



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



ESCUELA TÉCNICA
SUPERIOR INGENIEROS
INDUSTRIALES VALENCIA

TRABAJO FIN DE GRADO EN INGENIERÍA EN TECNOLOGÍAS INDUSTRIALES

**PROYECTO DE AUTOMATIZACIÓN DE UN
SISTEMA DE ALMACENAMIENTO DE
PIEZAS CON PLC TSX-PREMIUM Y M241
DE SCHNEIDER Y SCADA MEDIANTE
APLICACIÓN LABVIEW.**

AUTOR: PABLO GÓMEZ GALLARDO

TUTOR: RAÚL SIMARRO FERNÁNDEZ

Curso Académico: 2019-20

AGRADECIMIENTOS

A mi tutor Raúl Simarro por guiarme y enseñarme durante todo el trabajo, por su paciencia y por su disposición a resolver todas las dudas que me fueran surgiendo.

A mis amigos y compañeros que me ayudaron a seguir en los momentos más difíciles y que hacen que todo sea mucho más fácil.

A mis padres por darme siempre todo su apoyo y confianza sobre todo estos últimos cuatro años.

Por último, quiero agradecer a los técnicos del Departamento la ayuda prestada y su paciencia aquellas veces que tenían que abrirnos la clase o el almacén, y no han sido pocas.

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

RESUMEN

En este proyecto se ha diseñado la automatización de un sistema de almacenamiento de piezas en el que intervienen un almacén con manipuladores automáticos y un brazo robot con ventosa. Cada uno de estos dos elementos está gobernado por un PLC distinto (TSX-Premium y M241), ambos de la marca Schneider. Se deberá programar y configurar las comunicaciones entre los dos autómatas mediante el acceso a la memoria compartida del PLC a través de la red. La programación de los automatismos se realiza en dos softwares diferentes: Unity Pro y Somachine, este último utiliza un entorno multiplataforma CODESYS, mediante lenguajes SFC, LD, ST Y FBD.

Se debe realizar un programa que reconozca las piezas por su color mediante una cámara motorizada y que permita la comunicación de esa información con los autómatas a través de un servidor OPC. Además, se realizará una aplicación SCADA en LabVIEW que permitirá la monitorización del sistema y el control del usuario utilizando el servidor OPC creado, para sus comunicaciones.

Se probará el sistema desarrollado sobre maquetas de planta para evaluar y comprobar el correcto funcionamiento implementado. Estas maquetas serán proporcionadas por el Departamento de Ingeniería de Sistemas y Automática (DISA).

Palabras Clave: Automatización, PLC, SCADA, Schneider.

ABSTRACT

In this project the automation of a parts storage system has been designed, involving a warehouse with automatic manipulators and a robot arm with a suction cup. Each of these two elements is governed by a different PLC (TSX-Premium and M241), both of the Schneider brand. Communications between the two PLCs must be programmed and configured by accessing the shared memory of the PLC through the network. The programming of the automatism will be carried out in two different software: Unity Pro and Somachine that uses a multiplatform CODESYS environment, using SFC, LD, ST and FBD languages

A program must be carried out that recognizes the pieces by their color, by means of a motorized camera, and that allows the communication with the automatons through the OPC server. In addition, a SCADA application will be made in LabVIEW and will allow the monitoring of the system and user interaction using the OPC server created, for its communications.

The system developed on plant models will be tested to evaluate and verify the correct functioning implemented. These models will be provided by the Department of Systems Engineering and Automation (DISA).

Keywords: Automation, PLC, Schneider, SCADA.

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

DOCUMENTOS DEL PROYECTO

MEMORIA	12
ANEXO 1: DECLARACIÓN DE VARIABLES	52
ANEXO 2: DISEÑO DEL AUTOMATISMO	62
MANUAL DE USUARIO	82
PRESUPUESTO	98

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

ÍNDICE

MEMORIA	12
CAPÍTULO 1: INTRODUCCIÓN.....	14
1.1. Objetivos.....	14
1.2. Antecedentes	15
1.3 Motivación y Justificación.....	15
1.4 Especificaciones del proceso.....	16
1.5 Normativa	17
CAPÍTULO 2: ESTUDIO DE ALTERNATIVAS Y SOLUCIÓN ADOPTADA.....	18
2.1 Simuladores	18
2.1.2 PLC Y Software de Programación	19
2.3 Lenguajes de programación.....	21
2.4 Comunicaciones	22
2.4.1 Comunicación entre Autómatas.....	22
2.4.2 Comunicación con el control y supervisión del proceso.....	23
2.5 Control y supervisión.....	23
CAPÍTULO 3: DESCRIPCIÓN DETALLADA DEL EQUIPO DE TRABAJO	24
3.1 Brazo ventosa.....	24
3.2 Almacén vertical.....	25
3.3 Cámara Kinect	26
3.4 Depósitos.....	27
CAPÍTULO 4: FUNCIONAMIENTO Y SOLUCION DETALLADA.....	28
4.1 Diagrama de flujo de comunicaciones del proyecto.....	28
4.2 Distribución y recorrido de las piezas.....	29
4.3 Calibración de los equipos	30
4.3 Programación del Proceso	31
4.3.1 Unity Pro - Almacén Vertical.....	31
4.3.2 Somachine - Brazo Ventosa	38
4.4 Comunicación entre autómatas.....	40
4.5 Servidor OPC.....	42
4.6 SCADA.....	44
CONCLUSIONES.....	48
BIBLIOGRAFÍA.....	50
ANEXO 1: DECLARACIÓN DE VARIABLES.....	52
1.INTRODUCCIÓN	54
2. VARIABLES EN UNITY PRO.....	55
3. VARIABLES EN SOMACHINE	59

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

ANEXO 2: DISEÑO DEL AUTOMATISMO 62

1. INTRODUCCIÓN 64

2. UNITY PRO - ALMACÉN. 65

2.1 Grafcet Principal (Z)..... 66

2.2 Volver a posición inicial (L)..... 69

2.3 Meter/Sacar (T)..... 70

2.4 Celda en proceso (K)..... 71

2.5 Estado celda (F)..... 71

2.6 Estado almacén (A)..... 72

2.7 Color Celda (C)..... 73

2.8 Preferencia meter (P)..... 74

2.9 Preferencia Sacar (S)..... 75

2.10 Modos de funcionamiento y emergencia. 76

3. SOMACHINE - BRAZO VENTOSA..... 77

3.1 Grafcet Principal de movimientos (B)..... 77

3.2 Grafcet compresor y ventosa (V)..... 78

3.3 Volver a posición inicial (J)..... 79

3.4 Modos de Funcionamiento y Emergencia..... 79

MANUAL DE USUARIO..... 82

1. INTRODUCCIÓN 84

2. PUESTA EN MARCHA..... 84

2.1. Hardware 84

2.2 Software 85

2.2.1 Servidor OPC- KepServerEX..... 86

2.2.2 Almacén Vertical- Unity Pro..... 87

2.2.3 Brazo Ventosa - Somachine 88

2.2.4 LabVIEW..... 89

3. FUNCIONAMIENTO DEL SCADA..... 90

3.1 Meter Pieza..... 90

3.2 Sacar Pieza 92

3.3 Estados del almacén..... 93

3.4 Modos de funcionamiento y parada de emergencia..... 93

PRESUPUESTO 98

1. INTRODUCCIÓN: 100

2. CUADRO Nº1: MANO DE OBRA..... 100

3. CUADRO Nº2: PRECIO DE LOS MATERIALES 100

Hardware..... 100

Software 101

Total 102

4. CUADRO Nº3: PRECIOS UNITARIOS 102

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

5. CUADRO N°4: PRECIOS DESCOMPUESTOS	102
Estudio de alternativas.....	102
Diseño del automatismo	103
Programación del automatismo	103
Desarrollo e implementación del control.....	103
Elaboración del documento del Proyecto	104
 6. PRESUPUESTO DE EJECUCIÓN Y POR CONTRATA	 104

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



ESCUELA TÉCNICA
SUPERIOR INGENIEROS
INDUSTRIALES VALENCIA

TRABAJO FIN DE GRADO EN INGENIERÍA EN TECNOLOGÍAS INDUSTRIALES

**PROYECTO DE AUTOMATIZACIÓN DE UN
SISTEMA DE ALMACENAMIENTO DE
PIEZAS CON PLC TSX-PREMIUM Y M241 DE
SCHNEIDER Y SCADA MEDIANTE
APLICACIÓN LABVIEW.**

MEMORIA

AUTOR: PABLO GÓMEZ GALLARDO

TUTOR: RAÚL SIMARRO FERNÁNDEZ

Curso Académico: 2019-20

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

CAPÍTULO 1: INTRODUCCIÓN

1.1. Objetivos

El objetivo del proyecto consiste en la automatización de un sistema de almacenamiento de piezas que el usuario controlará desde una aplicación SCADA. Se proporcionarán diferentes métodos para la ordenación y extracción de las piezas, un sistema de paro de emergencia y la posibilidad de un control manual del proceso.

El cliente cuenta en su industria con las máquinas reales y es tarea del ingeniero la programación del proceso y el control, utilizando simuladores de planta para la comprobación del buen funcionamiento. Se han utilizado un almacén automatizado y un brazo ventosa, ambas maquetas proporcionadas por el Departamento de Ingeniería de Sistemas y Automática (DISA) de la Universidad que se conectarán a dos autómatas diferentes además de una cámara que se configurará para el reconocimiento del color de las piezas. Todos, tendrán que comunicarse entre ellos para la obtención de un resultado satisfactorio. Dicha comunicación incluye un servidor OPC y comunicación entre autómatas a través de red con servicio de Maestro/Esclavo.

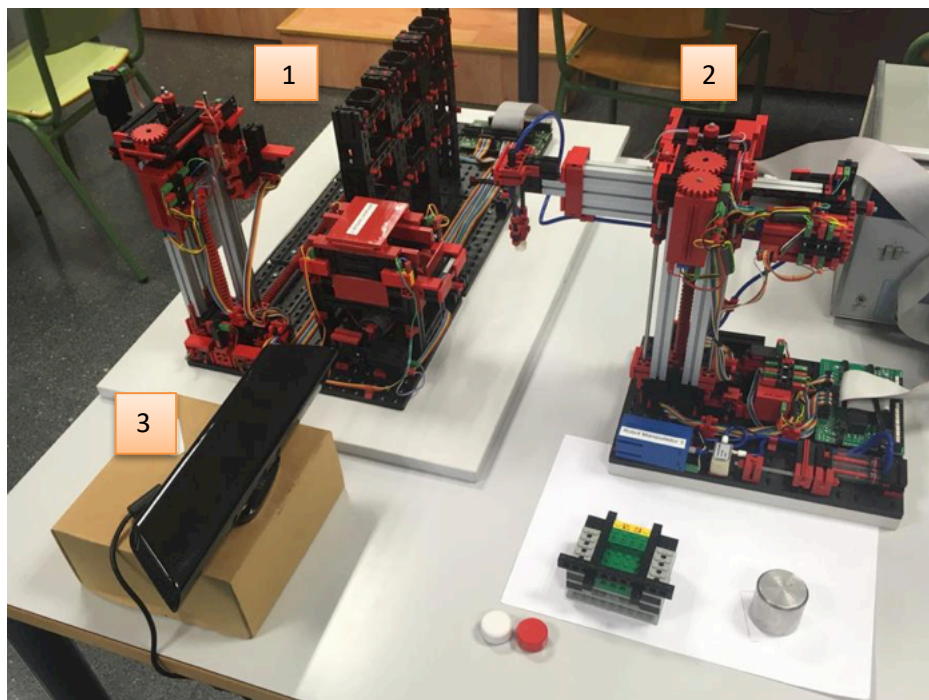


Figura 1. Distribución de las maquetas. Fuente: elaboración propia

1	Almacén elevado
2	Brazo ventosa
3	Cámara de reconocimiento

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

1.2. Antecedentes.

Hoy en día se está asistiendo a una nueva revolución del modelo Industrial establecido, concretamente a la cuarta, que dará paso a industrias completamente digitalizadas, automatizadas y conectadas a las que se ha llamado Las Industrias 4.0. Es de especial interés que no solo está al alcance de unas pocas empresas privilegiadas con grandes recursos, sino que se encuentra disponible para pequeñas empresas que quieran obtener resultados más eficientes y competitivos y así poder crecer en un mercado cada vez más global. (Industria 4.0, la cuarta revolución industrial y la inteligencia operacional, 2019)

Uno de los efectos más claros lo encontramos en el almacenamiento y distribución de productos que en muchas empresas se encuentran ya totalmente robotizados y automatizados lo que supone una gran ventaja frente a un competidor que continua con un modelo más manual. Este trabajo pretende ser un primer paso para dar una solución a la creciente demanda de empresas que buscan esta transición y aun con las limitaciones que este trabajo puede suponer, significa toda una declaración de intenciones en el estudio y accesibilidad a estas Fábricas del futuro en servicio de la sociedad (Carlotti, 2019).

1.3 Motivación y Justificación

El enorme terreno por descubrir que supone esta nueva revolución industrial y su importancia histórica, convierten estas nuevas tecnologías en un tema apasionante de estudio y dedicación al que se le suma el interés suscitado durante la carrera a través de asignaturas como Informática, Sistemas Automáticos y Tecnología Automática y la optativa de Laboratorio de Automatización y Control. Esta última asignatura, se puede cursar durante el segundo cuatrimestre de 4º curso. En mi caso se cursó con el objetivo de profundizar en los conocimientos de las materias previamente nombradas y finalmente llevar a cabo el Trabajo de Fin de Grado sobre algún tema que en esta se estudiase.

Durante la asignatura se realizaron cuatro seminarios: Robótica, Instrumentación, Control y Automatización. Fue sobre Automatización, el Trabajo que se eligió llevar a cabo para la evaluación de dicha asignatura en aras de realizar un TFG que supusiera una continuación, avance y profundización de este. Durante este Trabajo, se sentaron las bases teóricas y los primeros acercamientos a las herramientas como softwares, PLC's y simuladores que posteriormente, culminarían en este Proyecto.

La posibilidad de aprender algunos softwares de programación como son: Somachine, Unity Pro y LabVIEW altamente demandados hoy en día por las empresas de este sector, incrementa aún más la motivación en la realización de este proyecto.

1.4 Especificaciones del proceso

Las especificaciones mínimas que el cliente demanda y que configuran el funcionamiento del almacén, teniendo presente las limitaciones que supone la realización del proyecto con simuladores a pequeña escala, son:

1. Todas las decisiones del usuario sobre el proceso, se llevarán a cabo a través de una pantalla que contenga el SCADA, donde también se mostrarán los cambios, avisos y estados que durante el proceso se produzcan como emergencias, celdas libres del almacén etc.
2. Sobre un depósito fijo, el cliente dejará las piezas/productos que se quieran almacenar. A su lado habrá otro depósito donde el sistema colocará las piezas que se extraigan.
3. La parte encargada de recoger y dejar las piezas es el brazo robot que conecta estos depósitos con la entrada del almacén donde esperan las cajas a que se extraiga o introduzca una pieza. El mecanismo de entrada consiste en una cinta por donde viajan las cajas y que comunica la entrada mencionada con el interior del almacén donde un robot móvil se encarga de transportarlas desde o hasta la celda correspondiente.
4. Tres modos de funcionamiento: automático, paro y manual. El arranque se hace desde paro y supone el fin de un ciclo del proceso, es decir, cuando se active se terminará de meter/sacar la pieza que en ese momento se esté transportando y luego permanecerá en reposo. El modo manual detiene el proceso instantáneamente y a través del SCADA podremos manipular las salidas de los dos elementos de manera manual. Posteriormente el proceso comenzaría desde el principio de nuevo.
5. Paro de emergencia: Dos setas de emergencia que permitan detener el proceso de manera instantánea activando una luz de emergencia. En el caso del almacén será necesario para volver al funcionamiento automático pasar antes por el modo manual y colocar la caja que se ha quedado a medio camino en un punto concreto que en el Manual de Usuario se detalla. La peculiaridad de la emergencia en el brazo es que mientras transporta una pieza no pueden desactivarse todas las salidas ya que la pieza caería y probablemente se dañaría por tanto debe diseñarse una opción que resuelva este problema.

Las posibilidades para almacenar piezas, en lugares determinados que permitan llevar un cierto orden de los productos que se van dejando serán las siguientes:

- Almacenar según el tipo de pieza: esto se llevará a cabo por una cámara programada para diferenciar entre tres tipos de colores (rojo, azul y blanco). Posteriormente se colocarán las piezas en las posiciones reservadas a dichos colores.
- Un método rápido de inserción de piezas que las va colocando en las primeras posiciones disponibles del almacén sin tener en cuenta su color, aunque este quedará registrado.

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

Las opciones para la extracción de las piezas son:

- Extracción manual, a través de una pantalla esquemática del almacén donde aparece el estado de la celda (vacía, pieza roja, azul o blanca) el usuario con esta información elige la pieza que desea liberar.
- Vaciado automático: un vaciado completo del almacén de manera automática sin distinguir el tipo de pieza que se extrae y siguiendo un orden que en este caso coincide con el del almacenamiento del método rápido.

Se proporciona información más detallada sobre los métodos de ordenación en el documento Manual del Usuario.

1.5 Normativa.

Para la realización del Proyecto se ha tenido en cuenta la legislación actual en aquellos aspectos que afectan al Trabajo. Durante el documento se indicarán que partes están sometidas a esta legislación. Las normas seguidas son las siguientes:

EN ISO 13849-1:2006. Seguridad de las máquinas. Partes del sistema de mando relativas a seguridad. Parte 1: Principios generales para el diseño (Gutierrez, 2019)

EN ISO 13849-2:2004. Seguridad de las máquinas. Partes del sistema de mando relativas a seguridad. Parte 2: Validación (Gutierrez, 2019).

IEC 61131-3. Estandarización de los lenguajes de programación para los controladores lógicos programables (IEC, 2018b).

UNE-EN 60848:2013. Lenguaje de especificación GRAFCET para diagramas funcionales secuenciales (AENOR, 2016).

IEC 62541. Especificaciones de la arquitectura unificada OPC (IEC, 2018c).

IEC 60870-5-104. Definición del uso de una red TCP/IP como medio de comunicación (IEC, 2018a).

CAPÍTULO 2: ESTUDIO DE ALTERNATIVAS Y SOLUCIÓN ADOPTADA

Para conseguir una solución viable al problema, durante la fase preliminar de este proyecto, se contemplaron varias alternativas de trabajo a todas las partes que conforman el proyecto teniendo en cuenta qué equipos se hallaban disponibles en el DISA, sus ventajas y sus inconvenientes. Durante este capítulo también se irán describiendo que elementos se escogían y el por qué.

2.1 Simuladores

Para simular el proyecto y comprobar que lo que se iba diseñando funcionaba correctamente a la vez que se descubrían y corregían errores que iban apareciendo sin la necesidad de hacerlo con equipos reales con el riesgo y coste que ello conlleva. Se propusieron estas dos alternativas:

- **Factory I/O:** Este software brinda la posibilidad de diseñar procesos industriales y automatizarlos a través de autómatas, microcontroladores y FPGA y visualizar su funcionamiento por pantalla en tiempo real a modo de videojuego. Sus ventajas son sus altos gráficos y calidad de sonido que proporcionan al usuario una experiencia hiperrealista con simulaciones de gran precisión. Como inconveniente, su limitación en cuanto a maquinas/procesos disponibles. (I/O, 2019)
- **Maquetas FischerTechnik:** Esta marca alemana, se dedica a la fabricación de juguetes robotizados formados por sistemas mecánicos y electrónicos sobre placas de madera que imitan a escala, sistemas industriales reales. Gracias a su flexibilidad y funcionalidad son ampliamente utilizados en la educación y formación y es por eso que el DISA cuenta en sus almacenes con una gran cantidad de estas maquetas. Proporcionan un gran abanico de posibilidades en cuanto a que proceso se quiere simular. En este caso se contaba con un almacén vertical que coincidía con las especificaciones del cliente. Puntos a favor son la diversidad de procesos y la sensación de crear algo que es tangible en la realidad. El principal inconveniente es los problemas mecánicos que surgen debido al alto uso que estos elementos reciben.
- **Cámara Kinect:** cámara utilizada en consolas que presenta buenas características y que podría ser utilizada mediante un programa para detectar colores lo que simularía el reconocimiento de distintos tipos de pieza.

Debido a las limitaciones comentadas sobre el Factory I/O y a la existencia dentro del Departamento de un almacén. Se decidió finalmente emplear una maqueta de FischerTechnik junto con la cámara Kinect ya que de esta última no se encontraron alternativas

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

2.1.2 PLC Y Software de Programación

Un PLC o autómatas programable es un sistema electrónico utilizado normalmente en un ambiente industrial. Emplea una memoria programable para el almacenamiento interno de instrucciones orientadas al usuario, con el propósito de controlar a través de entradas y salidas (digitales y/o analógicas) los distintos procesos industriales (Wikipedia, 2019).

