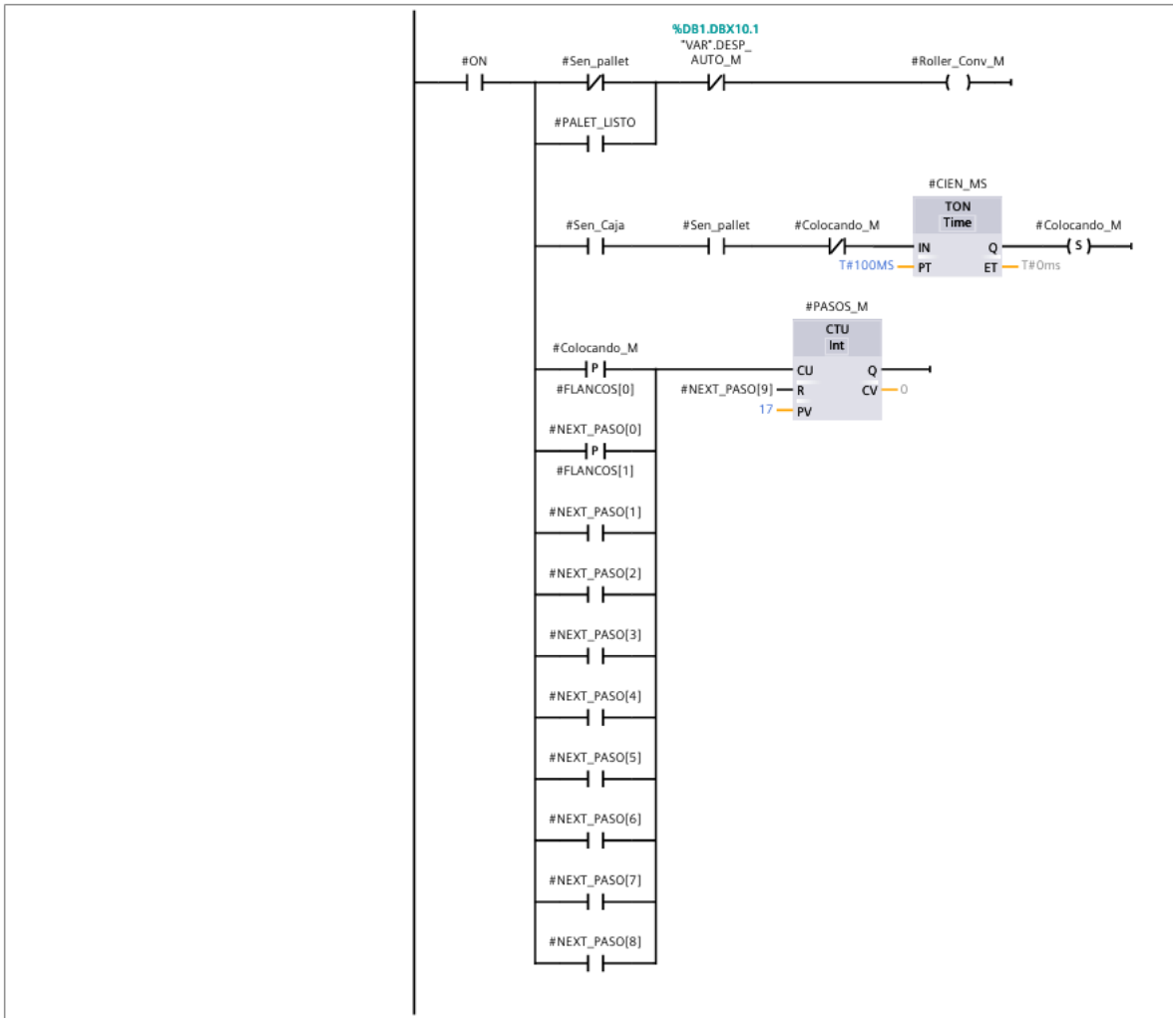
Figura 6.4

```
0001 IF #PARTIR OR #PARTIR_2 OR #SALIDA = TRUE AND #PARAR = TRUE AND #PARAR_2=FALSE THEN
0002   #SALIDA := TRUE;
0003 ELSE
0004   #SALIDA := FALSE;
0005 END_IF;//PARTIR PARAR
0006
0007 IF #SALIDA = TRUE THEN
0008   #LP_VERDE := TRUE;
0009   #LP_ROJA := FALSE;
0010   #LP_DESP_AUTO_L := "VAR".DESP_AUTO_L;
0011   #LP_DESP_AUTO_M := "VAR".DESP_AUTO_M;
0012 ELSE
0013   #LP_VERDE := FALSE;
0014   #LP_ROJA := TRUE;
0015 END_IF;//LUCES PILOTO
0016
0017 IF #SALIDA = TRUE AND "VAR".DESP_AUTO_L = FALSE AND "VAR".DESP_AUTO_M = FALSE THEN
0018   #EMISOR := 6;
0019 END_IF;
0020 IF #SALIDA = TRUE AND "VAR".DESP_AUTO_L = TRUE AND "VAR".DESP_AUTO_M = FALSE THEN
0021   #EMISOR := 2;
0022 END_IF;//CAJA A EMITIR
0023 IF #SALIDA = TRUE AND "VAR".DESP_AUTO_L = FALSE AND "VAR".DESP_AUTO_M = TRUE THEN
0024   #EMISOR := 4;
0025 END_IF;
0026
```

Figura 6.5

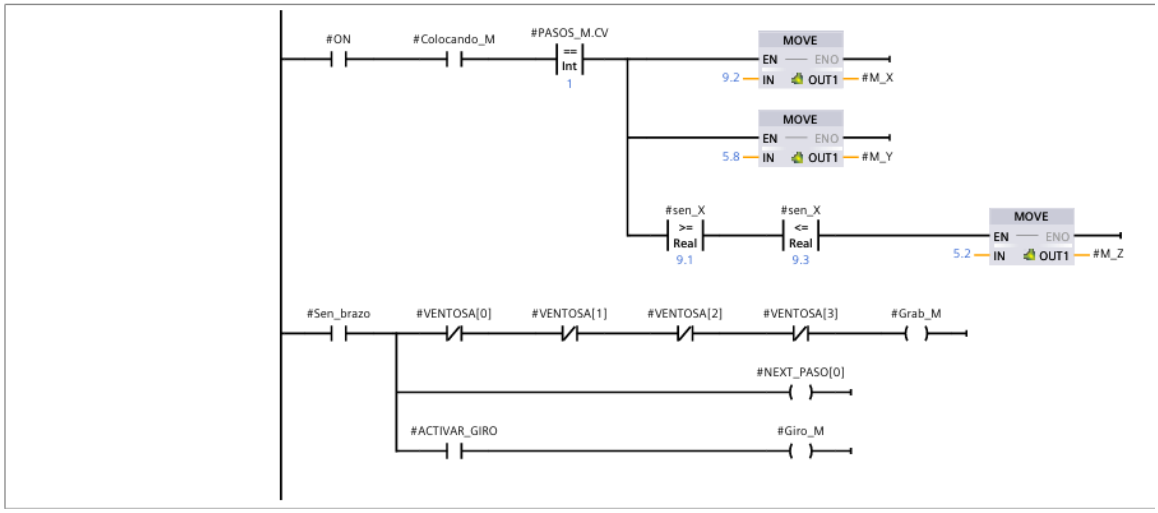
6.2.2. PALETIZADOR_M [FB3]