Como alternativa a estos dispositivos se encuentran las tarjetas de adquisición de datos (DAQ) y microcontroladores, pero que debido al gran número de entradas y salidas que se necesitan en el proyecto, a la facilidad de programación y que es el tipo de dispositivo más frecuente en el control de procesos industriales, se eligió finalmente trabajar con PLC's.

Actualmente existe una fuerte demanda de estos dispositivos en tareas de automatización industrial y en consecuencia, un gran número de autómatas en el mercado de diferentes marcas, pero también diversos modelos dentro de una misma marca. En este caso se contaba en el Departamento con las siguientes marcas:

- Siemens: SIMATIC S7-1200.
- Schneider: Modicon M241 y TSX-Premium.
- Omron: CJ2M.

Todos poseen las suficientes especificaciones técnicas como para solventar el problema que se plantea, por lo que paso a ser pieza clave en su elección, el software que utiliza cada uno de ellos para la programación del proceso y que suele ser propio de la marca.

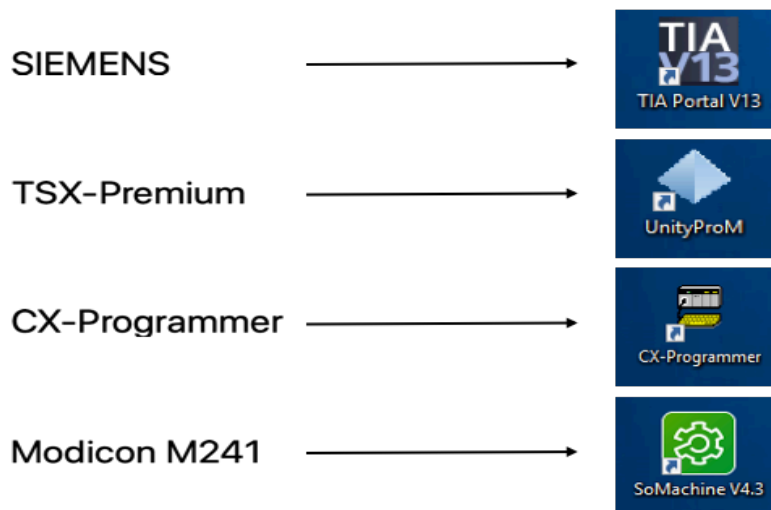


Figura 2. Marcas y modelos de PLC con sus respectivos softwares de programación.

Fuente: elaboración propia.

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

En asignaturas como Tecnología Automática y Laboratorio de Automatización y Control se aprendió a utilizar de manera básica Unity Pro con el autómatas TSX-Premium. Era interesante dominar con profundidad algún software de programación y explotar al máximo todas sus prestaciones. Por lo que se escogió este autómatas en la programación del Almacén vertical.



Figura 3. TSX-Premium de Schneider

En el proyecto, hay dos elementos que necesitan ser controlados por eso se decidió utilizar dos autómatas diferentes y aprender a comunicarlos. Esta situación es bastante común en empresas donde hay un PLC distinto en cada subproceso y puede que no sean de la misma marca, por tanto, es tarea del ingeniero saber comunicarlos.

En este punto es posible que se haya notado la complejidad que supone que cada marca posea un entorno de programación diferente con las limitaciones que esto crea. Esta fue la razón por la que se propuso el software de Somachine que emplea un entorno multiplataforma que recibe el nombre de CODESYS. Este entorno pretende simplificar la problemática anterior con un lenguaje unificado para la programación de PLC's de diferentes fabricantes y que mucha gente considera como el futuro. Marcas como Siemens y Omron siguen utilizando sus propios programas por lo que se decidió finalmente utilizar el M241 de Schneider que sí permite este tipo de entorno como segundo autómatas en la programación del Brazo robot. (8 Razones para aprender a programar CodeSYS, 2015)



Figura 4. Modicon M241 de Schneider

2.3 Lenguajes de programación

Según el estándar internacional, **IEC 61131-3** mencionado en el apartado 1.5. Normativa, quedan definidos cuales son los estándares de los siguientes 5 lenguajes de programación aceptados en la programación de autómatas. Los 5, se pueden agrupar en 2 grupos más amplios: Literales y Gráficos.

Lista y Textual	Lista de instrucciones(IL)	Lenguaje de tipo ensamblador, poco habitual en la automatización y muy rápido de ejecutar. Es utilizado sobretodo en la programación de microcontroladores. (Satoshi, 2017)
	Texto estructurado(ST)	Lenguaje de alto nivel empleado para realizar sentencias más complicadas como secuencias y condiciones de una manera veloz. Manejo de diferentes tipos de datos. (Satoshi, 2017)
Gráfico	Diagrama de contactos(LD)	También conocido como Ladder, está basado en diagramas de contactos eléctricos y es fácil de entender. Hoy en día es el más utilizado. Sentencias secuenciales sencillas. Difícil de utilizar en procesos más complicados. (Satoshi, 2017)
	Diagrama secuencial de funciones(SFC)	Representa de forma gráfica mediante la combinación de etapa-transición. Refleja secuencias de acciones propias de un automatismo. Tiene su origen en el lenguaje normalizado Grafcet. No es recomendado para programas de cálculo. (Satoshi, 2017)
	Diagrama de bloques de funciones(FBD)	Gran parecido al lenguaje de contactos, se lee de izquierda a derecha y está basado en bloques de funciones con entradas y salidas que pueden conectarse entre ellos. Es un lenguaje ideal para programas simples y para leer entradas de sensores, pero no lo es tanto para programas complejos. (Satoshi, 2017)

Tabla 1. Diferentes lenguajes de programación de autómatas y propiedades.

Fuente: Elaboración propia

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

Dentro de un mismo proyecto se pueden utilizar diferentes tipos de lenguaje y como se ha visto, cada lenguaje es idóneo en situaciones diferentes. A lo largo del trabajo se emplean 4 de los 5 lenguajes vistos:

- Diagrama Secuencial de Funciones (SFC): es el lenguaje predominante del proyecto, ya que definirá la secuencia de movimientos que realizará el automatismo del almacén y del brazo, además de para recoger los distintos estados y situaciones que vayan cambiando a lo largo del proceso y que son necesario conocer para el correcto funcionamiento.
- Lenguaje de contactos (LD): su misión es la de programar las distintas transiciones que sean algo complejas y que no dependan únicamente de una variable booleana, como podría ser, la activación de un sensor además de otras condiciones para el paso de variables.
- Lenguaje estructurado (ST): su empleo se ha reducido a una sola sección para la asignación de valores a determinadas direcciones de memoria.
- Lenguaje de bloques de funciones (FBD): se utilizó en funciones concretas de gestión de secciones SFC con el objetivo de implementar los modos de funcionamiento y paros de emergencia.

2.4 Comunicaciones

2.4.1 Comunicación entre Autómatas

En las empresas dentro de un proceso existen varios subprocesos que tienen características comunes, estos subprocesos es frecuente que se encuentren controlados por diferentes autómatas que tienen que coordinarse para que el proceso global se ejecute. Esta comunicación exige conocer la información actualizada de los demás autómatas como de sus sensores correspondientes. Esta comunicación se realiza normalmente a través de Ethernet y sus protocolos de red TCP/IP (Transport Control Protocol / Internet Protocol) mediante 3 servicios distintos:

- Peticiones de mensajería aperiódica.
- Exploración de E/S o Acceso a memoria compartida: mediante el establecimiento de roles Maestro/Esclavo o Cliente/Servidor entre 2 autómatas. El Maestro podrá leer y escribir en el esclavo en las direcciones de memoria compartida %MW (tipo entero de 16 bits). Existe el riesgo de sobrecargar la red si se intercambian muchos datos en tiempos de muestreo cortos.
- Datos globales: método potente con base en intercambios "multicast" que conecta tan sólo a aquellos equipos que quieren conocer información de esa dirección IP, por tanto, disminuye la carga en la red. (Fernandez)

Los 2 últimos servicios se aprendieron durante el seminario de Automatización de la asignatura de LAC y, dadas las características del proyecto, se eligió finalmente la Exploración de E/S, ya que se consideró la más adecuada y fácil de utilizar.

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

2.4.2 Comunicación con el control y supervisión del proceso

En la actualidad se emplea como estándar para estas comunicaciones, los servidores OPC que ponen fin a los problemas de drivers propietarios que dificultaba la interacción de datos con drivers distintos, asociados a distintos vendedores. Estos servidores utilizan una estructura Cliente-Servidor donde el servidor ofrece distintas direcciones de datos a las que el cliente es capaz de acceder para leerlos o escribir nuevos datos a través de la red Ethernet. (PDAControl, 2016)

La solución de estos servidores también se adopta en este trabajo y se utilizará el software KEPServerEX, desarrollado por Kepware, debido a que aparece instalado en los ordenadores del DISA y, además, se conocía su funcionamiento gracias a la optativa que se cursó. Pero, hay muchos otros como Wonderware y MatrikonOPC entre otros.



Figura 5. Logo de KEPServerEX

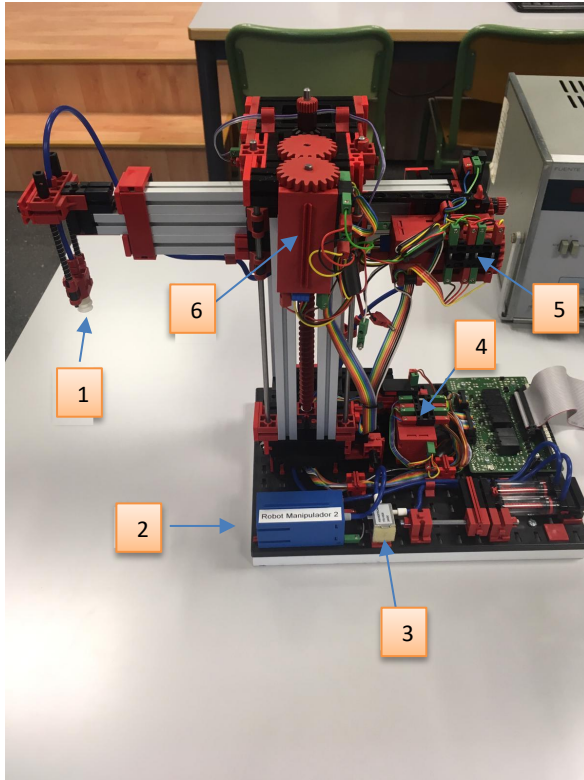
2.5 Control y supervisión

En esta tarea se utilizará las aplicaciones SCADA (Supervisory Control and Data Acquisition) tan popular en los últimos años y que permite controlar y conocer el estado del proceso, de una manera virtual y desde cualquier punto a través de una pantalla. En ella recibimos información de sensores y actuadores activos, además es capaz de suministrar datos respecto a la producción para su posterior gestión, control de calidad, mantenimiento e intervención para asegurar su correcto funcionamiento.

Para el desarrollo de un sistema SCADA existen multitud de softwares y dentro del DISA: NB-Designer para pantallas HMI Omron, CX-Supervisor, iFIX y LabVIEW. De estos 3 con características similares se elige el último. Desarrollado por National Instruments, LabVIEW, se caracteriza por un modo de programación totalmente gráfico que facilita la visualización y comprensión de la aplicación. Esta visualización hace que sea más fácil integrar hardware de medidas de cualquier proveedor, representar una lógica compleja en el diagrama, desarrollar algoritmos de análisis de datos y diseñar interfaces de usuario personalizadas (National Instruments, 2019). La posibilidad de programar en este software el programa encargado de la diferenciación del color de la pieza a partir de la imagen recibida por la cámara Kinect, supuso un motivo de peso en la elección.

CAPÍTULO 3: DESCRIPCIÓN DETALLADA DEL EQUIPO DE TRABAJO

3.1 Brazo ventosa



1	Ventosa
2	Compresor
3	Succión ventosa
4	Motor movimiento vertical + encoder vertical
5	Motor movimiento horizontal + encoder horizontal
6	Motor giro + encoder giro

Figura 6. Maqueta Brazo ventosa de FischerTechnik y sus elementos. Fuente: elaboración propia

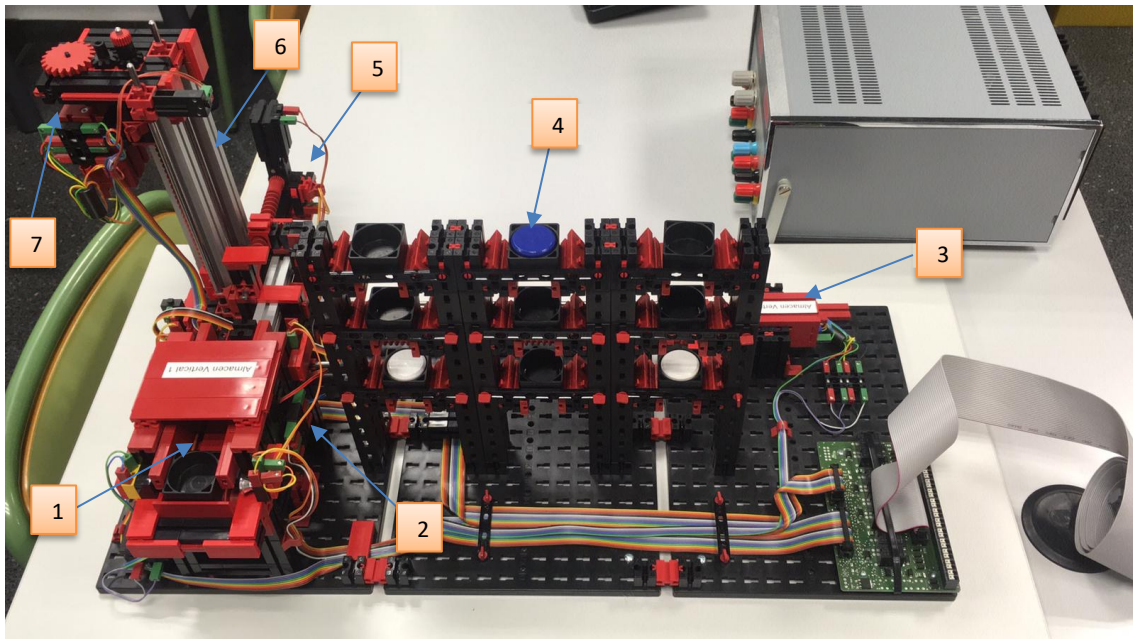
Se trata de un robot con movimiento en 3 ejes que cuenta con una ventosa para la succión de piezas de trabajo de manera precisa y veloz sobre espacios tridimensionales. Sus movimientos están comprendidos entre:

EJE X	EJE Y	EJE Z
270º de giro	Adelante/atrás 140 mm	Arriba/abajo 120 mm

Las dimensiones del brazo son de 22,2 x 48,2 x 38,2 cm. Pesa aproximadamente 3 kg y está alimentado a 24 V. Este equipo incluye:

- 3 motores de corriente continua, uno para cada eje, con encoder magnético que genera un pulso con cada vuelta del motor.
- 3 pulsadores (interruptores finales) para detectar los finales de carrera.
- Ventosa de vacío y compresor que ejecutan la succión de la pieza. (FischerTechnik, 2019)

3.2 Almacén vertical



1	Cinta con movimiento en 2 direcciones
2	Motor cinta
3	Motor movimiento horizontal
4	Celda con caja ocupada
5	Motor desplazamiento plataforma
6	Transelevador
7	Motor movimiento vertical

Figura 7. Maqueta Almacén vertical y sus elementos. Fuente: elaboración propia

Almacén robotizado que cuenta en su entrada con una cinta transportadora con 2 sentidos que permite guiar las piezas hacia el interior/exterior de este. El elemento que realiza el movimiento entre la entrada y las celdas es un transelevador con movimiento en los 3 ejes que transporta los portadores de piezas o cajas en cuyo interior viajan las piezas. Las piezas son cilíndricas de 3 centímetros de diámetro y de 3 colores distintos (rojo azul y blanco) como indica la Figura 8. Hay 9 celdas en total ordenadas en un matriz de 3x3.

Las dimensiones del almacén son de 47,2 x 72,2 x 38,2 cm. Pesa aproximadamente 8 kg y está alimentado a 24 V. Este equipo incluye:

- 3 motores de corriente continua, uno para cada eje. Los motores que permiten los movimientos del transelevador en horizontal y en vertical, llevan encoder.
- 1 mini motor de doble sentido de corriente continua en la cinta de entrada.
- 4 pulsadores (interruptores fin de carrera).
- 2 fototransistores y 2 barreras de luz LED que funcionan como dos sensores para detectar cajas en los dos extremos de la cinta de entrada. (FischerTechnik, 2019)

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.



Figura 8. Piezas de colores y caja.

3.3 Cámara Kinect



Figura 9. Cámara Kinect.

Este dispositivo de 24.9 x 6.6 x 6.7 cm, fue desarrollado por Microsoft para las videoconsolas Xbox en 2010 y aunque actualmente se ha retirado del mercado su tecnología se sigue usando con fines educativos. "La característica que lo hace diferente, es la capacidad de reconocer gestos, comandos de voz y objetos en imágenes (Hernández, 2016)". Esta cámara digital de reconocimiento presenta los siguientes elementos básicos:

- Cámara de vídeo de color RGB: como si se tratase de una webcam, captura las imágenes en vídeo a una velocidad máxima de 30 fps. Con esta información se obtienen detalles de objetos y personas en la sala. (BOXBYTE, 2010)
- Emisor de infrarrojos: mediante una cámara de profundidad, permite mapear en 3D la habitación con todos los objetos y personas que se encuentran en su interior.
- Micrófonos: Kinect incorpora cuatro micrófonos que calculan y reconocen la posición de la voz además de sistemas electrónicos que eliminan el ruido que pueda haber.
- Base inclinable automática que se ajusta a la posición de la persona mediante un motor.

"Pero lo que realmente dota a este dispositivo de inteligencia, es el software. Kinect es capaz de capturar una cantidad increíble de datos. Siempre fijando su objetivo en las cosas que se mueven en su entorno (Hernández, 2016)". En el Proyecto solo se utiliza la cámara de video de color RGB y la base inclinable para apuntar con precisión a la pieza.

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

Para realizar el reconocimiento de color sobre las piezas se ha elaborado un programa en LabVIEW donde sobre un área de la imagen capturada se descompone el espectro cromático en un vector de 10 casillas donde aparece en cada casilla que porcentaje en tanto por 1 del total representa esa franja cromática. Con los porcentajes que se obtengan en las casillas se asocia a una combinación cromática, es decir, a un color. Cuando se detectan colores como azul, rojo o blanco se activa una variable y en el programa de LabVIEW se iluminará un led del color correspondiente.

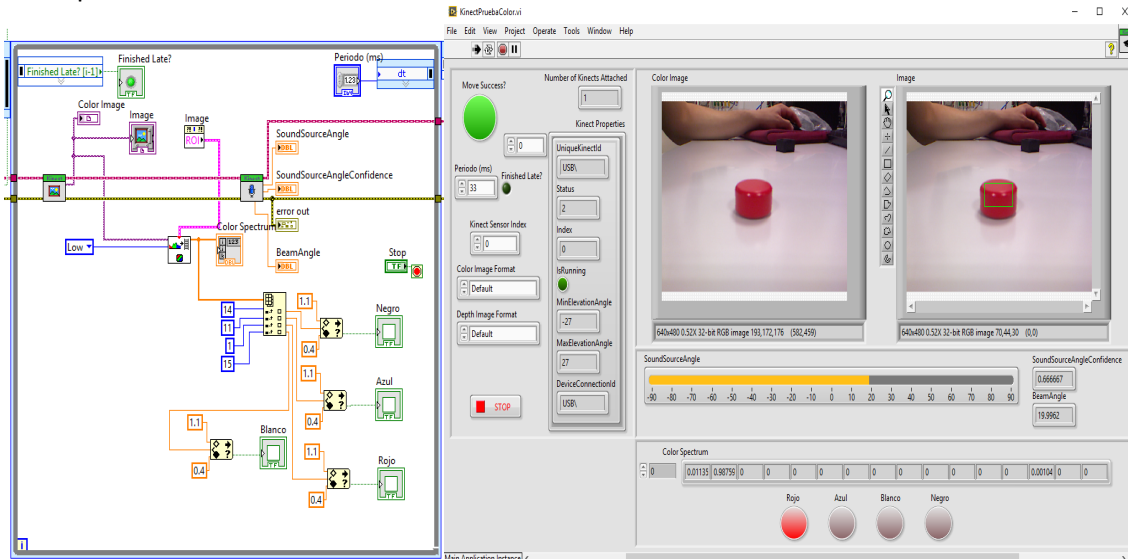


Figura 10. Programa de la cámara Kinect en LabVIEW. Fuente: elaboración propia.

3.4 Depósitos

Se utilizará un depósito que simula el lugar donde el operario debe dejar las piezas que se vayan a meter y donde se dirigirá el brazo para cogerlas. Para la salida de piezas ocurre lo mismo y se utilizará otro depósito diferente donde el brazo soltará las piezas que se vayan sacando.



Figura 11. Aspecto del depósito

CAPÍTULO 4: FUNCIONAMIENTO Y SOLUCION DETALLADA

Una vez conocidas las alternativas y que elementos son los que finalmente se utilizaron. Se dará paso a la explicación detallada de la solución indicando los pasos que se fueron siguiendo en cada una de las etapas y aquellos aspectos que presenten una mayor importancia. Todas aquellas Figuras que son pantallazos de programas son por tanto de elaboración propia, aunque no se indique en el pie de foto.

4.1 Diagrama de flujo de comunicaciones del proyecto

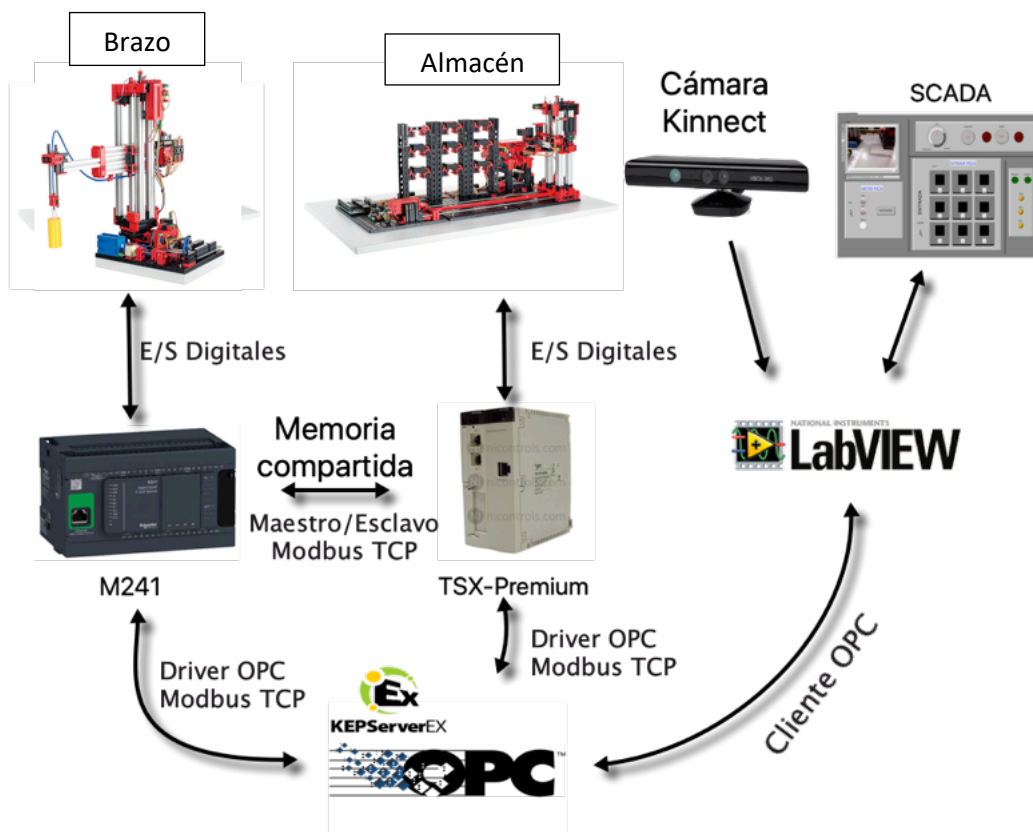


Figura 12 Esquema de las comunicaciones del proyecto. Fuente: elaboración propia

Teniendo en cuenta el principal objetivo del Proyecto que es automatizar un proceso de almacenamiento de piezas con almacén y brazo robot, se construye una solución cuyo flujo de información queda reflejado en el diagrama de la Figura 12. El almacén se gobierna con un autómatas TSX-Premium de Schneider programado en Unity Pro y el brazo con un M241 de la misma marca programado en Somachine. Los autómatas conectan las entradas y salidas digitales de las maquetas con un cable D-sub. Ambos PLC's requieren, para el desarrollo de la automatización, de información y decisiones que reciben de 2 maneras. La primera de ellas mediante la comunicación que realizan entre ellos a través del acceso a la memoria compartida con protocolo maestro/esclavo sobre ModBus TCP. La segunda, mediante un servidor OPC que accede a sus direcciones de memoria para escribir todo aquello que el usuario controla desde el SCADA junto con la información sobre el color de piezas recogida por la cámara Kinect que se ha programado en LabVIEW. Además, LabVIEW usa el servidor para poder leer aquella información de importancia que es mostrada por el SCADA para la monitorización del proceso.

4.2 Distribución y recorrido de las piezas

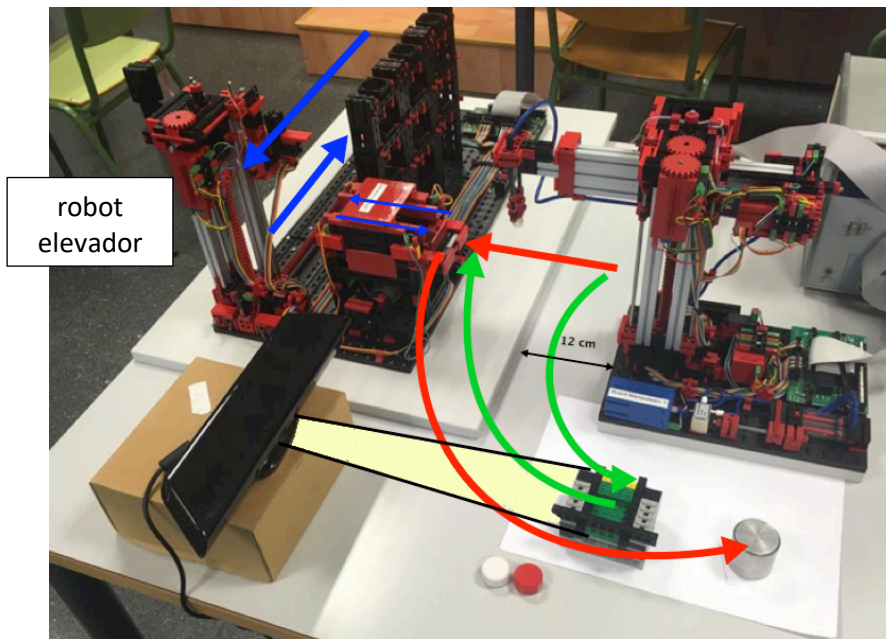


Figura 13. Recorrido de los robots. Fuente: elaboración propia

Como muestran la Figura 13, esta es la distribución que se siguió con las maquetas en la mesa de trabajo dentro de un aula del DISA y en ella aparecen marcadas las flechas con las direcciones de los movimientos. El almacén que es el primero en moverse ya sea para meter o sacar piezas, tiene que hacer siempre el mismo movimiento, ir hacia las celdas, coger la caja (llena o vacía) y llevarla hacia la cinta. La cinta lleva la caja hacia la entrada donde permanece en espera. El brazo, recibe la información de que la pieza está esperando y si tiene que coger una pieza porque lo que se está haciendo es meter, seguirá la flecha verde. Si, por el contrario, se está sacando una pieza, las flechas rojas son las que indican el recorrido.

Las medidas empleadas no muestran demasiada importancia ya que, durante el proceso, las maquetas se mueven debido a su poco peso. Las referencias que se seguían son: una distancia de 12 cm entre ambas maquetas y que el brazo coincidiera en su posición inicial con la entrada del almacén. Las referencias entre el brazo y los depósitos se apuntaron en un folio que se colocaba debajo de estos.

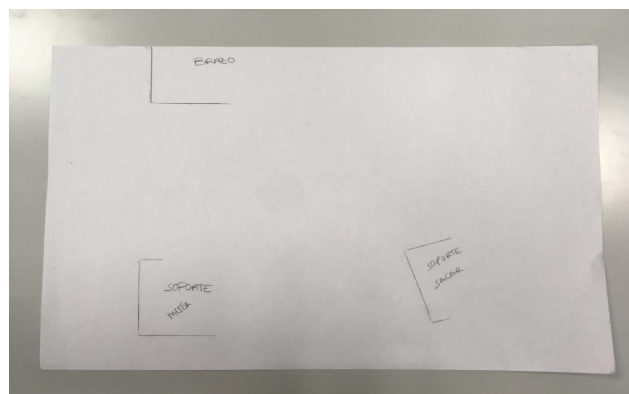


Figura 14 Distribución brazo y depósitos

4.3 Calibración de los equipos

Como se mencionaba en el Capítulo 3, ambas maquetas, presentan motores de corriente continua con un encoder que transmite pulsos eléctricos cada vez que el motor ha dado una vuelta. Esto nos permite conocer, mediante un sencillo programa, la posición en la que se encuentra una parte móvil de los robots respecto su posición de origen ya que, en variables creadas como "CuentaVertical" se les sumará uno a cada pulso del encoder. De esta manera, se tiene un número entre 0 y 1000 que representa la distancia entre estos dos puntos.

Este dato se coloca posteriormente en una transición comparando (como en la Figura 15) el número de pulsos que lleva un elemento móvil con el número que representa el lugar destino, si se supera el número, la transición se activa y el movimiento cesa dando paso a otro movimiento.

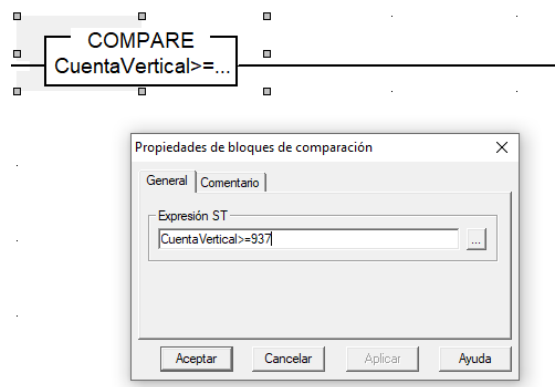


Figura 15. Transición basada en un comparador en Unity Pro

La calibración requiere la activación manual de las salidas, para ello se configuró una pantalla en Somachine y Unity Pro con las salidas y sensores del almacén y el brazo además de un visor del encoder tal y como se muestra en la Figura 16. A mano se apuntaron los números para las distintas posiciones de la secuencia y así posteriormente introducirlas en el proceso automático.

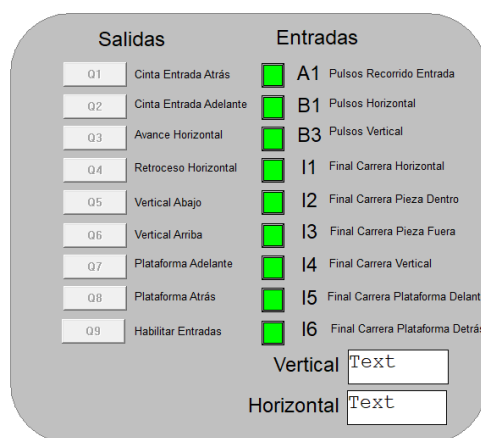


Figura 16. Pantalla control manual

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

Es necesario que la activación de una salida mientras se calibra sea seguida, es decir, sin parones, ya que los pulsos del encoder se producen a gran velocidad y si se para, se producen imprecisiones debido a la acumulación y solapamiento de estos.

4.3 Programación del Proceso

Una vez fueron conocidas las posiciones respecto al encoder que aseguraban que los movimientos de los robots fueran precisos se pasó al diseño de las secuencias del proceso. Para ello antes de programar nada, en una hoja, se realizaron los esquemas de estos siguiendo los modelos Grafcet aprendidos en la asignatura de Tecnología Automática y usando como guía las especificaciones que el cliente marcaba y que se han descrito en el apartado Especificaciones del Proceso.

El Grafcet es un modelo de representación de automatismos desarrollado en Francia y que describe gráficamente una secuencia. Se basa en etapas que se activan sucesivamente, cada etapa lleva asociada una acción o salida y el paso entre estados se produce cuando se cumple una transición que desactiva la anterior y activa la siguiente. El primer estado aparece doblemente recuadrado y se llama estado inicial. El lenguaje Grafcet es muy similar al lenguaje SFC que incorporan muchos programas. (GRAF CET, 2019)

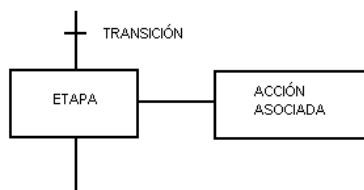


Figura 17 Bases del lenguaje Grafcet

Con un esquema claro de lo que se tenía que hacer, se programaron estas secuencias en las dos aplicaciones elegidas para ello.

4.3.1 Unity Pro - Almacén Vertical

Al inicio del programa aparece una barra a la izquierda que contiene la actividad MAST (Master o actividad principal) la cual se ejecutará en bucle y es en ella donde se incluyen todas las secciones de trabajo. Una sección de trabajo es un bloque de elementos comunes que se programan con un lenguaje de los vistos anteriormente. Después es posible añadir secciones dentro de una sección en un lenguaje diferente a esta. Esto se emplea mucho dentro de una sección SFC al incluir secciones para las transiciones complejas que no tengan una sola variable normalmente programadas en lenguaje LD.

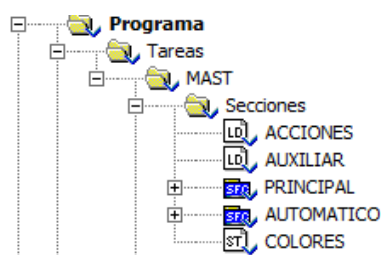


Figura 18. Secciones en Unity Pro

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

Los Graficets están incluidos en el Anexo Diseño del Automatismo y solo se mostrarán y se señalará la información relevante para que cuando se consulte dicho Anexo, el lector entienda mejor el contenido. En total se crearon 5 secciones principales que se describen a continuación:

- Principal (SFC): esta sección contiene los Graficet de todos los posibles estados del almacén (vacío, libre y lleno) de las celdas (libre y color de la pieza que la ocupa), información sobre la ordenación de piezas

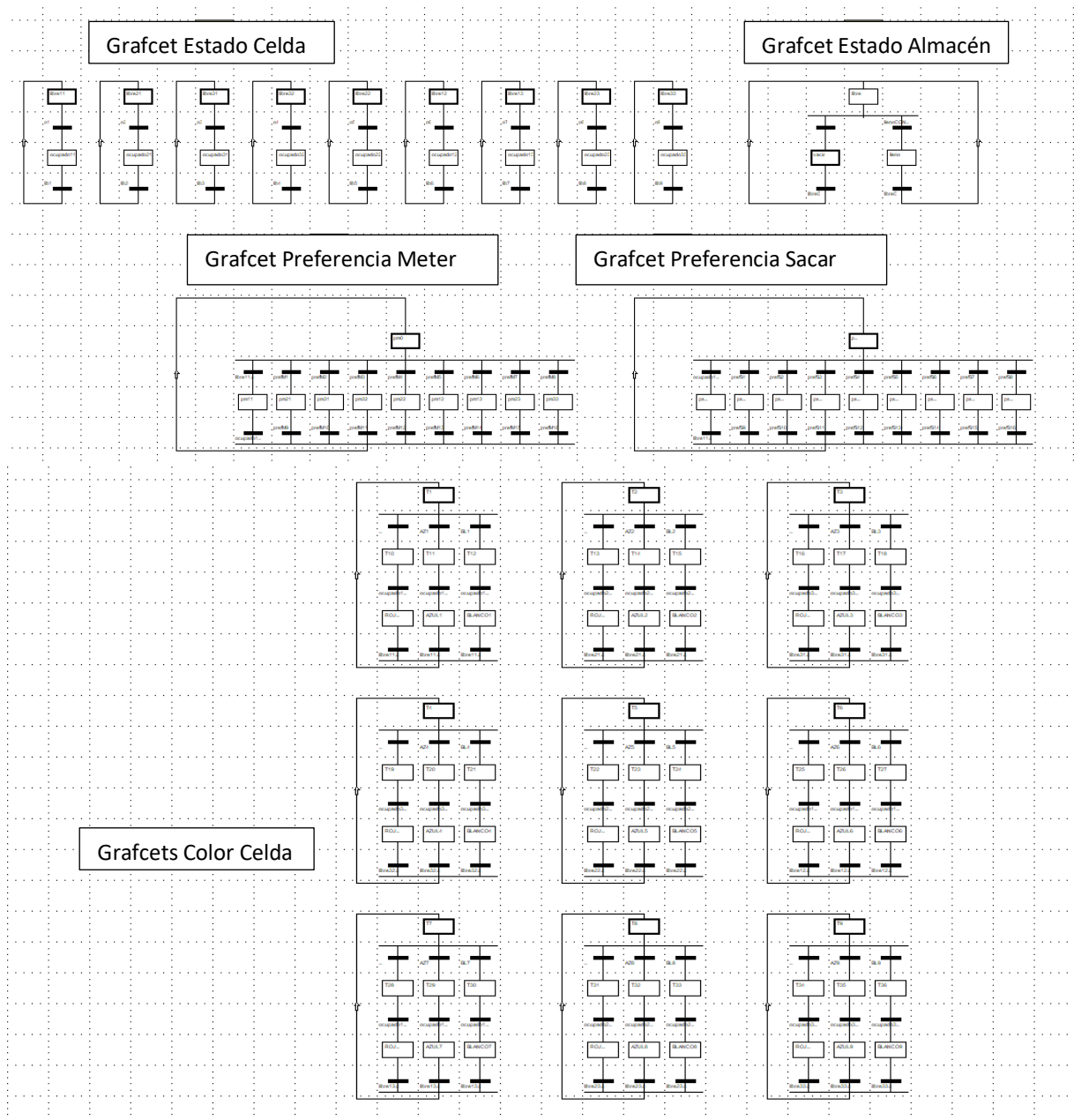
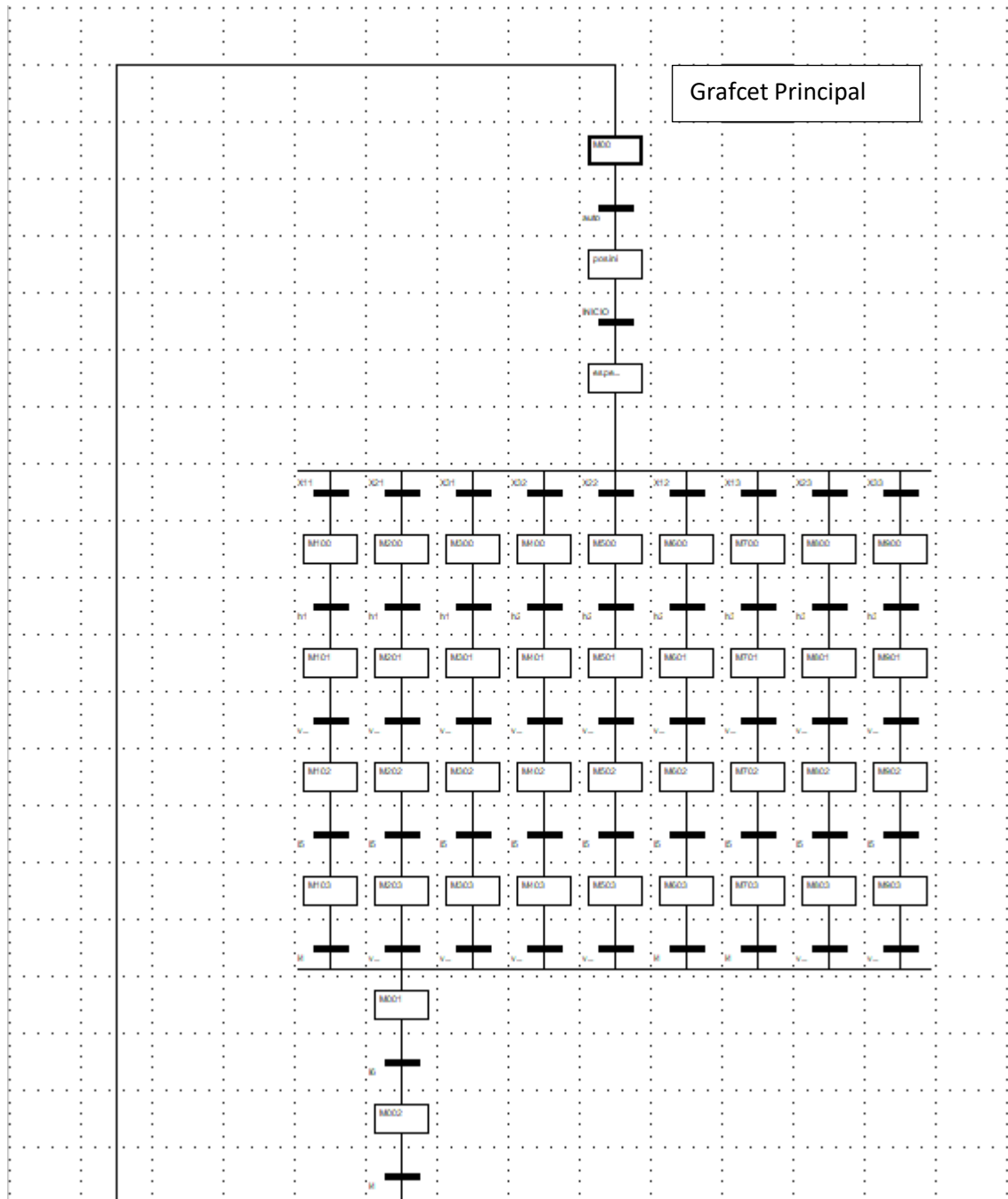
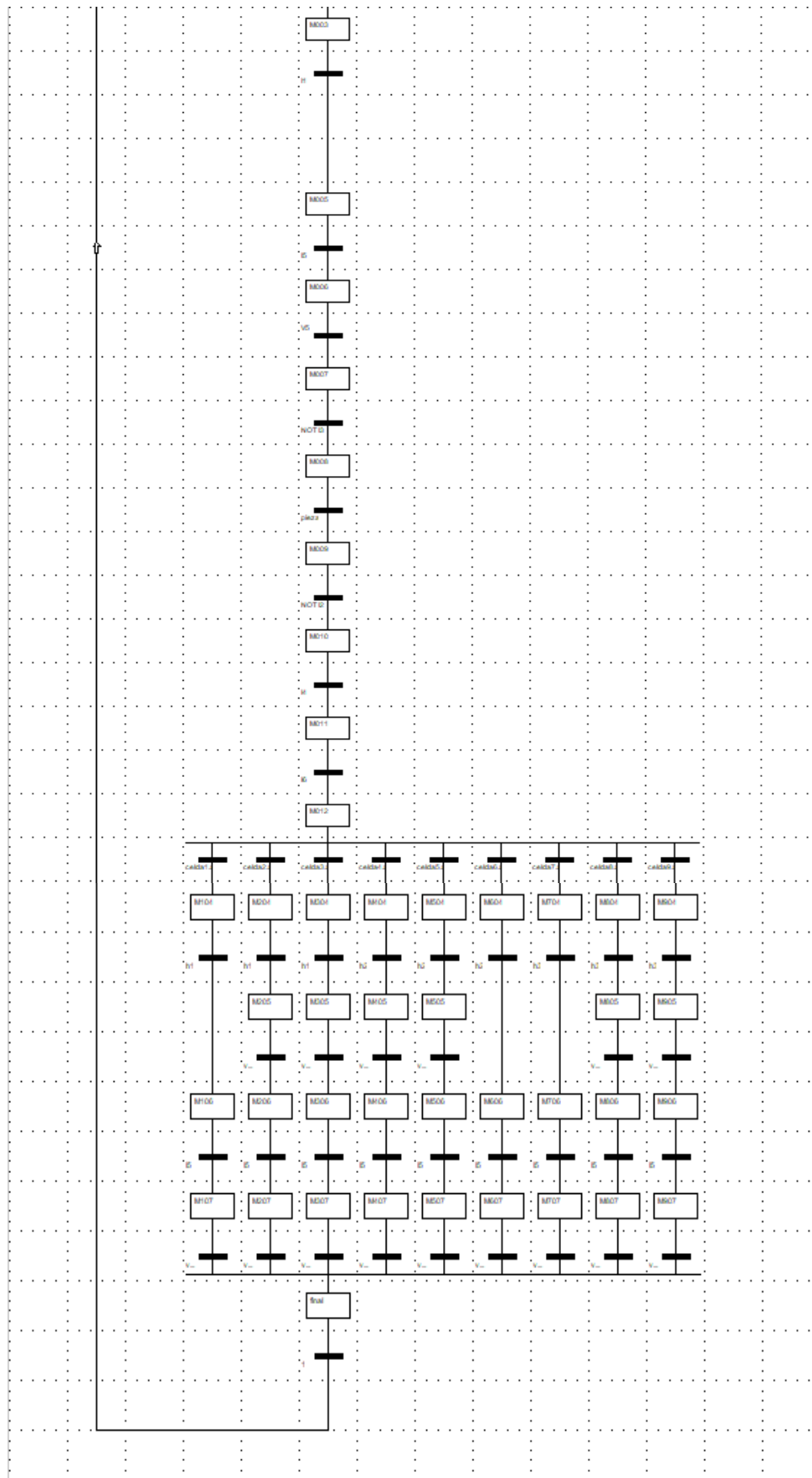