Network 1: INICIO DEL CICLO / PASOS / SET COLOCANDO



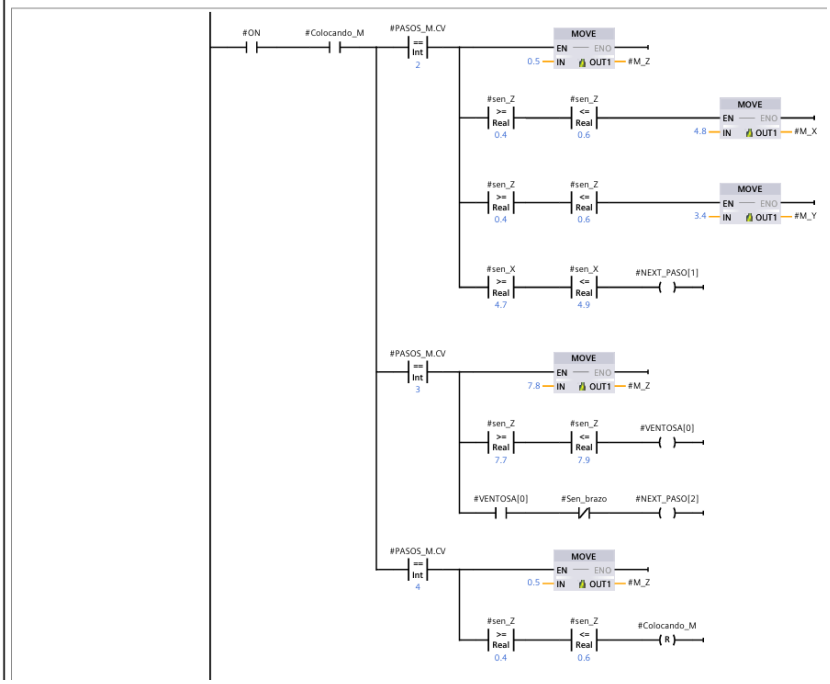
Network 2: TOMA DE CAJA / GIRO

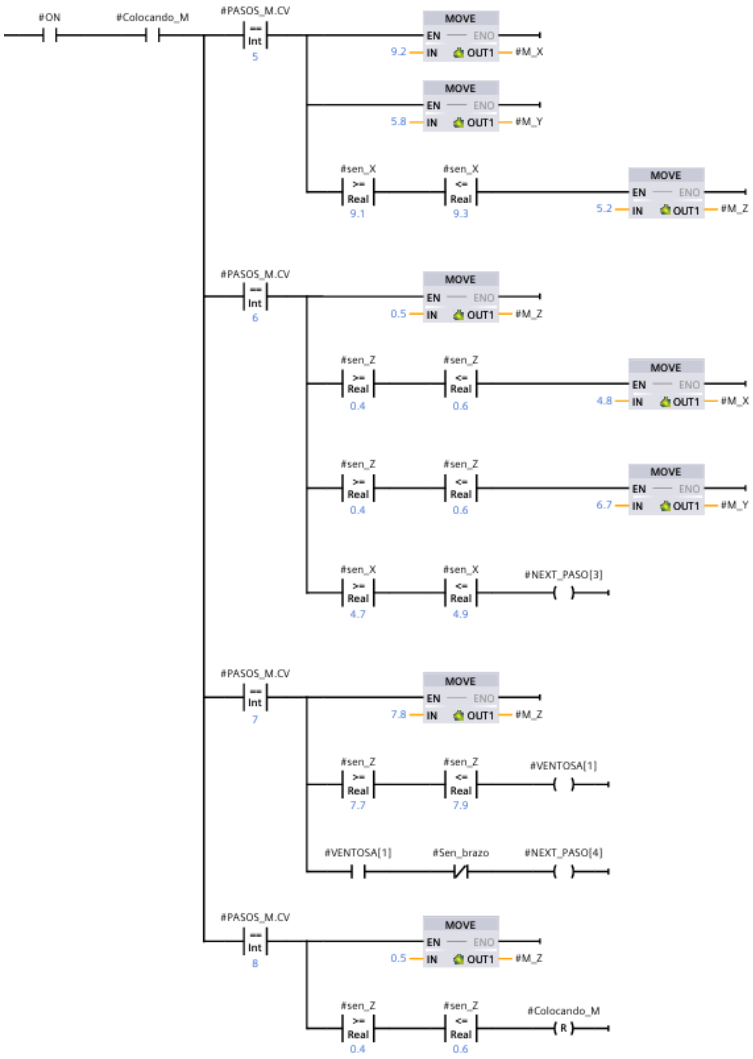
Totally Integrated Automation Portal	

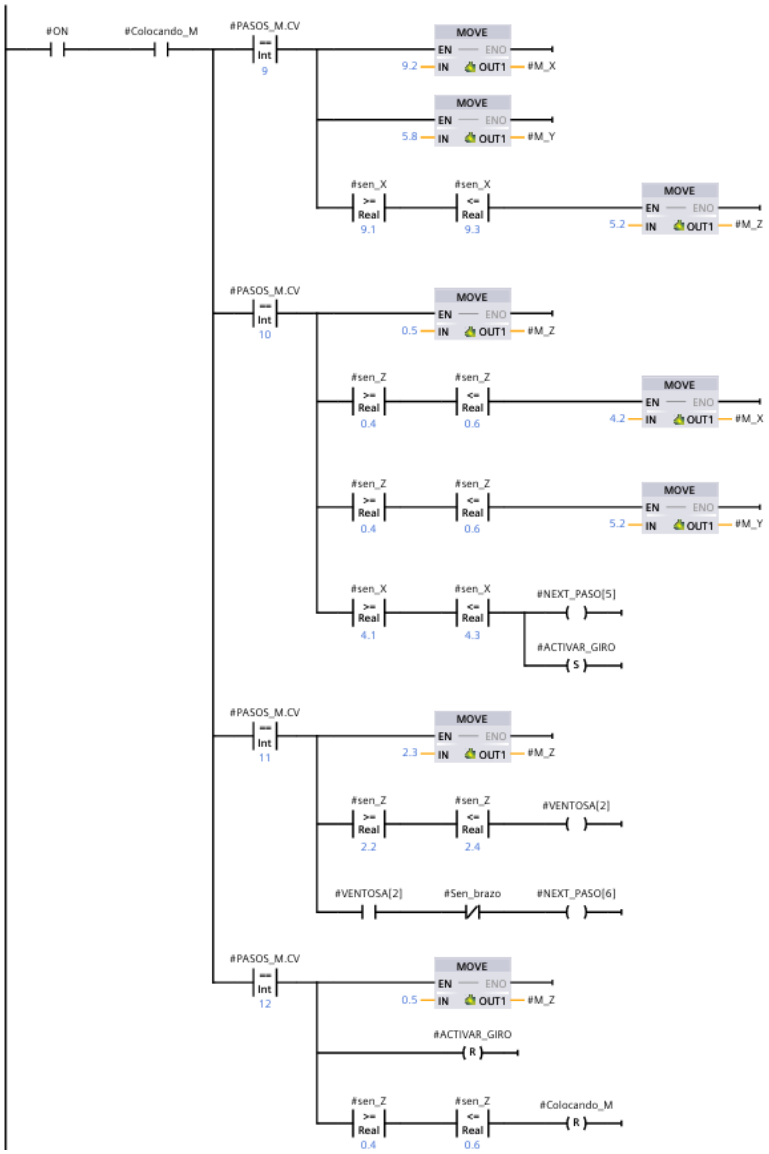


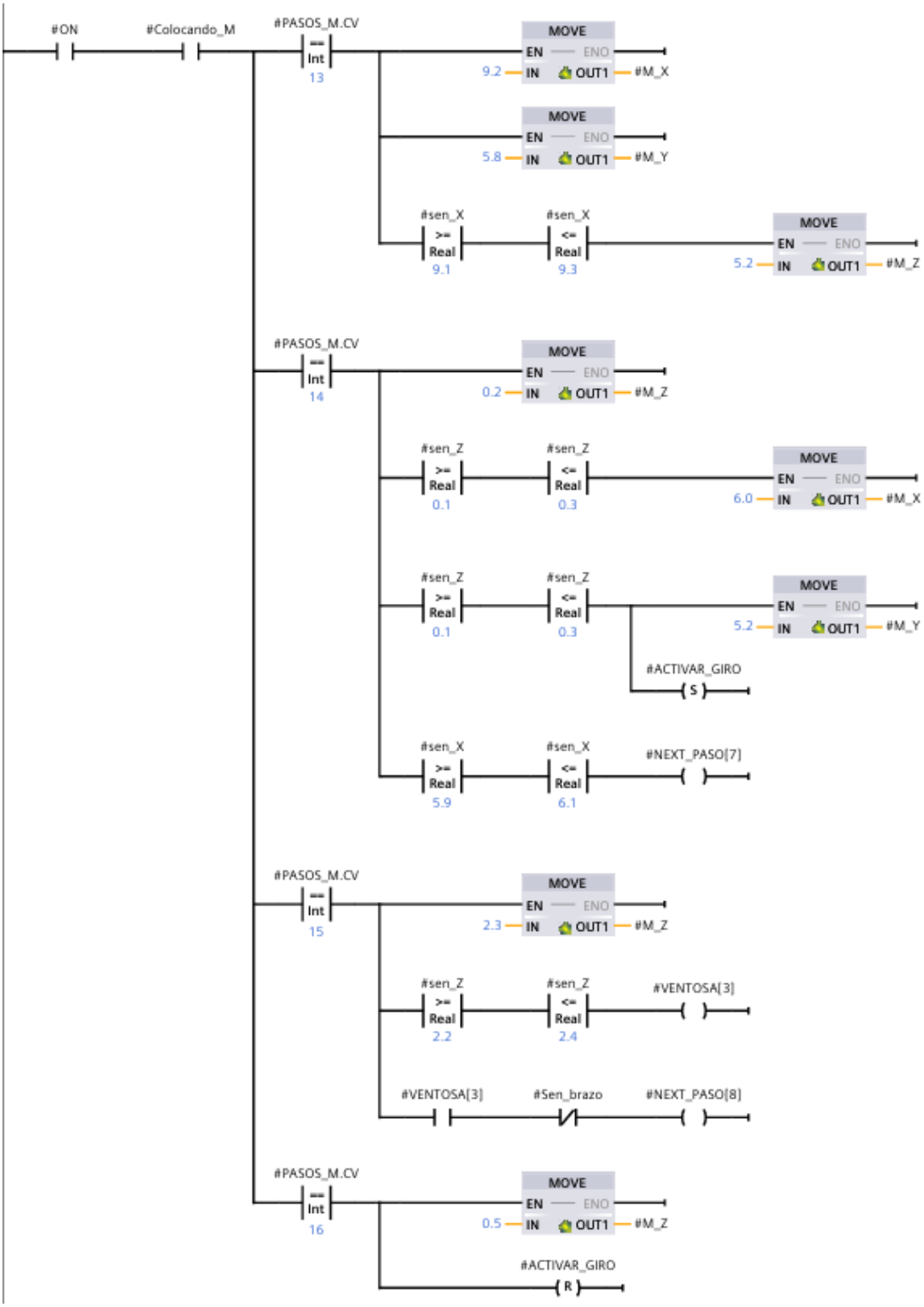
Network 3: COLOCAR CAJA 1

DEPOSITAR CAJA 1

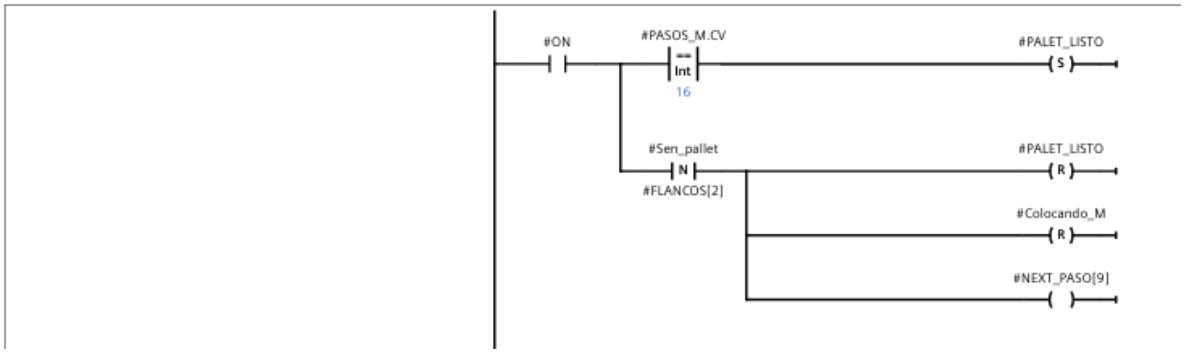



#ON
#Colocando_M
#PASOS_M.CV
==
Int
5
MOVE
EN
ENO
9.2
IN
OUT1
#M_X
MOVE
EN
ENO
5.8
IN
OUT1
#M_Y
#sen_X
>=
Real
9.1
#sen_X
<=
Real
9.3
MOVE
EN
ENO
5.2
IN
OUT1
#M_Z
#PASOS_M.CV
==
Int
6
MOVE
EN
ENO
0.5
IN
OUT1
#M_Z
#sen_Z
>=
Real
0.4
#sen_Z
<=
Real
0.6
MOVE
EN
ENO
4.8
IN
OUT1
#M_X
#sen_Z
>=
Real
0.4
#sen_Z
<=
Real
0.6
MOVE
EN
ENO
6.7
IN
OUT1
#M_Y
#sen_X
>=
Real
4.7
#sen_X
<=
Real
4.9
#NEXT_PASO[3]
#PASOS_M.CV
==
Int
7
MOVE
EN
ENO
7.8
IN
OUT1
#M_Z
#sen_Z
>=
Real
7.7
#sen_Z
<=
Real
7.9
#VENTOSA[1]
#VENTOSA[1]
#Sen_brazo
#NEXT_PASO[4]
#PASOS_M.CV
==
Int
8
MOVE
EN
ENO
0.5
IN
OUT1
#M_Z
#sen_Z
>=
Real
0.4
#sen_Z
<=
Real
0.6
#Colocando_M
R