Figura 19. Sección Principal del Unity Pro

- Automático (SFC): contiene la secuencia del modo automático del almacén, así como, otros Grafquets auxiliares que proporcionan a la secuencia principal información necesaria como por ejemplo que celda está siendo intervenida o diferenciar si se está metiendo o sacando una pieza. Además, otros Grafquets parciales que pertenecen a la secuencia principal pero que su secuencia, más independiente, se puede hacer separado de este, es el ejemplo del regreso a la posición inicial.



Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.



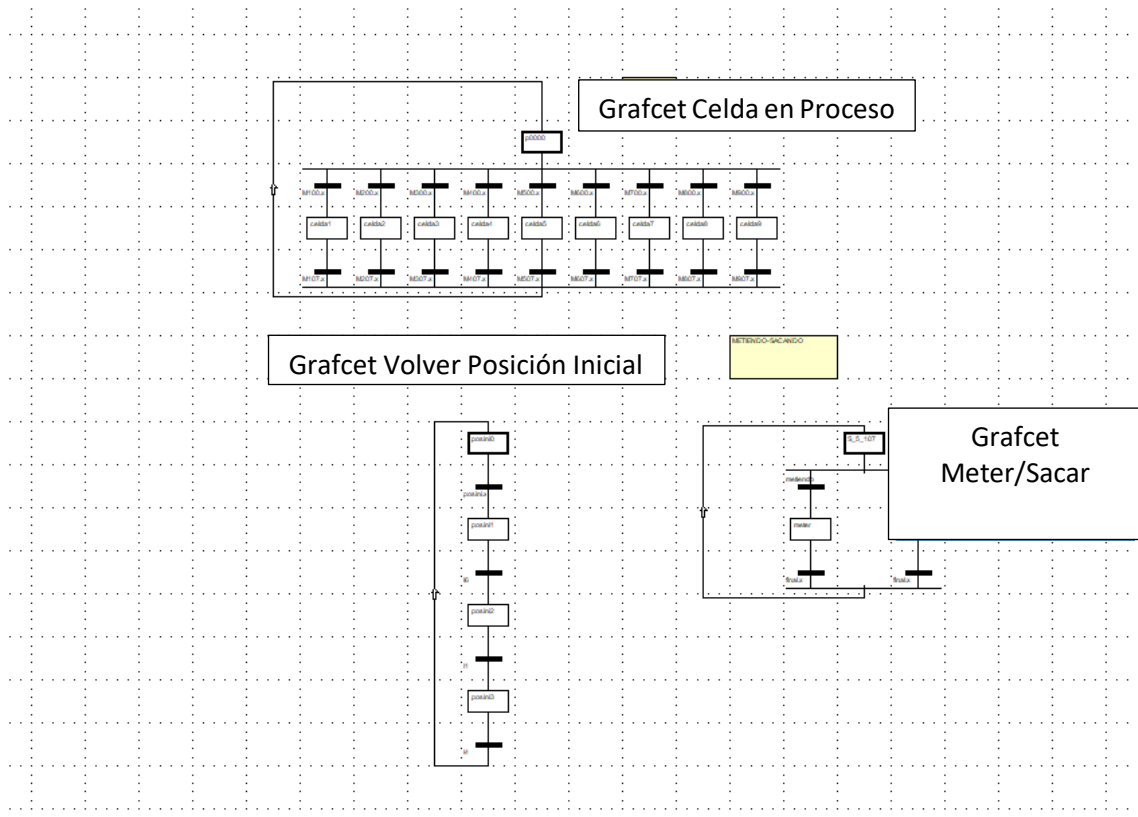


Figura 20. Sección automatismo de Unity Pro

- Acciones (LD): aquí se conecta cada salida de los robots con todos los posibles estados que podrían activarlas incluidas las variables de los botones del modo manual.

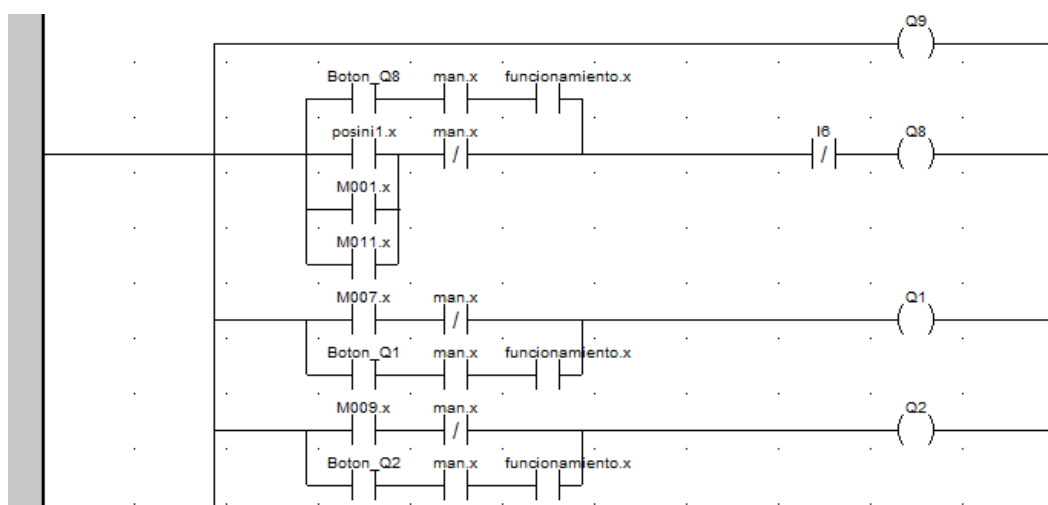


Figura 21. Ladder de Acciones es Unity Pro

- Auxiliar (LD): las transiciones complejas que dependen de más de una variable y no pueden incluirse como condiciones en las transiciones de las secciones SFC y que al repetirse varias veces durante el programa se programan por separado. También pertenecen la activación de otras variables de utilidad, la asignación de determinadas direcciones de memoria a algunas variables que necesitan ser compartidas, así como funciones especiales que se explicarán a continuación para configurar los distintos modos de funcionamiento.
- Colores (ST): en ella se actualiza el color de la pieza que ocupa una celda y se guardan esos datos en direcciones de memoria compartida %MW.

```
IF (ROJO1.X) THEN
%MW30:=1;
ELSIF (AZUL1.X) THEN
%MW30:=2;
ELSIF (BLANCO1.X) THEN
%MW30:=3;
ELSE
%MW30:=0;
```

Figura 22. Código en ST Unity Pro

Cabe destacar que las variables que se crean son globales y para acceder a ellas solo es necesario introducir su nombre. Otro punto fundamental a tratar es la implementación de la seguridad y de los tres modos de funcionamiento (automático, paro y manual). Para ello se investigó que funciones poseía Unity Pro que permitiesen controlar Graficets con la intervención de determinadas variables. Se encontró finalmente el bloque de la Figura 23 que reúne todas las funciones útiles para la gestión de SFC's.

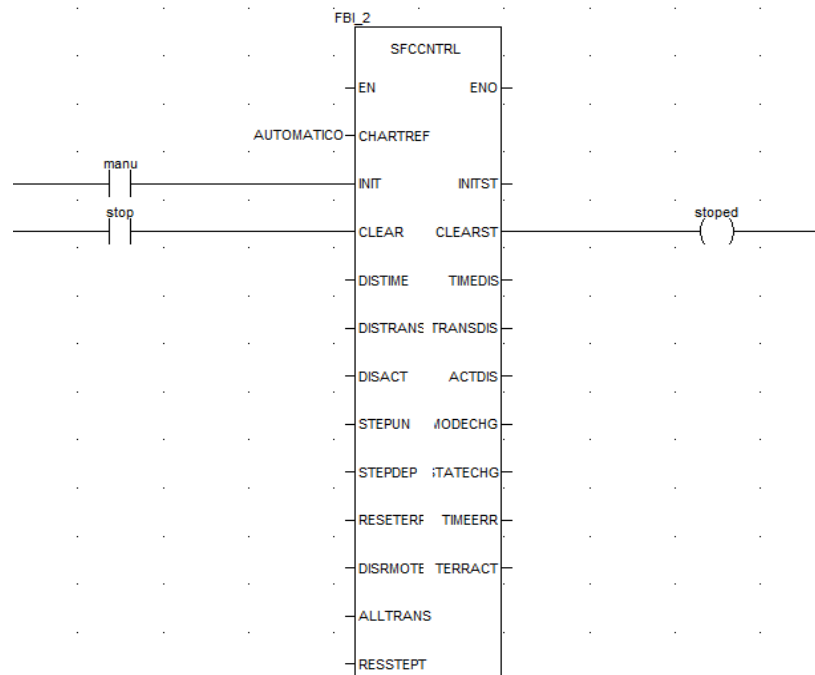


Figura 23. Bloque de función para las emergencias

Este bloque llamado SFCNTRL que se programó en la sección Auxiliar, agrupa varias funciones de control de secciones SFC, pero solo se utilizaron 2 de ellas. La sección que se controlará es Automático y aparece conectada con CHARTREF. CLEAR paraliza todos los Grafquets de la sección controlada y quedan detenidos hasta que se desactive. Para su reanudación es necesario que se active INIT, en este caso con manual. INIT reinicia al paso inicial cuando pasa de 1 a 0 (flanco de bajada) y siempre que CLEAR esté desactivado. En ese momento, los Grafquets comienzan de nuevo. Todo esto se traduce en que cuando se pulsa Stop durante una emergencia, se detenga todo el Automatismo y una vez desactivado sea necesario pasar por el modo manual. En el Anexo Manual de Usuario se especificará que debe hacerse.

El paro se consigue sencillamente con una transición al inicio del Grafquet del modo automático de que solo comience si está activado el modo automático. Por tanto, si la ruleta se encuentra en paro al acabar todo el proceso quedará en reposo en el estado inicial ya que al inicio y final de la secuencia se lleva el elevador a la posición inicial.

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

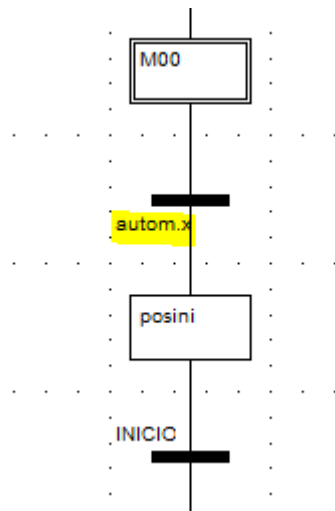


Figura 24. Transición para inicio del modo automático

4.3.2. Somachine - Brazo Ventosa

Interfaz similar a la de Unity Pro. La parte de programación se realiza desde la pestaña de Aplicación y en ella también nos aparece la tarea MAST con la diferencia de que aquí las secciones reciben el nombre de POU y si queremos que se ejecuten habrá que arrastrarlas hasta la carpeta MAST como indica la Figura 25 cosa que en Unity ocurría de manera automática.

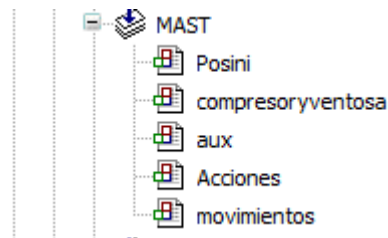


Figura 25 MAST de la aplicación en Somachine

Otra diferencia es que un POU en lenguaje SFC incluye solo un Grafcet y aunque la manera de crearlo sea mucho más intuitiva se considera como punto negativo respecto al Unity donde se pueden crear varios dentro de una sección. Siguiendo con las diferencias, las variables que se creen desde un POU se guardarán automáticamente como variables locales. Esto quiere decir que podemos llamar a 2 variables de diferentes secciones con el mismo nombre sin que nos aparezca un error. Para distinguir una de otra, fuera de su POU será necesario referirla de la siguiente manera "NombreSección.NombreVariable" tal y como muestra la Figura 26.

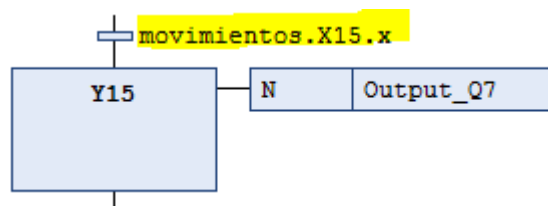


Figura 26. Tratamiento de las variables locales en Somachine

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

La programación del brazo ha sido más corta respecto al almacén y similar en su estructura, su contenido se incluye en el anexo y de él se destaca: el Grafcet principal con las dos posibles intervenciones del brazo (meter/sacar), otro para el retorno a la posición inicial llamado al inicio y final de la secuencia principal. Vuelve a ser de gran importancia la implementación de los diferentes modos de marcha y parada emergencia que en este caso presentaba características propias.

Como se comentó en las 1.4 Especificaciones del cliente existe una problemática en la parada de emergencia del brazo ya que normalmente una parada bloquea la alimentación de la máquina, deteniéndose por completo. Sin embargo, en el brazo, ese detenimiento no es total en caso de que esté trasladando una pieza, sino que los elementos encargados de la succión (compresor y ventosa) permanecen activos. La idea que se adopta para solventarlo es que el Grafcet Principal a partir de que el brazo coja la pieza, se divide en 2 Grafcets idénticos con la diferencia que uno se encarga de la parte de los movimientos y el otro de la succión. Se encuentran sincronizados entre los 2 y ante una emergencia se detendría solo el Grafcet de movimientos que volvería al paso inicial. Como se ha dicho, una vez que se desactivase la seta de emergencia, el brazo volvería a coger/sacar mientras mantiene la compresión y succión activada porque, las transiciones de uno, están relacionadas con el avance del otro.

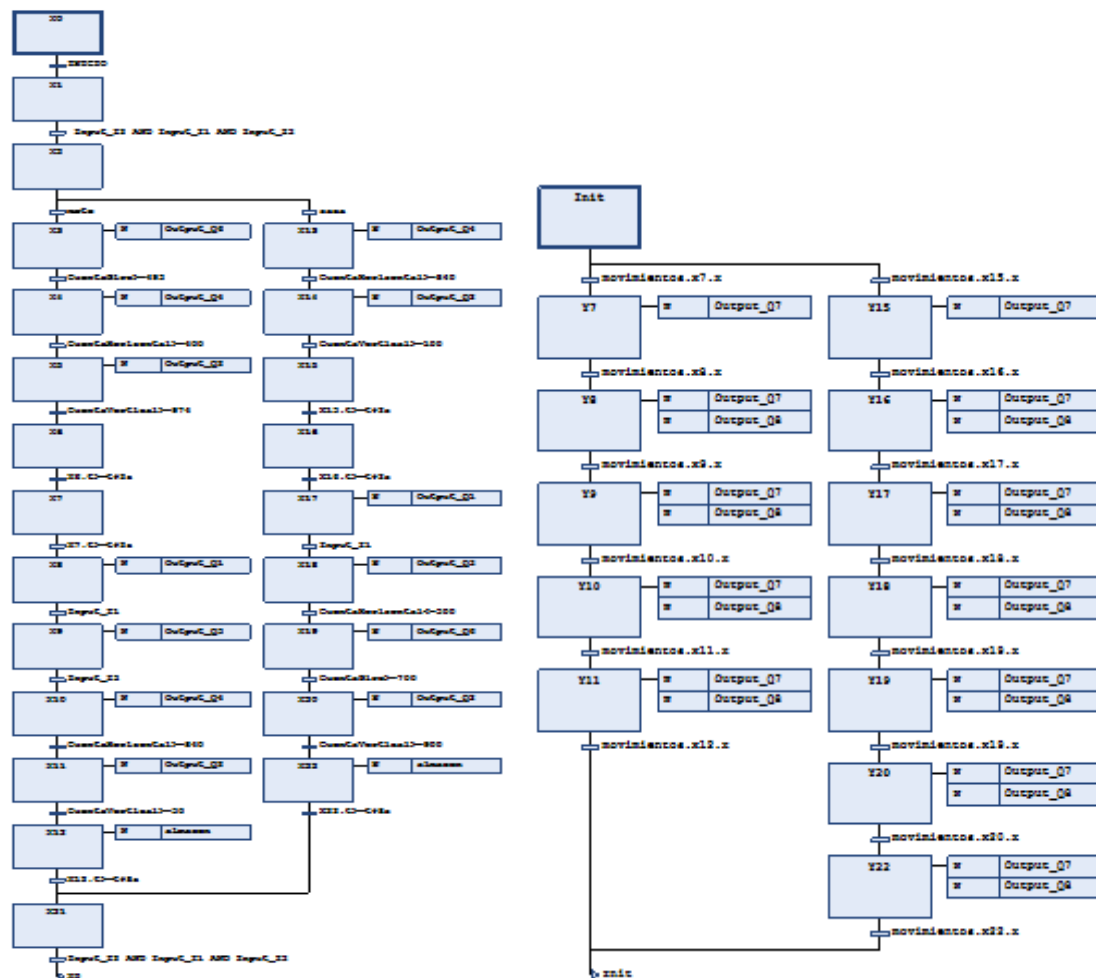


Figura 27 Grafcets de movimientos(izquierda) y compresor y ventosa(derecha)

Cuando la secuencia reiniciada vuelva al punto donde se detuvo, ambos Graficets volverán a avanzar de manera sincronizada. Durante todo el tiempo que permanezca la seta pulsada, un led rojo quedará iluminado indicando valga la redundancia que se trata de una emergencia. De igual manera, el modo manual también detendría el proceso de la misma forma en que lo hace la emergencia, con la incorporación en el panel de control manual del SCADA de un botón de "soltar" que libera la pieza si, finalmente, no queremos que vuelva a ser llevada hacia donde se dirigía. Esto es especialmente útil en casos donde se advierte que la pieza está en mal estado y se quiere devolver. El control manual solo será posible si no está activada la seta de emergencia. En el Anexo Manual de Usuario se proporcionará más información sobre el manejo de estas herramientas de una manera segura y eficaz.

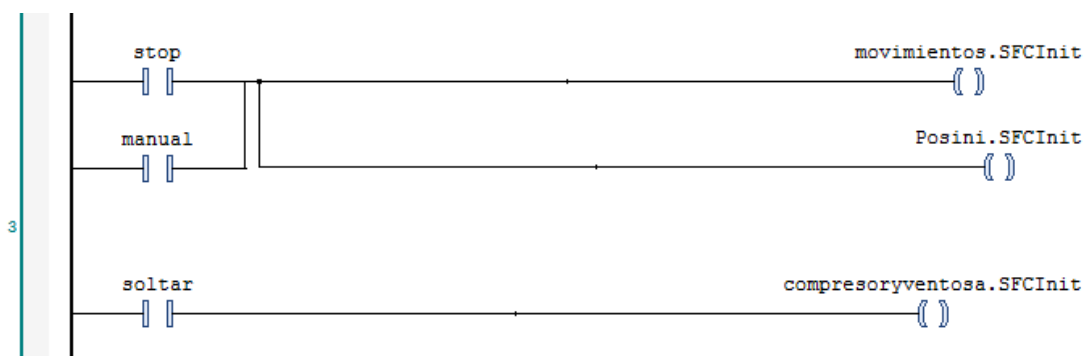


Figura 28 Implementación de paro y reinicio de un Graficet

4.4 Comunicación entre autómatas

Como se mencionó en el Capítulo 2, para dicha comunicación se empleó el servicio de memoria compartida a través de la red ethernet mediante el protocolo maestro/esclavo sobre ModBus. Esta comunicación afecta al PLC TSX Premium y al M241. La configuración se realizará a través del software Unity Pro que se recuerda que es el que se encargará del almacén vertical a través del TSX. Se trata de un procedimiento sencillo que se describe a continuación.

Dentro de la barra principal se busca la opción que pone Ethernet 1. Se abrirá una pestaña con varias opciones y se deberá pulsar la opción de Exploración de E/S. Si aparece bloqueada es debido a que la casilla de Exploración de E/S dentro del cuadro "Utilidades del Módulo" está puesto como NO. Se deberá cambiar a Sí. Una vez se accede a dicha opción, aparece un panel como el de la Figura 29. Ahora mismo, la IP con la que se conecte nuestro programa en este caso será la IP de un autómata TSX Premium, tomará el rol de maestro. Desde ese panel configuraremos la IP del autómata que realiza el rol de esclavo y por tanto del que podremos tanto leer como escribir en sus direcciones de memoria compartida. En el proyecto se emplearon las 2 posibilidades y se resume de manera gráfica en la Figura 30.

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.



Figura 29 Configuración Memoria Compartida



Figura 30 Esquema Memoria compartida. Fuente: elaboración propia.

Desde Somachine, es decir desde la aplicación del esclavo, solo se debería asegurar que la casilla de Servidor de Modbus apareciera activa.

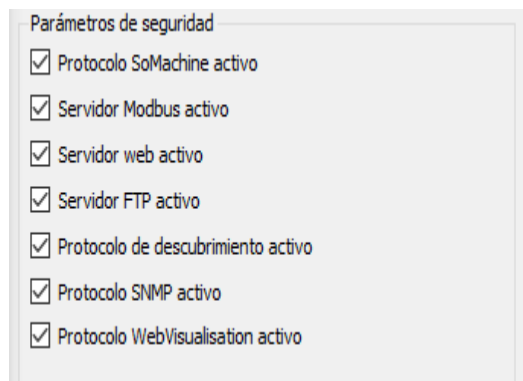


Figura 31 Activación Servidor

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

4.5 Servidor OPC

La creación de este servidor es el primero paso para desarrollar posteriormente el sistema SCADA y poder así escribir y leer en la memoria de los PLC's. El programa que se utiliza es KEPServerEX. En la Figura 32 se detallan los 3 pasos que se tienen que seguir en su creación y posteriormente, se indicarán que opciones se marcaron para este proceso.

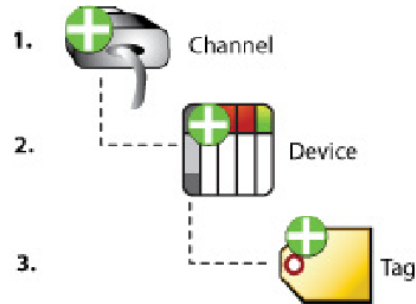


Figura 32 Elementos a configurar KEPServerEX

En el primer paso se creó un canal que recibe el nombre de TFG y agrupa todos los dispositivos que se comunican con el servidor cuando se active. Se marcarán todas las opciones por defecto excepto el driver que se emplea que en este caso es Modbus TCP/IP Ethernet.

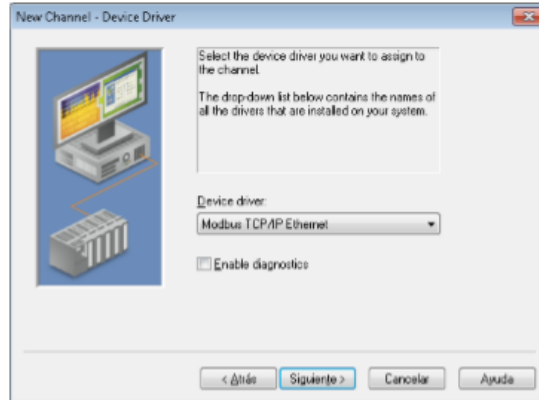


Figura 33 . Configuración del Canal en KEPServerEX

Para este proyecto se agregan 2 dispositivos almacén y brazo. Ambos utilizan un modelo Applicom ya que el direccionamiento de la memoria es similar al proporcionado en la programación del autómatas (el de Modbus también valdría) y un submodelo TSX premium. Después se escribe la IP del PLC que queremos conectar en cada caso.

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

Por último, se agregarán aquellas posiciones de memoria con las que se interactúa. En el caso de Applicom se ponen las direcciones de manera directa con el formato %MW0000X para tipos enteros con dimensión de 2 bytes y %MW0000X.Y para aquellas direcciones a nivel de bit.

Tag Name	Address	Data Type	Scan R...	Scaling	De
almBoton	%MW00002	Word	200	None	
almCB1	%MW00030	Word	200	None	
almCB2	%MW00031	Word	200	None	
almCB3	%MW00032	Word	200	None	
almCB4	%MW00033	Word	200	None	
almCB5	%MW00034	Word	200	None	
almCB6	%MW00035	Word	200	None	
almCB7	%MW00036	Word	200	None	
almCB8	%MW00037	Word	200	None	
almCB9	%MW00038	Word	200	None	
almColor	%MW00001	Word	200	None	
almInd	%MW00021	Word	200	None	
almMANUAL	%MW00005	Word	200	None	
almModo	%MW00003	Word	200	None	
almStopp	%MW00004.0	Boolean	200	None	

Figura 34. Variables y sus direcciones en el almacén.

Tag Name	Address	Data Type	Scan R...	Scaling	De
brazoMANUAL	%MW00005	Word	100	None	
brazoModo	%MW00003	Word	100	None	
brazoStopp	%MW00004.0	Boolean	100	None	

Figura 35. Variables y sus direcciones en el brazo.

El Scan Rate por defecto es de 0.1 segundos, pero se observó que en las variables que compartía el almacén, existía un menor solapamiento si se aumentaba ese tiempo de muestreo en una décima de segundo más.

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

4.6 SCADA

El primer paso es crear un proyecto en LabVIEW al que en este caso se ha llamado "proyectoTFG". El proyecto contiene 2 librerías: una almacena el servidor OPC y la otra las posiciones de memoria que en este se comparten, y un archivo .vi que contiene el Scada.

Una vez almacenado el servidor en la librería, LabVIEW, es capaz de usar y registrar las variables y sus direcciones de memoria contenidas en el servidor que hemos incluido. Esto se realizó en otra librería diferente que se llamó "varCompartidas".

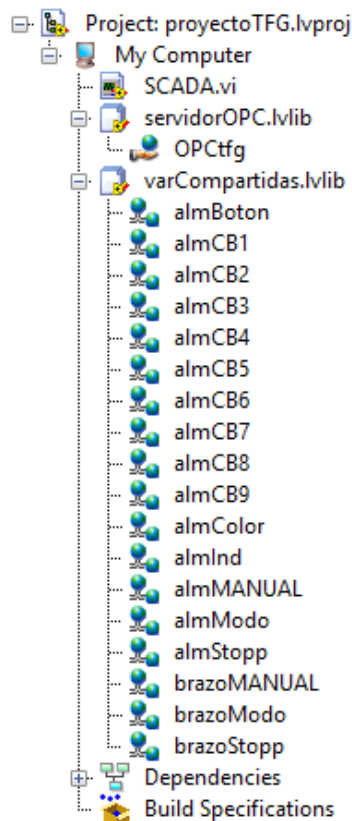


Figura 36. Contenido del Proyecto en LabVIEW

El paso siguiente es crear un archivo .vi que es el formato que utiliza LabVIEW para los SCADAS. Un archivo .vi consta de 2 partes: un panel frontal que es la interfaz que ve el usuario y contiene los elementos visuales y gráficos del SCADA y un diagrama de bloques donde se programan esos elementos visuales. Se utilizó para la programación el mismo archivo .vi donde se programó con anterioridad la cámara Kinect. Cualquier elemento que se añada en una pantalla, aparecerá en la otra, con la forma correspondiente. Normalmente se distingue entre controladores que nos permiten escribir en las direcciones de memoria y los indicadores que leen el estado de las variables, todo ello de forma gráfica. LabVIEW utiliza el color azul para objetos enteros (ruleta o caja de color) y el verde para booleanos (botones, interruptores, leds). Con estos conceptos básicos indicados, se pasa a la explicación de las dos pantallas que conforman el SCADA.

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

- Diagrama de bloques

Para añadir una variable contenida en el OPC a la pantalla de programación se debe arrastrar la variable deseada desde la librería "varCompartida". Una variable, se puede utilizar para leer o escribir y esto se configura desde el botón derecho en la opción Acces Mode.



Figura 37. Variable entera en modo escritura

En el diagrama todas las acciones que se quiere que se repitan cada periodo de tiempo hay que arrastrarlas al bucle principal que aparece con una forma cuadrada. En ella, aparece un elemento donde se puede configurar el tiempo de muestreo en este caso se eligió 100 ms para que las reacciones del proceso sucedieran con la mayor inmediatez y coherencia con otros tiempos de muestreo ya configurados.

En la Figura 38, se observa cómo se programaron los controles del movimiento manual del almacén. El control se realiza mediante interruptores, estos, son objetos booleanos que se agrupan en una variable de tipo entera. Para ello se crea un vector con la opción de build array. Las casillas comienzan en la posición 0 (arriba del todo) y en este caso se crean tantas casillas como controles se tenga. Cuando el interruptor se active, un 1 ocupara la posición correspondiente, cuando no, un 0. Mediante un conversor "array to num" se pasa el vector que representa un numero en binario con su LSB en la casilla 0 a un número entero que es el que se pasa a la dirección de memoria. Luego desde el Unity Pro se accede a esas direcciones bit a bit para extraer la información que se requiera.

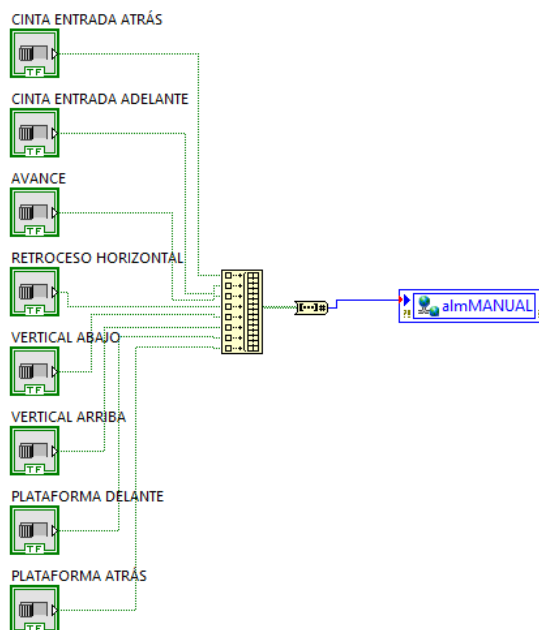


Figura 38. Programación controles en LabVIEW

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

Semejante es el proceso para los indicadores que leen el contenido de la memoria. El proceso se repite, pero a la inversa. Aquí es necesario indicar el número del bit que se quiere leer.

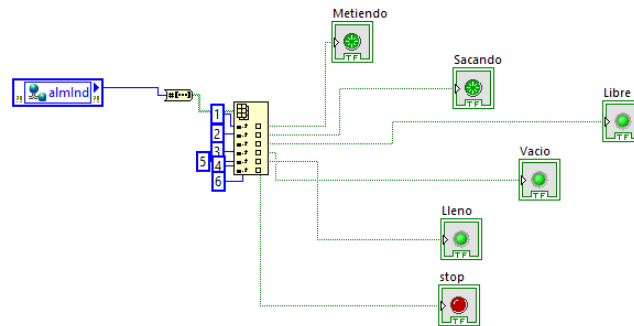
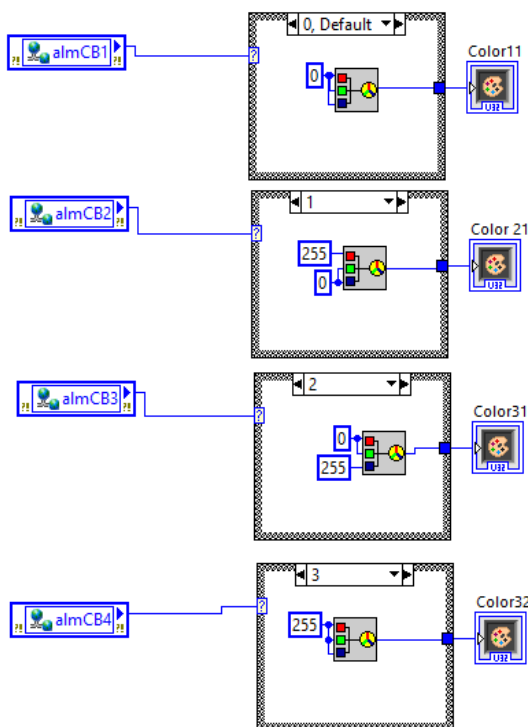


Figura 39. Programación Indicadores LabVIEW.

Un caso especial son las luces que tienen que mostrar el color de la pieza que ocupa el almacén, al ser 4 opciones (ninguno, blanco, azul o rojo) se trata de un indicador entero llamado Color Box. El valor de la memoria se indica en la Tabla 2. La dirección se conecta a una estructura "case" donde podemos colocar una sentencia en su interior diferente para cada valor. La función en este caso es la misma y es "RGB to color". RGB son las iniciales en inglés de los 3 colores primarios a partir de los cuales, variando su intensidad, se puede crear cualquier color. La intensidad es un número de 8 bits es decir del 0 al 255 que se colocaban con una constante. En la Figura 40 se muestran los 4 casos posibles.



Case 0	Vacío (negro)
Case 1	rojo
Case 2	Azul
Case 3	Blanco

Figura 40. Estructuras "case" y función "RGB to color"

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

- Panel Frontal

Todos los elementos se incluyen dentro de una estructura que permite crear varias pestañas para mayor claridad de la interfaz. Se utilizó un Tab Control con 2 pestañas, una, para el control del proceso automático, los modos de funcionamiento y las emergencias y otra, para el control del modo manual. Si se quiere que al iniciar el SCADA, aparezcan unos valores predeterminados como por ejemplo en la ruleta de modos el estado de paro es necesario que, al pinchar con el botón derecho sobre este elemento, se seleccione la opción "make values default".

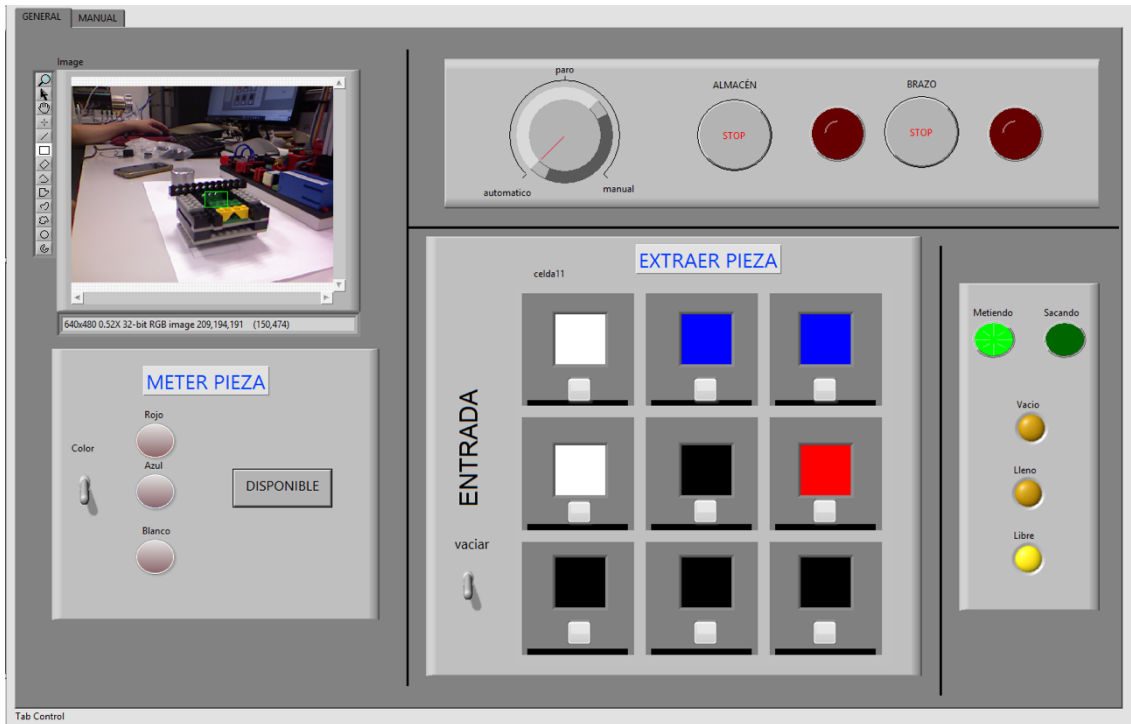


Figura 41. Aspecto del panel SCADA

CONCLUSIONES

Con la finalización de este proyecto de automatización se pasa a valorar si se alcanzaron aquellos objetivos planteados en un inicio, y que quedan recogidos al principio del documento, y plantear las conclusiones que se han podido extraer durante la elaboración del mismo.