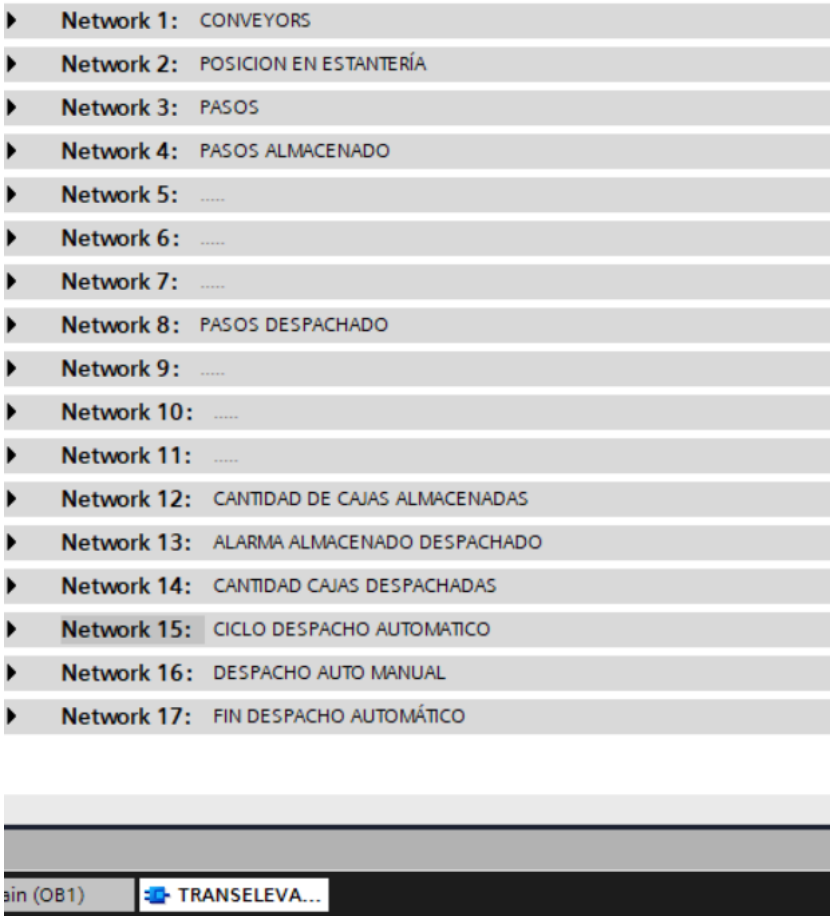

#ON
#Colocando_M
#PASOS_M.CV
==
Int
9
MOVE
EN
ENO
9.2
IN
OUT1
#M_X
MOVE
EN
ENO
5.8
IN
OUT1
#M_Y
#sen_X
>=
Real
9.1
#sen_X
<=
Real
9.3
MOVE
EN
ENO
5.2
IN
OUT1
#M_Z
#PASOS_M.CV
==
Int
10
MOVE
EN
ENO
0.5
IN
OUT1
#M_Z
#sen_Z
>=
Real
0.4
#sen_Z
<=
Real
0.6
MOVE
EN
ENO
4.2
IN
OUT1
#M_X
#sen_Z
>=
Real
0.4
#sen_Z
<=
Real
0.6
MOVE
EN
ENO
5.2
IN
OUT1
#M_Y
#sen_X
>=
Real
4.1
#sen_X
<=
Real
4.3
#NEXT_PASO[5]
#ACTIVAR_GIRO
S
#PASOS_M.CV
==
Int
11
MOVE
EN
ENO
2.3
IN
OUT1
#M_Z
#sen_Z
>=
Real
2.2
#sen_Z
<=
Real
2.4
#VENTOSA[2]
#VENTOSA[2]
#Sen_brazo
#NEXT_PASO[6]
#PASOS_M.CV
==
Int
12
MOVE
EN
ENO
0.5
IN
OUT1
#M_Z
#ACTIVAR_GIRO
R
#sen_Z
>=
Real
0.4
#sen_Z
<=
Real
0.6
#Colocando_M
R

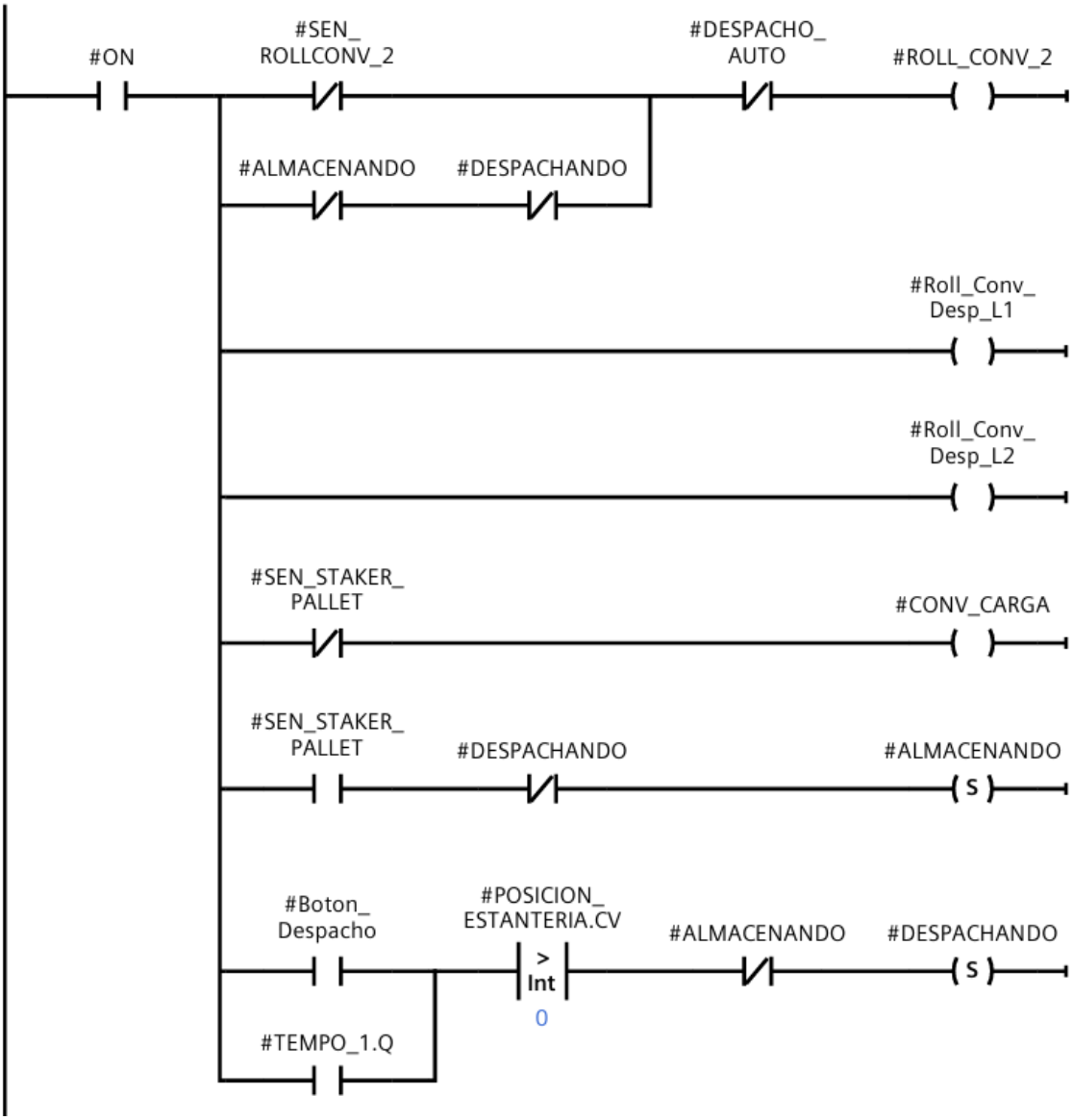

#ON
#Colocando_M
#PASOS_M.CV
== Int
13
MOVE
EN
ENO
9.2
IN
OUT1
#M_X
MOVE
EN
ENO
5.8
IN
OUT1
#M_Y
#sen_X
>= Real
9.1
#sen_X
<= Real
9.3
MOVE
EN
ENO
5.2
IN
OUT1
#M_Z
#PASOS_M.CV
== Int
14
MOVE
EN
ENO
0.2
IN
OUT1
#M_Z
#sen_Z
>= Real
0.1
#sen_Z
<= Real
0.3
MOVE
EN
ENO
6.0
IN
OUT1
#M_X
#sen_Z
>= Real
0.1
#sen_Z
<= Real
0.3
MOVE
EN
ENO
5.2
IN
OUT1
#M_Y
#ACTIVAR_GIRO
S
#sen_X
>= Real
5.9
#sen_X
<= Real
6.1
#NEXT_PASO[7]
#PASOS_M.CV
== Int
15
MOVE
EN
ENO
2.3
IN
OUT1
#M_Z
#sen_Z
>= Real
2.2
#sen_Z
<= Real
2.4
#VENTOSA[3]
#VENTOSA[3]
#Sen_brazo
#NEXT_PASO[8]
#PASOS_M.CV
== Int
16
MOVE
EN
ENO
0.5
IN
OUT1
#M_Z
#ACTIVAR_GIRO
R

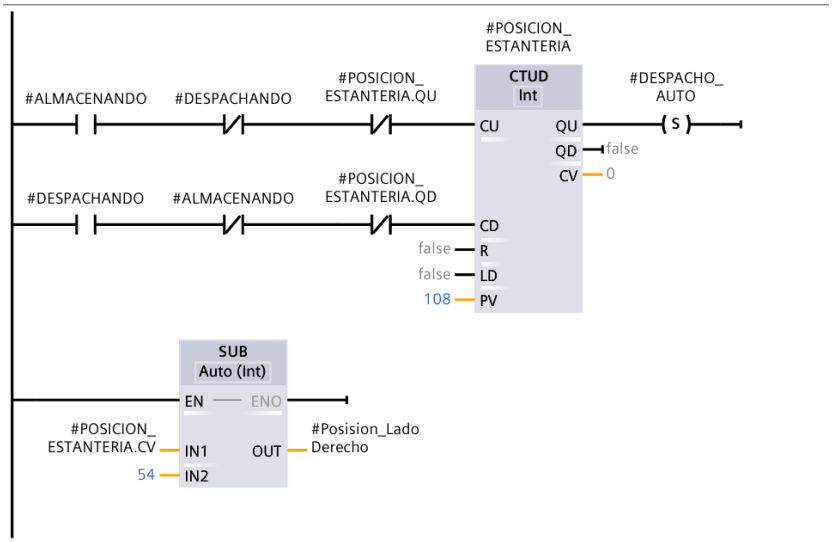
Network 7: RESET CICLO



6.2.3. TRANSELEVADORES [FB4]




#ON
#SEN_ ROLLCONV_2
#DESPACHO_ AUTO
#ROLL_CONV_2
#ALMACENANDO
#DESPACHANDO
#Roll_Conv_ Desp_L1
#Roll_Conv_ Desp_L2
#SEN_STAKER_ PALLET
#CONV_CARGA
#SEN_STAKER_ PALLET
#DESPACHANDO
#ALMACENANDO
S
#Boton_ Despacho
#POSICION_ ESTANTERIA.CV
>
Int
0
#ALMACENANDO
#DESPACHANDO
S
#TEMPO_1.Q


#POSICION_ ESTANTERIA
CTUD
Int
#ALMACENANDO
#DESPACHANDO
#POSICION_ ESTANTERIA.QU
CU
QU
QD
CV
false
0
#DESPACHO_ AUTO
S
#DESPACHANDO
#ALMACENANDO
#POSICION_ ESTANTERIA.QD
CD
false
R
false
LD
108
PV
SUB
Auto (Int)
EN
ENO
#POSICION_ ESTANTERIA.CV
IN1
54
IN2
OUT
#Posision_Lado Derecho