Primero de todo se considera que se cumple el cometido principal de realizar un sistema de almacenamiento de piezas y que además mediante un sistema SCADA, el usuario es capaz de controlar. Esto se comprueba con la puesta en marcha del sistema mediante los simuladores obteniendo los resultados esperados. Además, se añadieron funciones específicas cuyo objetivo era mejorar y enriquecer el proyecto de forma notable. A continuación, se valorará cada una de estas:

- Distintos lenguajes de programación: se intentó utilizar el máximo número de lenguajes, 4 en este caso, y relacionarlos entre ellos ya que cada uno muestra sus propias ventajas y dotan al programador de mayor versatilidad.
- Cámara de reconocimiento: no se quiso perder la oportunidad de mejorar el sistema añadiendo un detector de color que se tuvo que integrar en el sistema. Terminó por aportar realismo al SCADA y al sistema en general, ya que la mayoría de almacenes son capaces de reconocer el tipo de pieza que están almacenando.
- Distintos modos de ordenación: sin duda la parte más larga y tediosa algunas veces, ya que se tuvo que tener muchas variables y situaciones en cuenta. Sin embargo, sirvieron para mejorar el pensamiento lógico y racional.
- Manejo y comunicación de varios PLC's: con el tutor se coincidió en que este punto sería extremadamente útil y una vez terminado creo que así ha sido. Aunque de la misma marca, presentaban claras diferencias los dos autómatas, pero su comunicación se pudo realizar de manera sencilla y eficaz. Dota al estudiante de una mayor confianza en caso de tener que enfrentarse a un nuevo dispositivo en el futuro.
- Softwares de programación: En el mercado existen casi tantos softwares como marcas de PLC (y no son pocas) por tanto, se consideró necesario aprender distintos programas entre ellos uno que empleara un entorno CODESYS ya que, aunque es demasiado pronto puede que en el futuro adquieran una gran importancia. Se consiguió dominar el Unity Pro y Somachine con bastante soltura, aunque al principio puede parecer que son complicados y que es mejor siempre el que ya se sabe utilizar.
- Servidor OPC: Este sistema de distribución de información a la red resulto sorprendente en cuanto a su sencillez y versatilidad ya que continuamente se estaban añadiendo direcciones nuevas sobre todo de la parte del SCADA de manera muy eficiente lo que permitió una monitorización más realista y completa.

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

La parte de redacción que incluye el documento ha sido más larga de lo que se esperaba tal vez porque durante la carrera no se realizaron demasiados trabajos y no se estaba acostumbrado. Además, la necesidad de que lo escrito sea fácilmente comprensible e ilustrativo incrementaba los tiempos de trabajo que muchas veces no se calculaban bien. De todas formas, conforme el documento cogía forma, la velocidad y agilidad fue mayor y se considera que han proporcionado al alumno mayor destreza y confianza en la realización de este tipo de escritos.

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

BIBLIOGRAFÍA

- (s.f.). Obtenido de National Instruments: <https://www.ni.com/es-es/shop/labview.html>
- 8 Razones para aprender a programar CodeSYS. (27 de Noviembre de 2015). Obtenido de infoPLC: <https://www.infopl.net/blogs-automatizacion/item/103031-razones-aprender-codesys>
- BOXBYTE. (30 de Junio de 2010). *Especificaciones técnicas de Kinect*. Obtenido de FayerWayer: <https://www.fayerwayer.com/2010/06/especificaciones-tecnicas-de-kinect/>
- Carlotti, F. (26 de 03 de 2019). *La Industria 4.0 abre la automatización a los almacenes pequeños*. Obtenido de <http://www.interempresas.net/Logistica/Articulos/240906-La-Industria-40-abre-la-automatizacion-a-los-almacenes-pequenos.html>
- Fernandez, R. S. (s.f.). Seminario Automatización de Procesos. Valencia, España.
- FischerTechnik. (2019). *Almacén elevado automatizado 24V - Education*. Obtenido de fischertechnik: <https://www.fischertechnik.de/es-es/productos/simular/modelos-de-entrenamientos/536631-sim-almacen-elevado-automatizado-24v-education>
- FischerTechnik. (2019). *Manipulador de aspiración al vacío 24V - Education*. Obtenido de fischertechnik: <https://www.fischertechnik.de/es-es/productos/simular/modelos-de-entrenamientos/536630-sim-manipulador-de-aspiracion-al-vacio-24v-education>
- GRAF CET. (s.f.). Obtenido de Wikipedia: https://es.wikipedia.org/wiki/GRAF_CET
- GRAF CET. (29 de Julio de 2019). Obtenido de Wikipedia: https://es.wikipedia.org/wiki/GRAF_CET
- Gutierrez, B. (Junio de 2019). *SEGURIDAD EN MAQUINARIA EN ISO 13849-1*. Obtenido de infoPLC: http://www.infopl.net/files/documentacion/seguridad_normativa/infoPLC_net_Seguridad_Maquina_SEiS_Maquinaria.pdf
- Hernández, L. d. (2016). *Sensor Kinect, inteligencia artificial al alcance de todos*. Obtenido de programarfacil: <https://programarfacil.com/podcast/86-sensor-kinect-inteligencia-artificial/>
- I/O, F. (Junio de 2019). *FACTORY I/O*. Obtenido de icl didactica: <https://www.icl-didactica.com/factory-i-o/>
- Industria 4.0, la cuarta revolución industrial y la inteligencia operacional*. (14 de Enero de 2019). Obtenido de <https://www.cic.es/industria-40-revolucion-industrial/>
- Moreno, E. G. (199). *Automatización de Procesos Industriales, Robótica y Automática*. Universidad Politécnica de Valencia. Valencia: Universitat Politècnica de València.
- National Instruments. (2019). *¿Que es LabView?* Obtenido de National Instruments: <https://www.ni.com/es-es/shop/labview.html>
- PDAControl. (12 de Julio de 2016). *Introducción a Servidores OPC*. Obtenido de PDAControl: <http://pdacontroles.com/introduccion-servidores-opc/>
- Satoshi. (12 de Marzo de 2017). *Codesys ®: IL, SFC, LD, FBD o ST, ¿Cuál elijo?* Obtenido de opiron electronics: <https://www.opiron.com/2017/03/12/codesys-mejor-lenguaje-programacion/>
- Wikipedia. (16 de Junio de 2019). *Controlador lógico programable*. Obtenido de Wikipedia: https://es.wikipedia.org/wiki/Controlador_l%C3%B3gico_programable
- IEC (2018a) *IEC 60870-5-104*. Disponible en: <https://webstore.iec.ch/publication/25035> (Accedido: 16 de junio de 2018).
- IEC (2018b) *IEC 61131*. Disponible en: <https://webstore.iec.ch/webstore/webstore.nsf/mysearchajax?Openform&key=iec61131&sorting=&start=1&onglet=1> (Accedido: 16 de junio de 2018).

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

IEC (2018c) *IEC 62541*. Disponible en: [https://webstore.iec.ch/searchform&q=iec 62541](https://webstore.iec.ch/searchform&q=iec%2062541) (Accedido: 16 de junio de 2018).

AENOR (2016) *Norma UNE-EN 60848:2013*. Disponible en: [http://www.aenor.es/aenor/normas/normas/fichanorma.asp?tipo=N&codigo=N0051395#.WyaxHVX 7TIV](http://www.aenor.es/aenor/normas/normas/fichanorma.asp?tipo=N&codigo=N0051395#.WyaxHVX7TIV) (Accedido: 17 de junio de 2018).

Apuntes de Proyectos Tema 4 Presupuestos, cuadros de precios y pliego de condiciones. Departamento de Proyectos de Ingeniería.

Apuntes de Proyectos Tema 2 Fases y ciclo de vida del Proyecto. Departamento de Proyectos de Ingeniería.

Apuntes de Aula de TAU de y Laboratorio de Automatización y Control. Seminario de Automatización. València: UPV

Piedrafita Moreno, R. (2003). *Ingeniería de la automatización industrial*. 2ª. Editado por RA-MA.

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



ESCUELA TÉCNICA
SUPERIOR INGENIEROS
INDUSTRIALES VALENCIA

TRABAJO FIN DE GRADO EN INGENIERÍA EN TECNOLOGÍAS INDUSTRIALES

**PROYECTO DE AUTOMATIZACIÓN DE UN
SISTEMA DE ALMACENAMIENTO DE
PIEZAS CON PLC TSX-PREMIUM Y M241 DE
SCHNEIDER Y SCADA MEDIANTE
APLICACIÓN LABVIEW.**

ANEXO 1: DECLARACIÓN DE VARIABLES

AUTOR: PABLO GÓMEZ GALLARDO

TUTOR: RAÚL SIMARRO FERNÁNDEZ

Curso Académico: 2019-20

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

1.INTRODUCCIÓN

En este documento se adjuntan las tablas con las variables internas que aparecen en la programación. De ellas se indica su nombre, tipo de dato, dirección de memoria del PLC y definición. De estos campos, los 2 últimos, en aquellas variables en Unity Pro que se habían creado como transiciones desde una sección SFC y que aparecen bloqueadas con un candado el programa, las ubicaba en direcciones internas desconocidas que no se muestran y que no es posible modificar. Tampoco se permite añadirle un comentario, por lo que será en el siguiente Anexo Diseño del Automatismo donde aparecen definida su función.

Se han clasificado las variables según el programa donde se alojan habiendo Unity Pro y Somachine. La primera imagen muestra el elemento del que se trata almacén o brazo robot y a continuación una Tabla donde aparecen registradas sus entradas (sensores) y salidas (actuadores) junto con su dirección en memoria en este caso sería la Dirección TSX para el almacén ya que está gobernada por el autómeta TSX- Premium y la Dirección M241 para el brazo porque es el autómeta con el mismo nombre el que le gobierna.

En último lugar, aparece la tabla con todas las variables que desde el SCADA, es decir desde el LabView, son compartidas primero con el Unity Pro y luego con el Somachine junto con su dirección en memoria la cual se configuró personalmente además de una breve definición de su función. Cabe mencionar que el lenguaje que se empleó para definir las variables es un lenguaje coloquial que facilitase la comprensión del programador en el desarrollo del proyecto por eso se verán nombres del tipo "libreC"o "llenoCOND".

2. VARIABLES EN UNITY PRO

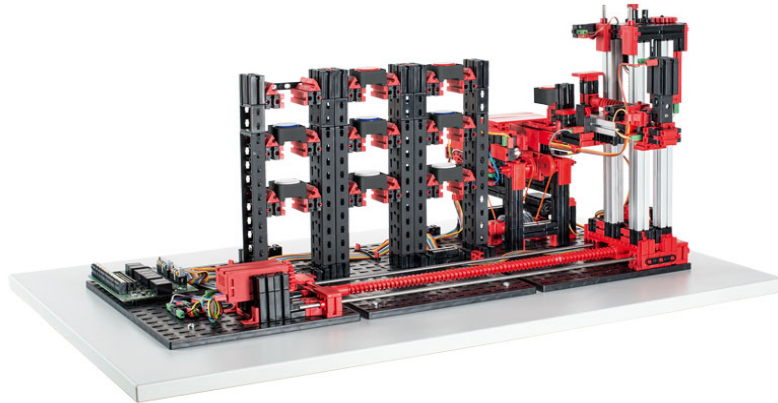


Figura 42. Almacén Vertical

Entrada	Descripción	Dirección M241	Dirección CJ2M	Dirección TSX	Dirección S7-1200
I1	Final de carrera referencia horizontal	%IX0.4	000.4	%IO.2.4	%IO.4
I2	Sensor barrera pieza dentro	%IX0.5	000.5	%IO.2.5	%IO.5
I3	Sensor barrera pieza fuera	%IX0.6	000.6	%IO.2.6	%IO.6
I4	Final de carrera referencial vertical	%IX0.7	000.7	%IO.2.7	%IO.7
I5	Final de carrera plataforma delante	%IX1.0	000.8	%IO.2.8	%I1.0
I6	Final de carrera plataforma detrás	%IX1.1	000.9	%IO.2.9	%I1.1
A1	Pulso pieza recorrido entrada (lower)	%IX0.0	000.0	%IO.2.0	%IO.0
A2	Pulso pieza recorrido entrada (upper)	%IX1.2	000.10	%IO.2.10	%I1.2
B1	Pulsos encoder movimiento horizontal	%IX0.1	000.1	%IO.2.1	%IO.1
B3	Pulsos encoder movimiento vertical	%IX0.2	000.2	%IO.2.2	%IO.2

Salida	Descripción	Dirección M241	Dirección CJ2M	Dirección TSX	Dirección S7-1200
Q1	Motor movimiento cinta entrada atrás	%QX0.4	001.0	%Q0.2.16	%Q0.0
Q2	Motor movimiento cinta entrada adelante	%QX0.5	001.1	%Q0.2.17	%Q0.1
Q3	Motor movimiento avance horizontal	%QX0.6	001.2	%Q0.2.18	%Q0.2
Q4	Motor movimiento retroceso horizontal	%QX0.7	001.3	%Q0.2.19	%Q0.3
Q5	Motor movimiento vertical abajo	%QX1.0	001.4	%Q0.2.20	%Q0.4
Q6	Motor movimiento vertical arriba	%QX1.1	001.5	%Q0.2.21	%Q0.5
Q7	Motor movimiento plataforma adelante	%QX1.2	001.6	%Q0.2.22	%Q0.6
Q8	Motor movimiento plataforma atrás	%QX1.3	001.7	%Q0.2.23	%Q0.7
Q9	Habilitar señales de entrada del proceso	%QX1.4	001.8	%Q0.2.24	%Q1.0

Tabla 2 Entradas y salidas del almacén. Fuente: Raúl Simarro Fernández

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

Nombre	Tipo	Dirección	Comentario
● auto	BOOL		Modo automatico
● AZ1	BOOL		La pieza que se va a introducir en la 1ª celda es azul
● AZ2	BOOL		La pieza que se va a introducir en la 2ª celda es azul
● AZ3	BOOL		La pieza que se va a introducir en la 3ª celda es azul
● AZ4	BOOL		La pieza que se va a introducir en la 4ª celda es azul
● AZ5	BOOL		La pieza que se va a introducir en la 5ª celda es azul
● AZ6	BOOL		La pieza que se va a introducir en la 6ª celda es azul
● AZ7	BOOL		La pieza que se va a introducir en la 7ª celda es azul
● AZ8	BOOL		La pieza que se va a introducir en la 8ª celda es azul
● AZ9	BOOL		La pieza que se va a introducir en la 9ª celda es azul
● BL1	BOOL		La pieza que se va a introducir en la 1ª celda es blanca
● BL2	BOOL		La pieza que se va a introducir en la 2ª celda es blanca
● BL3	BOOL		La pieza que se va a introducir en la 3ª celda es blanca
● BL4	BOOL		La pieza que se va a introducir en la 4ª celda es blanca
● BL5	BOOL		La pieza que se va a introducir en la 5ª celda es blanca
● BL6	BOOL		La pieza que se va a introducir en la 6ª celda es blanca
● BL7	BOOL		La pieza que se va a introducir en la 7ª celda es blanca
● BL8	BOOL		La pieza que se va a introducir en la 8ª celda es blanca
● BL9	BOOL		La pieza que se va a introducir en la 9ª celda es blanca
● color	BOOL		Color detectado y modo de ordenación activado
● CuentaHorizon...	INT		Cuenta los pulsos del encoder en el movimiento horizontal
● CuentaVertical	INT		Cuenta los pulsos del encoder en el movimiento vertical
● h1	BOOL		Distancia en horizontal de la 1ª columna
● h2	BOOL		Distancia en horizontal a la 2ª columna
● h3	BOOL		Distancia en horizontal a la 3ª columna
● INICIO	BOOL		
● lib 1	BOOL		
● lib 2	BOOL		
● lib 3	BOOL		
● lib 4	BOOL		
● lib 5	BOOL		
● lib 6	BOOL		
● lib 7	BOOL		
● lib 8	BOOL		
● lib 9	BOOL		
● libreC	BOOL		almacén libre
● llenoCOND	BOOL		
● manu	BOOL		modo manual
● metiendo	BOOL		
● o 1	BOOL		
● o 2	BOOL		
● o 3	BOOL		

Tabla 3. Variables internas

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

o4	BOOL		
o5	BOOL		
o6	BOOL		
o7	BOOL		
o8	BOOL		
o9	BOOL		
prefM1	BOOL		
prefM2	BOOL		
prefM3	BOOL		
prefM4	BOOL		
prefM5	BOOL		
prefM6	BOOL		
prefM7	BOOL		
prefM8	BOOL		
prefM9	BOOL		
prefM10	BOOL		
prefM11	BOOL		
prefM12	BOOL		
prefM13	BOOL		
prefM14	BOOL		
prefM15	BOOL		
prefM16	BOOL		
prefS1	BOOL		
prefS2	BOOL		
prefS3	BOOL		
prefS4	BOOL		
prefS5	BOOL		
prefS6	BOOL		
prefS7	BOOL		
prefS8	BOOL		
prefS9	BOOL		
prefS10	BOOL		
prefS11	BOOL		
prefS12	BOOL		
prefS13	BOOL		
prefS14	BOOL		
prefS15	BOOL		
prefS16	BOOL		

Tabla 4. Variables internas

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

R1	BOOL		La pieza que se va a introducir en la 1ª celda es roja
R2	BOOL		La pieza que se va a introducir en la 2ª celda es roja
R3	BOOL		La pieza que se va a introducir en la 3ª celda es roja
R4	BOOL		La pieza que se va a introducir en la 4ª celda es roja
R5	BOOL		La pieza que se va a introducir en la 5ª celda es roja
R6	BOOL		La pieza que se va a introducir en la 6ª celda es roja
R7	BOOL		La pieza que se va a introducir en la 7ª celda es roja
R8	BOOL		La pieza que se va a introducir en la 8ª celda es roja
R9	BOOL		La pieza que se va a introducir en la 9ª celda es roja
sacando	BOOL		
sneg	BOOL		No se esta pulsando ningun botón de extraer pieza
stoped	BOOL		El SFC automático está detenido
v1a	BOOL		Primer movimiento vertical en la 1ª fila
v2a	BOOL		Primer movimiento vertical en la 2ª fila
v2b	BOOL		Segundo movimiento vertical en la 2ª fila
v2c	BOOL		Segundo movimiento vertical en la 3ª fila
v3a	BOOL		Tercer movimiento vertical en la 1ª fila
v3b	BOOL		Tercer movimiento vertical en la 2ª fila
v3c	BOOL		Tercer movimiento vertical en la 3ª fila
V5	BOOL		
vC	BOOL		almacén vacío
X11	BOOL		
X12	BOOL		
X13	BOOL		
X21	BOOL		
X22	BOOL		
X23	BOOL		
X31	BOOL		
X32	BOOL		
X33	BOOL		

Tabla 5. Variables internas

AZUL	BOOL	%MW1.0	Color azul detectado por la camara
ROJO	BOOL	%MW1.1	Color rojo detectado por la cámara
BLANCO	BOOL	%MW1.2	Color blanco detectado por la cámara
s1	BOOL	%MW2.0	Sacar pieza celda 1
s2	BOOL	%MW2.1	Sacar pieza celda 2
s3	BOOL	%MW2.2	Sacar pieza celda 3
s4	BOOL	%MW2.3	Sacar pieza celda 4
s5	BOOL	%MW2.4	Sacar pieza celda 5
s6	BOOL	%MW2.5	Sacar pieza celda 6
s7	BOOL	%MW2.6	Sacar pieza celda 7
s8	BOOL	%MW2.7	Sacar pieza celda 8
s9	BOOL	%MW2.8	Sacar pieza celda 9
col	BOOL	%MW2.9	Modo de ordenación por color
disponible	BOOL	%MW2.10	Modo de ordenación por celda disponible
vaciado	BOOL	%MW2.11	Modo de extracción vaciado automático
modo	INT	%MW3	Modo de funcionamiento
stop	BOOL	%MW4.0	Boton de emrgencia del almacén
Boton_Q1	BOOL	%MW5.0	Botón motor movimiento cinta entrada atrás modo manual
Boton_Q2	BOOL	%MW5.1	Botón motor movimiento cinta entrada adelante modo manual
Boton_Q3	BOOL	%MW5.2	Botón motor movimiento avance horizontal modo manual
Boton_Q4	BOOL	%MW5.3	Botón motor movimiento retroceso horizontal modo manual
Boton_Q5	BOOL	%MW5.4	Botón motor movimiento vertical abajo modo manual
Boton_Q6	BOOL	%MW5.5	Botón motor movimiento vertical arriba modo manual
Boton_Q7	BOOL	%MW5.6	Botón motor movimiento plataforma adelante modo manual
Boton_Q8	BOOL	%MW5.7	Botón motor movimiento plataforma atrás modo manual
pieza	BOOL	%MW20.0	La caja se encuentra en la posición para que el brazo actúe

Tabla 6. Variables compartidas con el SCADA

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

3. VARIABLES EN SOMACHINE

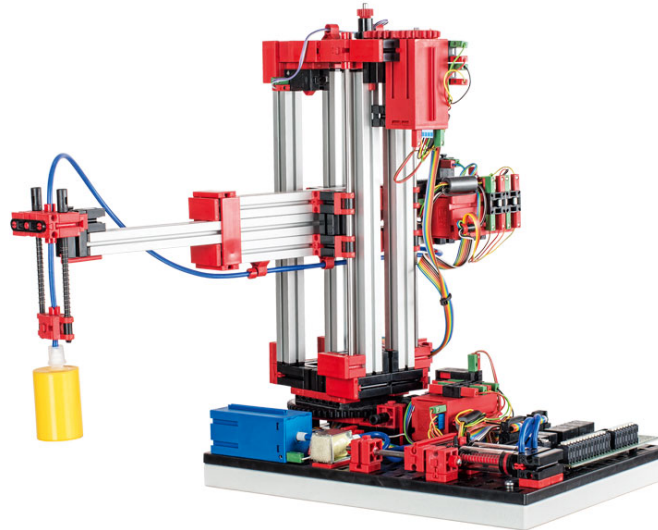


Figura 43. Brazo ventosa

Entrada	Descripción	Dirección	Dirección	Dirección	Dirección
		M241	CJ2M	TSX	S7-1200
I1	Final de carrera referencia vertical	%IX0.4	000.4	%IO.2.4	%IO.4
I2	Final de carrera referencia horizontal	%IX0.5	000.5	%IO.2.5	%IO.5
I3	Final de carrera referencia giro	%IX0.6	000.6	%IO.2.6	%IO.6
B1	Pulsos encoder movimiento vertical	%IX0.0	000.0	%IO.2.0	%IO.0
B3	Pulsos encoder movimiento horizontal	%IX0.1	000.1	%IO.2.1	%IO.1
B5	Pulsos encoder movimiento giratorio	%IX0.2	000.2	%IO.2.2	%IO.2