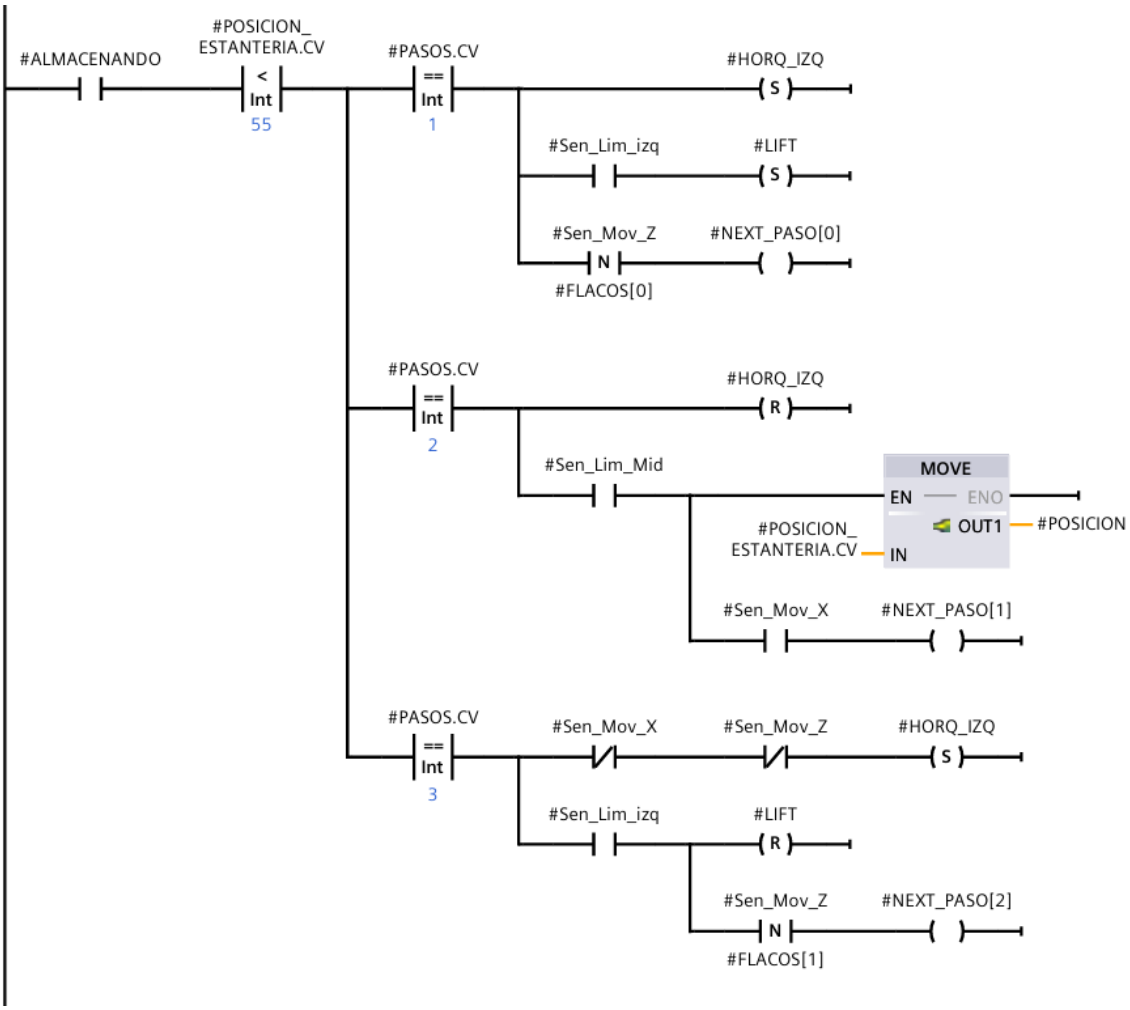


Figura 6.6: Net. 4 FB4

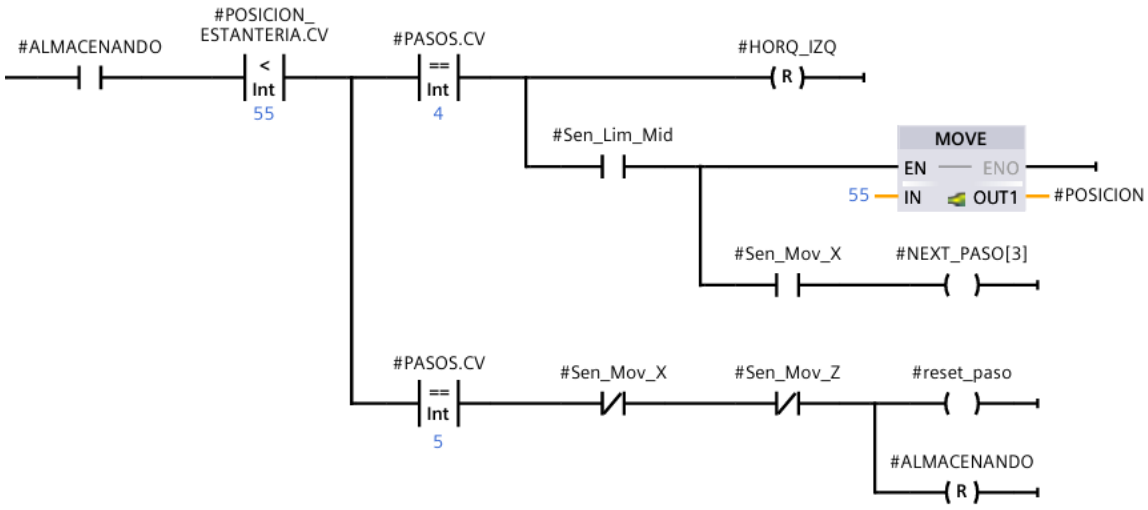
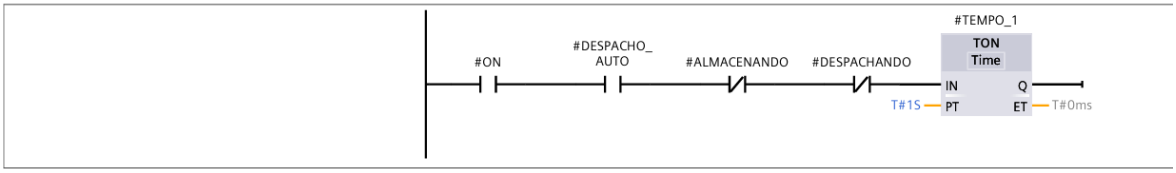
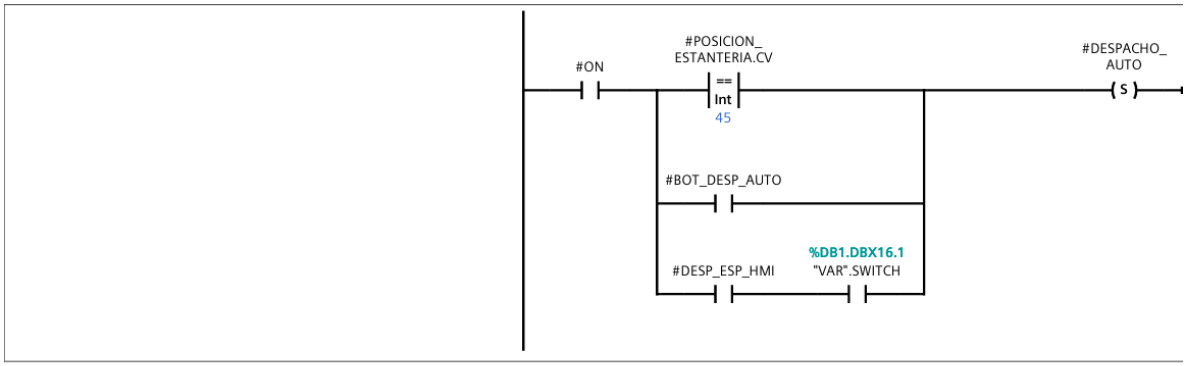