Salida	Descripción	Dirección	Dirección	Dirección	Dirección
		M241	CJ2M	TSX	S7-1200
Q1	Motor movimiento vertical arriba	%QX0.4	001.0	%Q0.2.16	%Q0.0
Q2	Motor movimiento vertical abajo	%QX0.5	001.1	%Q0.2.17	%Q0.1
Q3	Motor movimiento retroceso horizontal	%QX0.6	001.2	%Q0.2.18	%Q0.2
Q4	Motor movimiento avance horizontal	%QX0.7	001.3	%Q0.2.19	%Q0.3
Q5	Motor movimiento giro horario	%QX1.0	001.4	%Q0.2.20	%Q0.4
Q6	Motor movimiento giro antihorario	%QX1.1	001.5	%Q0.2.21	%Q0.5
Q7	Habilitar compresor	%QX1.2	001.6	%Q0.2.22	%Q0.6
Q8	Succión ventosa	%QX1.3	001.7	%Q0.2.23	%Q0.7
Q9	Habilitar señales de entrada del proceso	%QX1.4	001.8	%Q0.2.24	%Q1.0

Raúl Simarro Fernández – Departamento de Ingeniería de Sistemas y Automática (UPV)

Tabla 7. Entradas y salidas del brazo. Fuente: Raúl Simarro Fernández

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

16	VAR_GLOBAL	CuentaVertical		INT		Variables cuenta los pulsos del encoder motor vertical
17	VAR_GLOBAL	CuentaHorizontal		INT		Variables cuenta los pulsos del encoder motor horizontal
18	VAR_GLOBAL	CuentaGiro		INT		Variables cuenta los pulsos del encoder motor del giro
19	VAR_GLOBAL	MaxCuentaVertical		INT	1000	Límite de pulsos del movimiento vertical del brazo
20	VAR_GLOBAL	MaxCuentaHorizontal		INT	1000	Límite de pulsos del movimiento horizontal del brazo
21	VAR_GLOBAL	MaxCuentaGiro		INT	750	Límite de pulsos del movimiento de giro del brazo
22	VAR_GLOBAL	Boton_Q1	%MX10.0	BOOL		Botones control manual SCADA
23	VAR_GLOBAL	Boton_Q2	%MX10.1	BOOL		
24	VAR_GLOBAL	Boton_Q3	%MX10.2	BOOL		
25	VAR_GLOBAL	Boton_Q4	%MX10.3	BOOL		
26	VAR_GLOBAL	Boton_Q5	%MX10.4	BOOL		
27	VAR_GLOBAL	Boton_Q6	%MX10.5	BOOL		
28	VAR_GLOBAL	modo	%MW3	INT		modo de funcionamiento
29	VAR_GLOBAL	stop	%MX8.0	BOOL		stop de emergencia
30	VAR_GLOBAL	manual		BOOL		modo manual
31	VAR_GLOBAL	Haycaja	%MX2.0	BOOL		La caja del almacén ya esta lista. Inicio del proceso
32	VAR_GLOBAL	mete	%MX2.1	BOOL		Coger pieza
33	VAR_GLOBAL	saca	%MX2.2	BOOL		Sacar pieza
34	VAR_GLOBAL	almacen	%MX4.0	BOOL		La pieza ya ha sido extrída/colocada. El almacén vuelve a funcionar.
35	VAR_GLOBAL	soltar	%MX10.8	BOOL		Soltar pieza desde modo manual
36	VAR_GLOBAL	Boton_Q7	%MX10.6	BOOL		Botón compresor
37	VAR_GLOBAL	Boton_Q8	%MX10.7	BOOL		Botón ventosa

Tabla 8. Variables globales

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



ESCUELA TÉCNICA
SUPERIOR INGENIEROS
INDUSTRIALES VALENCIA

TRABAJO FIN DE GRADO EN INGENIERÍA EN TECNOLOGÍAS INDUSTRIALES

**PROYECTO DE AUTOMATIZACIÓN DE UN
SISTEMA DE ALMACENAMIENTO DE
PIEZAS CON PLC TSX-PREMIUM Y M241 DE
SCHNEIDER Y SCADA MEDIANTE
APLICACIÓN LABVIEW.**

ANEXO 2: DISEÑO DEL AUTOMATISMO

AUTOR: PABLO GÓMEZ GALLARDO

TUTOR: RAÚL SIMARRO FERNÁNDEZ

Curso Académico: 2019-20

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

1. INTRODUCCIÓN

Este documento muestra la lógica aplicada en la programación de los procesos los cuales se ilustrarán mediante lenguaje gráfico GRAFCET. Debido al gran tamaño de algunas transiciones, estas no se pueden incluir dentro del esquema y se añadirán imágenes del código en Ladder que se usó. Para este apartado se distingue entre el almacén vertical y el brazo robot programados en Unity Pro y Somachine respectivamente. Debido a la amplitud del programa, no aparece una sola secuencia global, sino que se crearon muchos Grafkets parciales y así, se moduló el problema en pequeños problemas con los que resulta más fácil trabajar. Se señalarán tal y como se implementaron, aquellas funciones que no tienen una representación sencilla en el lenguaje gráfico y que deben entenderse fuera de estos.

Tal y como se ha dicho, no existe un Grafket total, sino que está dividido en Grafkets parciales que poseen una funcionalidad propia. Los títulos de los apartados son en consecuencia las funciones de estos Grafkets parciales. Se seguirá una nomenclatura de los estados de acuerdo con la norma **UNE-EN 60848:2013** Lenguaje de especificación GRAFCET. Se usará una letra que se repite en todo el esquema de un mismo apartado acompañada de un número que dependerá del estado.

2. UNITY PRO - ALMACÉN.

El almacén como se dijo, está formado por nueve celdas de manera que para algunas funciones se crearon nueve Grafctets o nueve ramas sobre un mismo Grafctet. En este apartado solo se mostrará una celda, la primera, ya que las ocho restantes son idénticas. El resto de celdas, en cuanto a la nomenclatura se refiere, mantienen la letra propia del apartado y sus números continúan con la progresión decimal que para la primera celda se ha utilizado. Esto quiere decir que si en el Grafctet de la primera celda se contaron los estados 0,1,2 y 3 para la segunda celda se aplicarán los números 4,5,6 y 7 (siempre acompañados de la misma letra).

En la imagen de la Figura 44, se muestra el número que se le asigna a cada celda para dirigirse a ella a lo largo de la programación. Esta imagen también muestra el orden de prioridad que se sigue en las distintas formas de llenar y vaciar el almacén y que se verán a continuación.

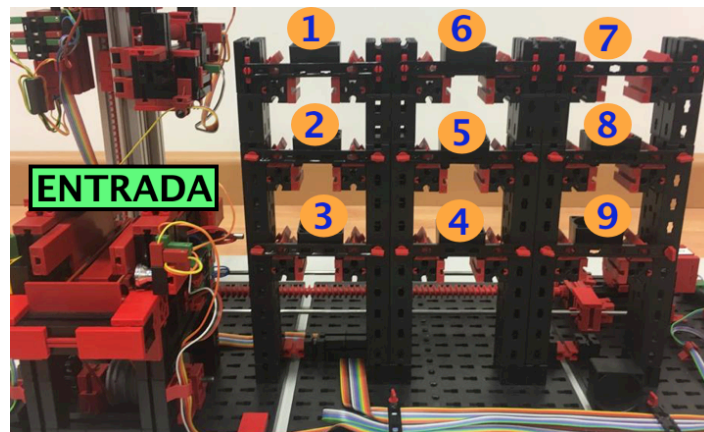


Figura 44. Orden de numeración de las celdas. Fuente: elaboración propia

A continuación, se relacionarán los distintos modos de ordenación que se ofrecen desde el SCADA con las variables correspondientes utilizadas para su programación y que aparecen publicadas en el Anexo Declaración de Variables. La explicación de estos modos queda recogida con gran detalle en el próximo Anexo, Manual de Usuario:

- En la parte de Extraer Piezas (Figura 71):
 - El interruptor que activa el vaciado automático bajo el título de vaciar corresponde a la variable ("vaciado").
 - Los botones colocados debajo de la celda aparecen definidos como (s1, s2, s3, s4, s5, s6, s7, s8, s9). Se creó una variable "sneg" que comprueba que ninguno de estos botones está pulsado.
- En la parte de Meter Piezas (Figura 67):
 - El interruptor de color corresponde con la variable "col", a su vez la activación de esta, junto con la detección de un color por el programa en LabVIEW de la cámara activan la variable "color" como muestra la Figura 45.
 - El botón de disponible está asociado con la variable "disponible".

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

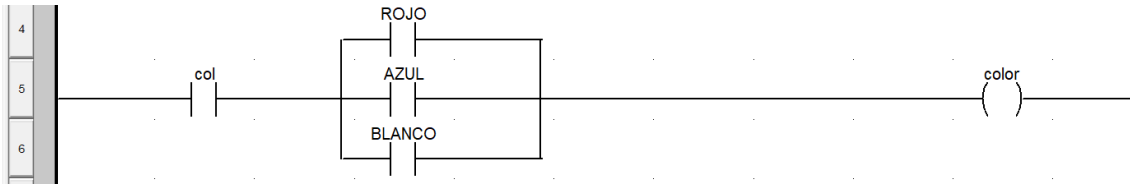
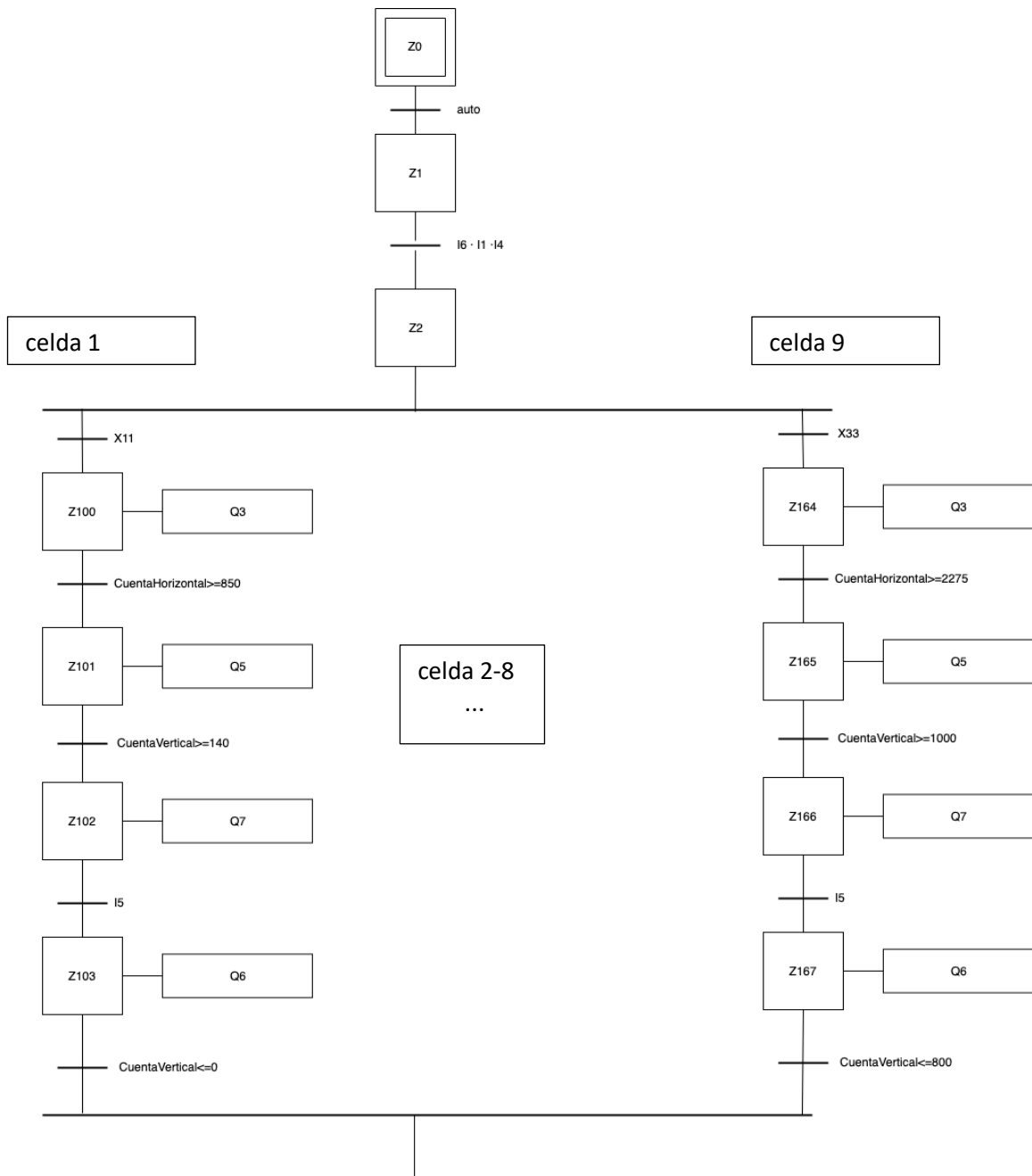


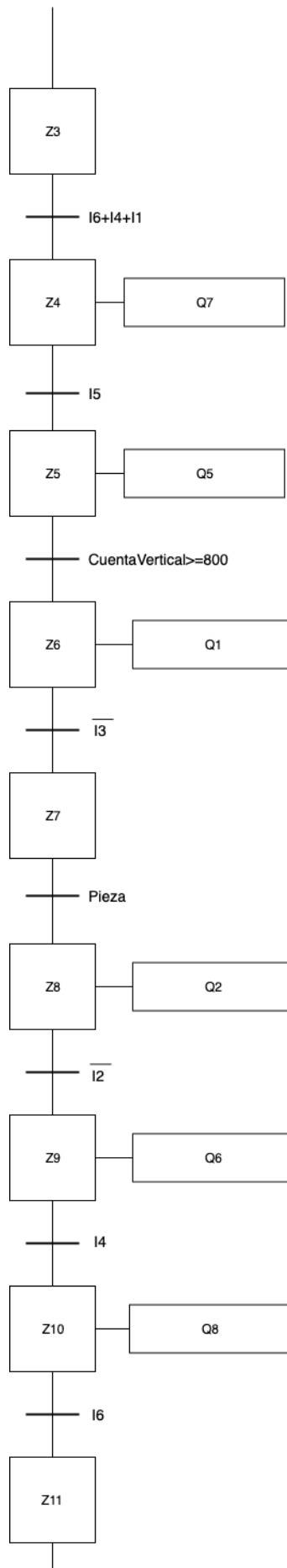
Figura 45. Definición en Ladder de la variable color.

2.1 Graficet Principal (Z)

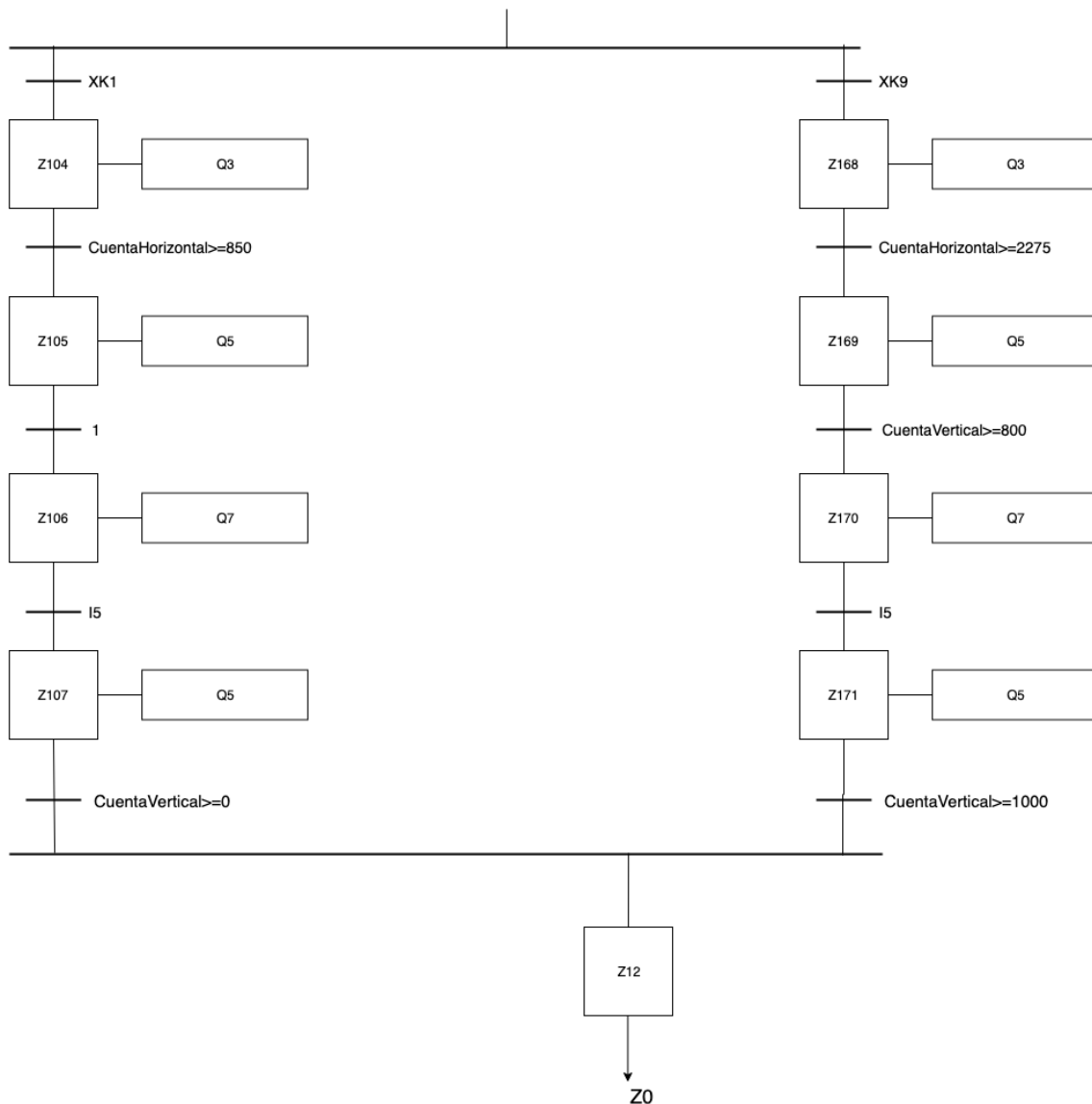
Contiene la secuencia del automatismo con los movimientos que realiza el almacén para meter y sacar piezas. Solo se muestra la primera y última celda, las 7 restantes son idénticas solo cambian los valores de las variables "Cuenta" que dependen de la localización de la celda.



Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.



Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.



La estructura es sencilla, en la primera parte, común, se vuelve a la posición inicial siempre que este activado el modo automático. Después, dependiendo del modo de ordenación/extracción que se encuentre el almacén y de su estado, se seguirá una de las 9 ramas que corresponden a cada una de las celdas (aquí solo queda representada la primera), lo que queda registrado en el Grafcet de Celda en Proceso que luego se mostrará.

Da igual que se esté metiendo o sacando ya que la secuencia es la misma. La primera parte es la de coger la caja de la celda, donde se observan las transiciones que comparan el valor del encoder en las variables "CuentaVertical/Horizontal" con el que se calculó en la calibración. La segunda parte, común a todas las celdas, llevaría la caja hacia la entrada mediante las cintas donde esperaría hasta que el brazo haya operado momento en que se activa la variable "pieza". Por último, la caja vacía o llena según el caso vuelve a su celda (segunda ramificación). El Grafcet sabe que celda era ya que, en el Grafcet de Celda en Proceso que se comentaba, permanece guardado el número de esta.