Figura 6.7: Net. 5 FB4

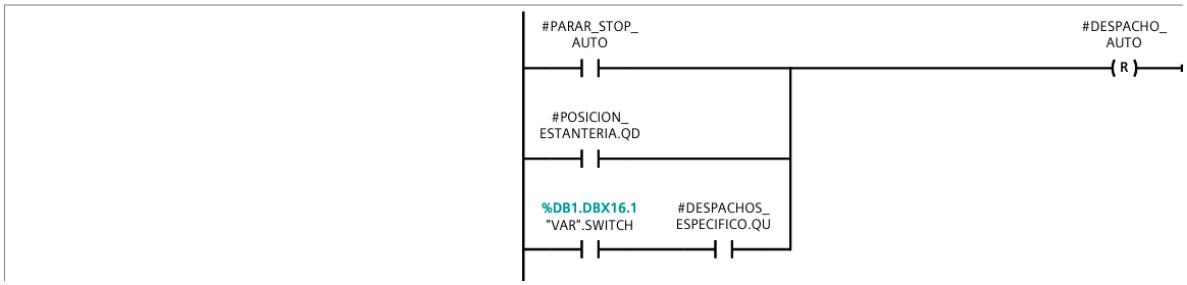
Network 15: CICLO DESPACHO AUTOMATICO



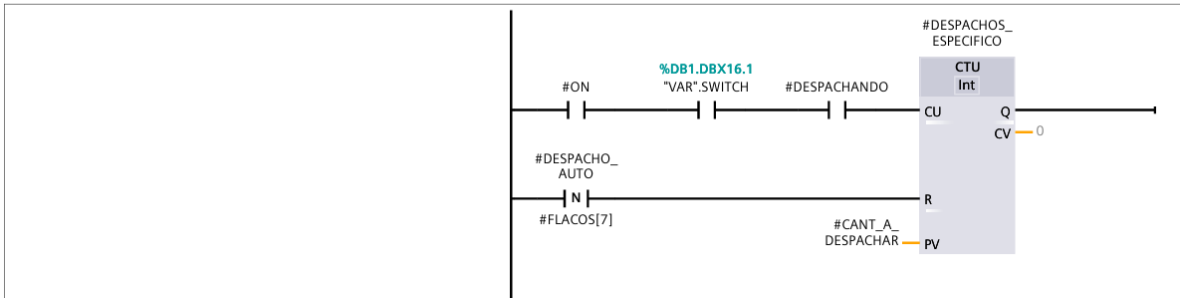
Network 16: DESPACHO AUTO MANUAL



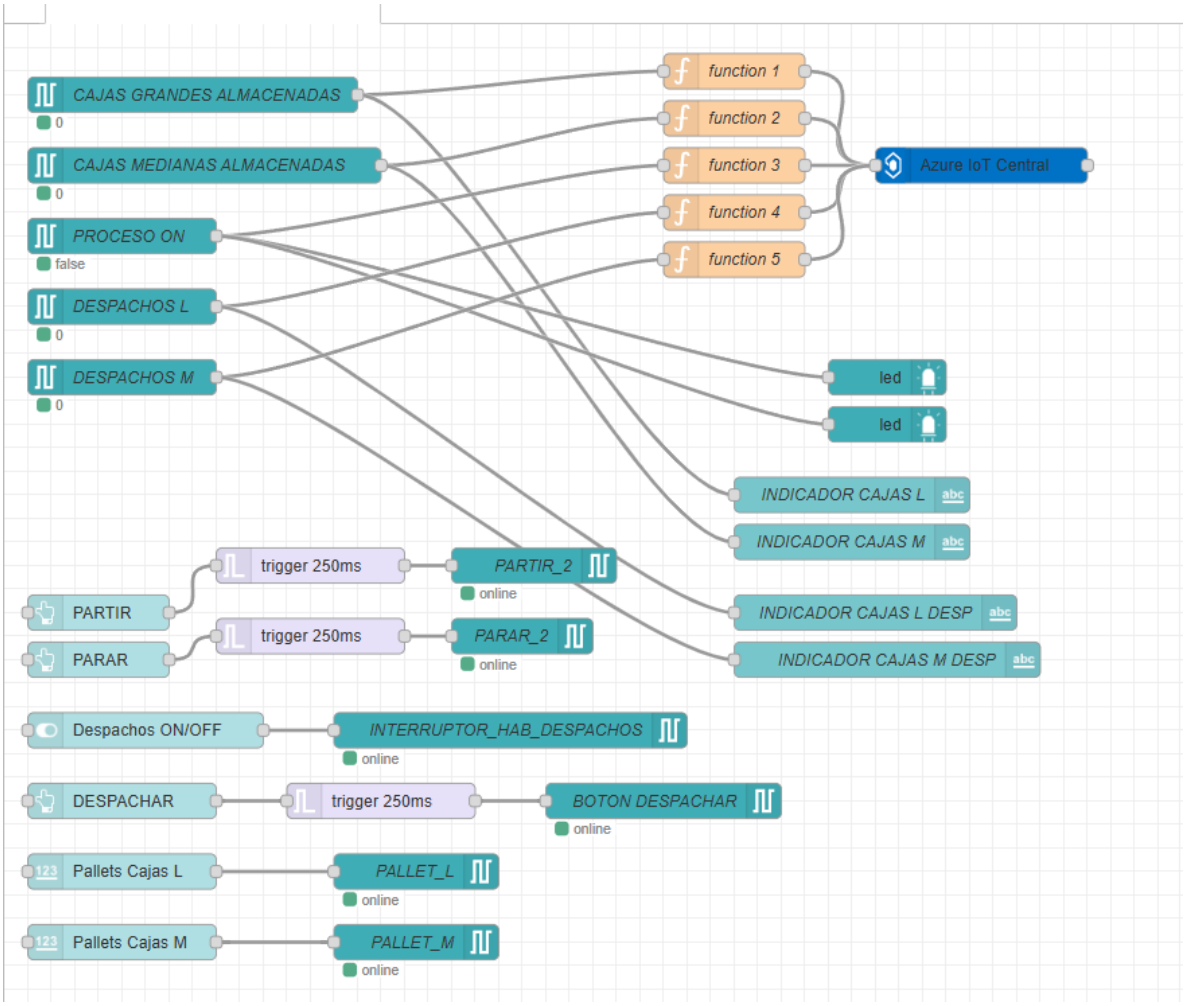
Network 17: FIN DESPACHO AUTOMÁTICO



Network 18:



6.3. Flujo Node Red



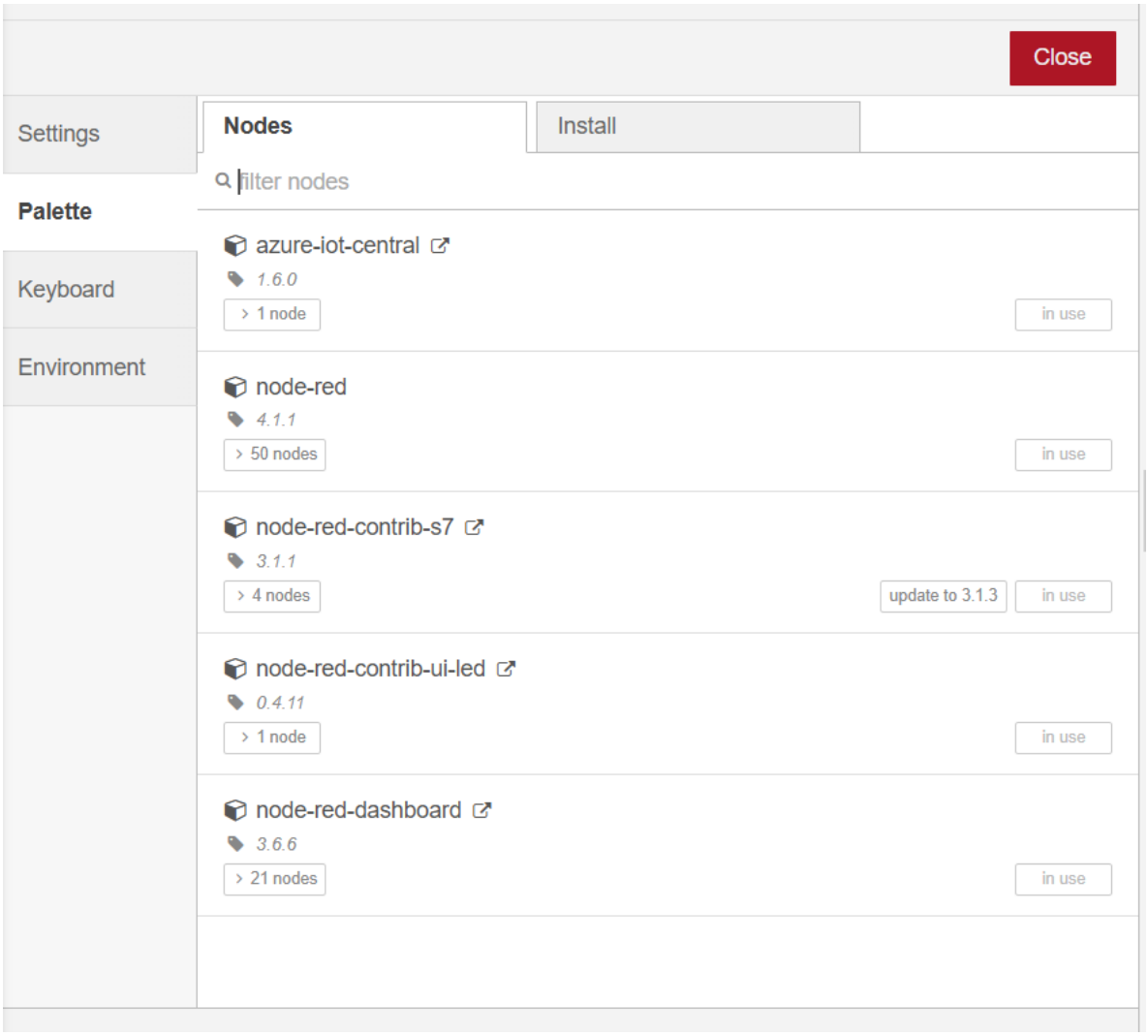
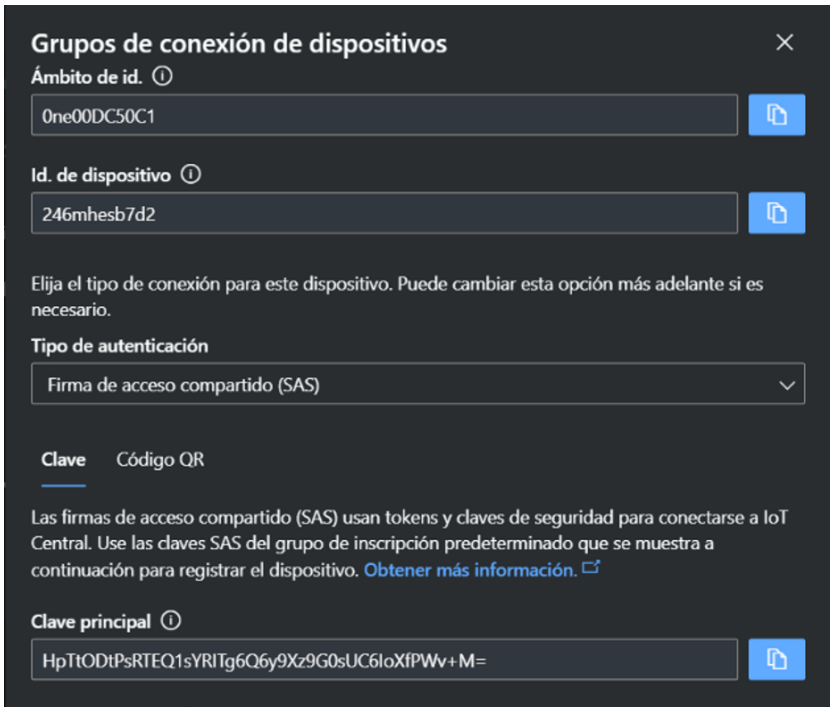
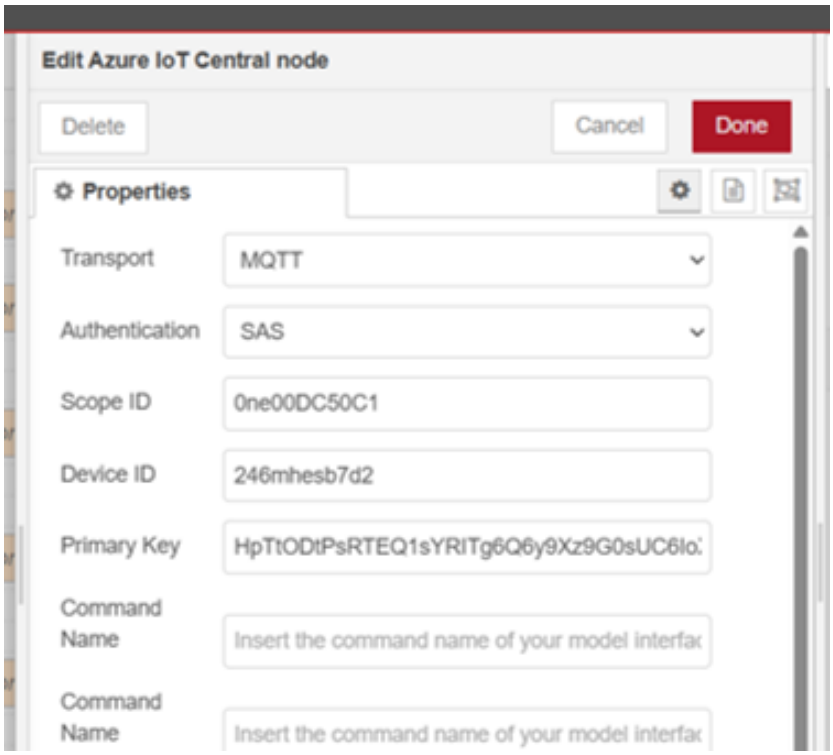


Figura 6.8: Librerías instaladas en NodeRed


Grupos de conexión de dispositivos
Ámbito de id.
0ne00DC50C1
Id. de dispositivo
246mhesb7d2
Elija el tipo de conexión para este dispositivo. Puede cambiar esta opción más adelante si es necesario.
Tipo de autenticación
Firma de acceso compartido (SAS)
Clave
Código QR
Las firmas de acceso compartido (SAS) usan tokens y claves de seguridad para conectarse a IoT Central. Use las claves SAS del grupo de inscripción predeterminado que se muestra a continuación para registrar el dispositivo. Obtener más información.
Clave principal
HpTtODtPsRTEQ1sYRITg6Q6y9Xz9G0sUC6IoXfPWv+M=


Edit Azure IoT Central node
Delete
Cancel
Done
Properties
Transport
MQTT
Authentication
SAS
Scope ID
0ne00DC50C1
Device ID
246mhesb7d2
Primary Key
HpTtODtPsRTEQ1sYRITg6Q6y9Xz9G0sUC6Io
Command Name
Insert the command name of your model interfac
Command Name
Insert the command name of your model interfac

7. Referencias

- Del Castillo Galarza, S. (2023). *Logística 4.0: Innovación y Eficiencia en la Cadena de Suministro.* Doxa Edition.
- Escudero Serrano, M.J.(2019). *Logística de almacenamiento 2.ª edición.* Ediciones Paraninfo, S.A.
- Node-RED Community. (s.f.). *node-red-contrib-s7 (Versión 3.1.3).* Node-RED Library. `https://flows.nodered.org/node/node-red-contrib-s7`.
- Node-RED Community. (s.f.). *node-red-azure-iot-central (Versión 0.0.2).* Node-RED Library. `https://flows.nodered.org/node/node-red-azure-iot-central`.
- Node-RED Community. (s.f.). *node-red-dashboard (Versión 3.6.6).* Node-RED Library. `https://flows.nodered.org/node/node-red-dashboard`.
- Siemens AG. (s.f.). *Programming Guideline for S7-1200/1500.* Siemens Industry Online Support. `https://cache.industry.siemens.com/dl/files/040/90885040/att_970576/v1/81318674_Programming_guideline_DOC_v16_en.pdf`.
- Camargo Barrera, H.A. [Alejandro Camargo Carrera] (15 de abril de 2024). *Tutorial conexión PLC SIM con Azure IoT* [Archivo de Video]. Youtube. `https://www.youtube.com/watch?v=vmuNHL0IYw0`.
- Factory I/O. (s.f.). Manual. `https://docs.factoryio.com/manual/`.