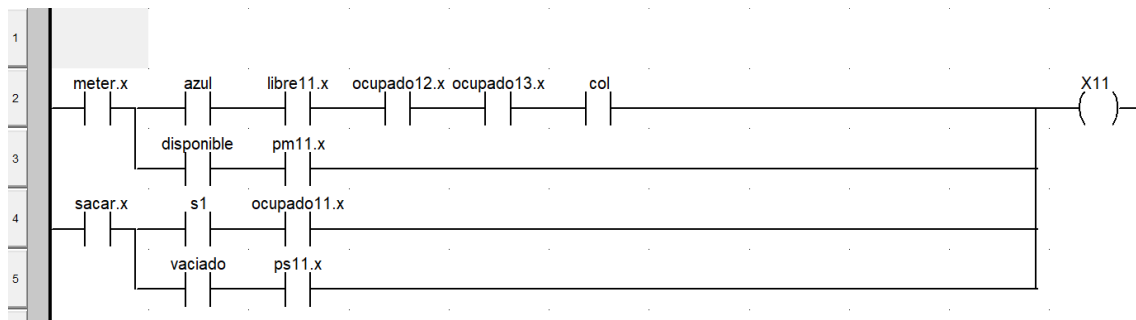


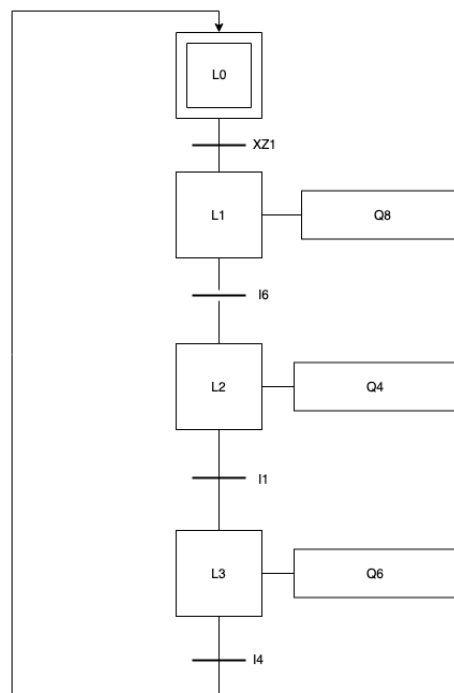
Figura 46. Transición en Ladder sobre la elección de pieza

En la Figura 46 se observan la estructura en lenguaje Ladder que el Grafcet principal utiliza al principio para elegir la celda con la que se trabajará ya sea para meter o sacar. Hay 2 posibilidades

- Si se quiere meter pieza: se observan 2 ramas en paralelo una para el modo que ordena según el color de pieza y la de abajo para el modo de ordenación por celda disponible para ello la celda tiene que estar vacía(disponible) y el Grafcet de preferencias debe mostrarse activado para dicha celda ("pm").
- Si se quiere sacar: el primer camino corresponde con el modo en el que el usuario elige que pieza sacar simplemente se comprueba que la celda este ocupada. La rama inferior corresponde con el vaciado automático donde se asegura la ocupación de la celda y su preferencia para sacar ("ps"), algo parecido con lo que ocurre en la parte de meter.

2.2 Volver a posición inicial (L)

Este esquema se activa nada más comenzar la secuencia principal.



2.3 Meter/Sacar (T)

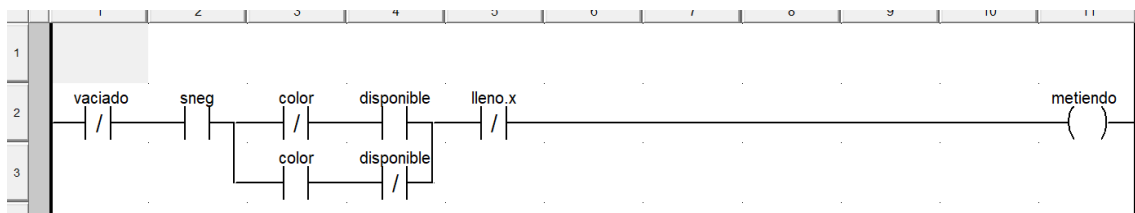
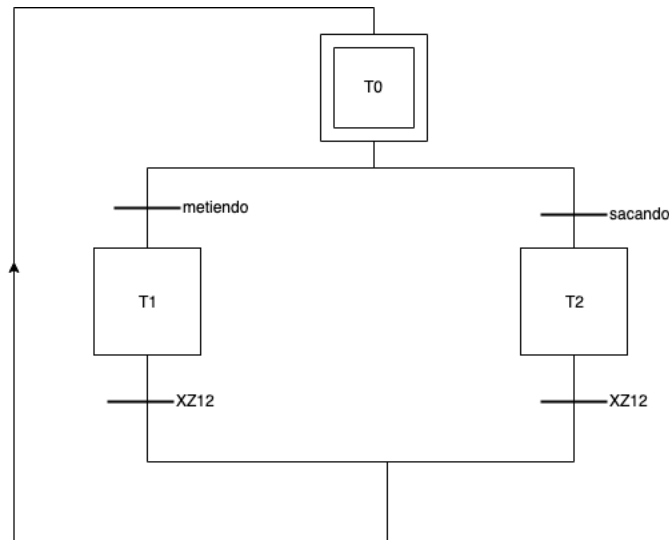


Figura 47. Transición en Ladder para meter

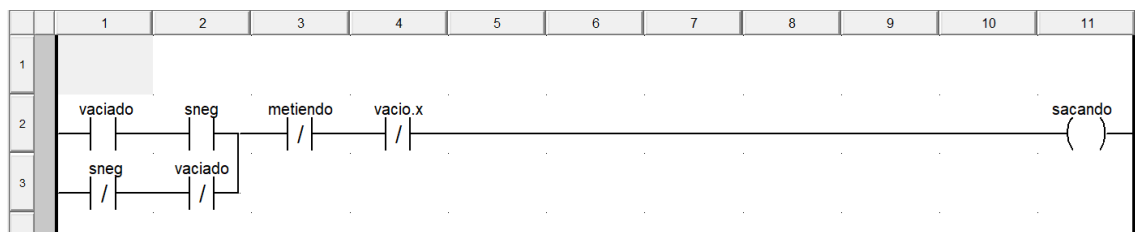


Figura 48. Transición en Ladder para sacar

En ambas Figuras, 47 y 48, se asocia meter y sacar con un modo de ordenación concreto, se asegura que no haya 2 modos o más marcados y por último que el almacén cumple con las condiciones mínimas que requiere dicho modo, por ejemplo, si se quiere sacar una pieza, el almacén no esté vacío.

Aunque los movimientos del robot en el almacén son los mismos, es necesario saber si se está metiendo o sacando una pieza ya que la acción del brazo es completamente distinta. Esta información es almacenada en direcciones de memoria del tipo %MW que a su vez se escriben en el autómatas esclavo (el del brazo) para que disponga de esta información (Figura 49).

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

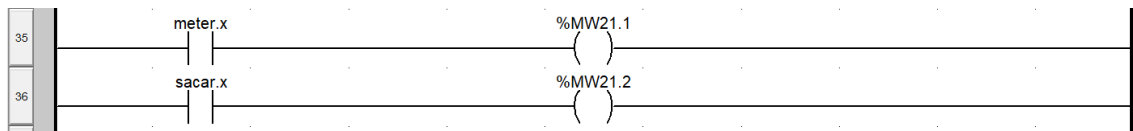
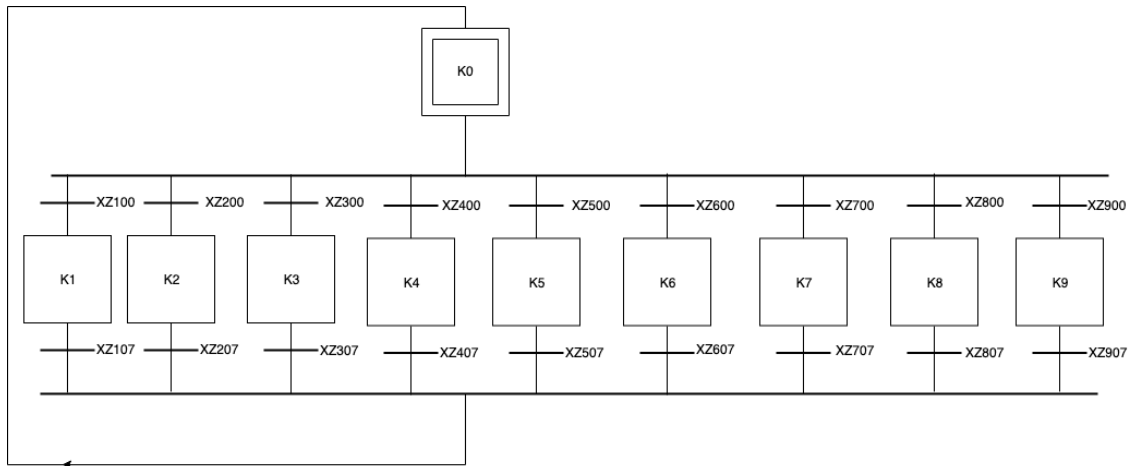


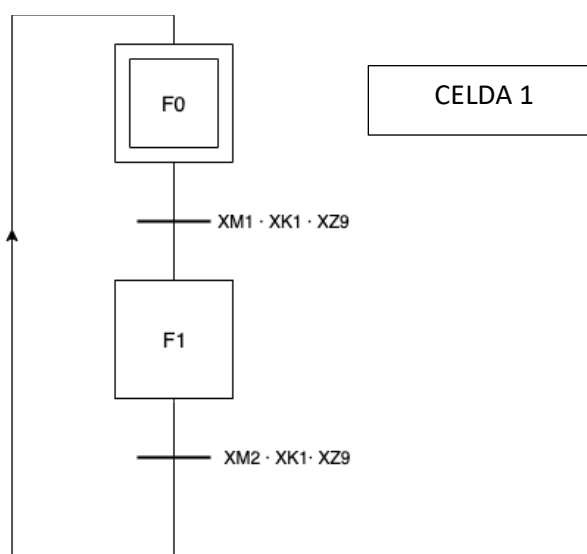
Figura 49. Configuración de la dirección de memoria de los estados meter y sacar

2.4 Celda en proceso (K)



Este Grafset al que se ha hecho referencia en el apartado 2.1 Grafset Principal permite conocer en que celda se está operando ya que esta información es requerida en varias ocasiones para el desarrollo del proceso. Su activación y desactivación depende del Grafset Principal desde el inicio de la manipulación (Z100) hasta su finalización (Z107).

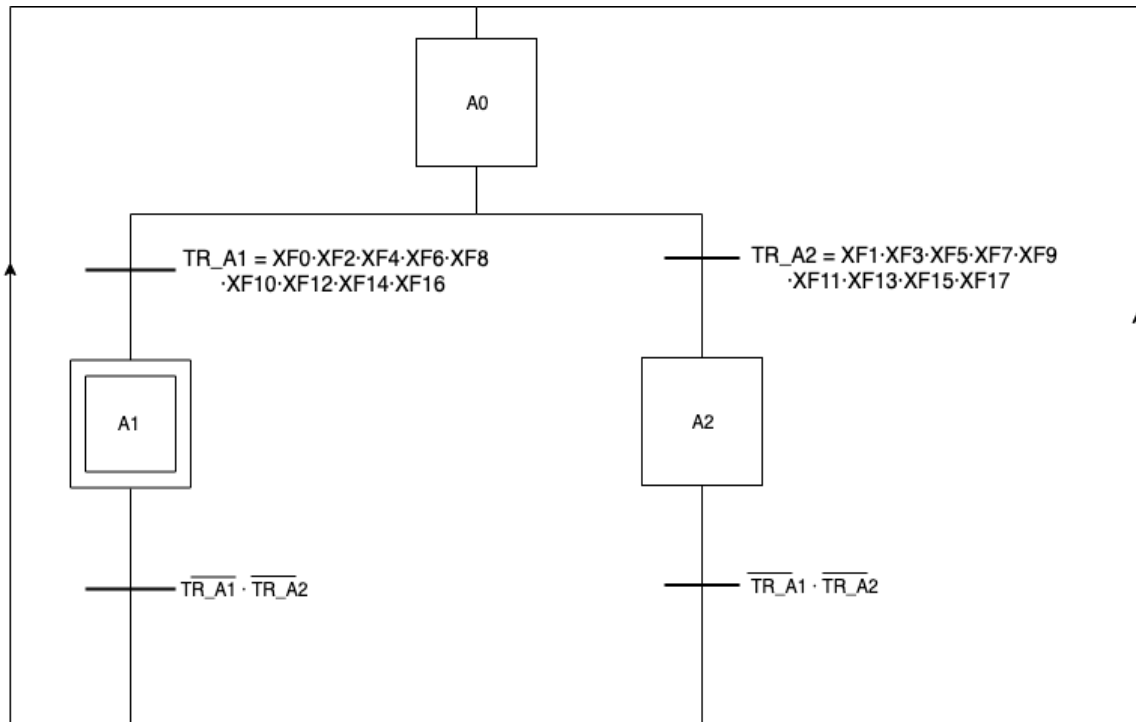
2.5 Estado celda (F)



Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

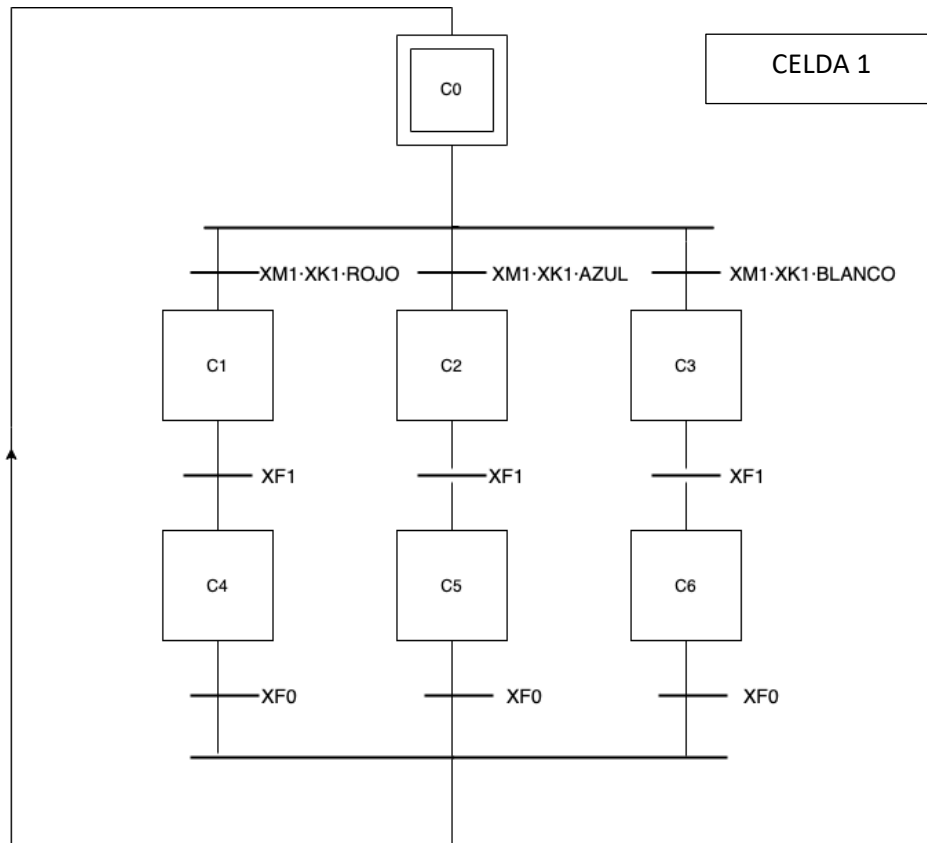
Alterna entre los dos estados en los que puede estar una celda: libre (F0) y ocupada (F1). La primera transición aparece definida en el Anexo anterior como (o1) y la segunda como (I1). Lo mismo para el resto de celdas.

2.6 Estado almacén (A)



El estado del almacén como elemento global, se puede encontrar vacío(A1), libre(A0) o lleno(A2). Las condiciones para que se active vacío o lleno, es que todas las celdas estén libres o todas las celdas estén ocupadas respectivamente. Si no está ni vacío ni lleno, se encuentra libre. El paso inicial como se puede deducir es vacío. En el anexo anterior la transición TR_A1 equivaldría a la variable "vC", "TR_A2" a "llenoCOND" y las dos transiciones restantes a "libreC".

2.7 Color Celda (C)



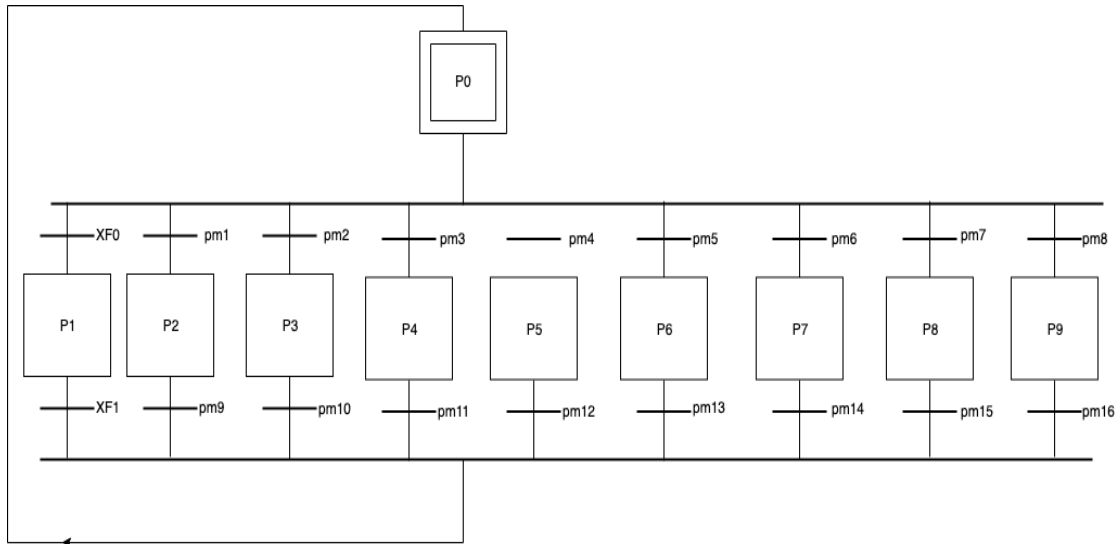
Lo forman tres columnas, una para cada color (Rojo, Azul y Blanco), y el paso inicial que está activado si no hay pieza. En el Anexo aparecen las primeras transiciones para cada color con el nombre de R1, AZ1 y BL1 en ese orden. Se añade un paso intermedio (C1, C2, C3) porque la variable del color (ROJO, AZUL, BLANCO) solo permanece activa mientras la cámara enfoca a la pieza. Una vez que la pieza, es agarrada, esa variable se desactiva y se quiere que el color aparezca cuando finalmente la celda es ocupada y no desde un inicio.

Los estados C4, C5 y C6 en el proyecto aparecen en el programa con el nombre de ROJO1, AZUL1 Y BLANCO1 respectivamente y dependiendo de cuál de ellos se encuentre activo se pasará un valor a la variable %MW30 como se observa en las primeras líneas de código de la Figura 50.

```
IF (ROJO1.X) THEN
  %MW30:=1;
ELSIF (AZUL1.X) THEN
  %MW30:=2;
ELSIF (BLANCO1.X) THEN
  %MW30:=3;
ELSE
  %MW30:=0;
```

Figura 50. Código en Unity Pro

2.8 Preferencia meter (P)

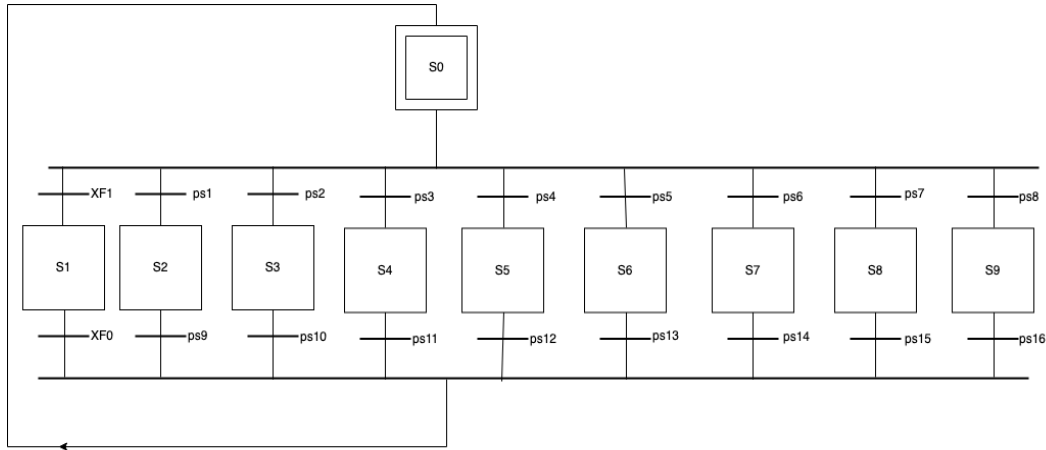


$$\begin{aligned}
 pm1 &= XF2 \cdot XF1 & pm9 &= XF3 + XF0 \\
 pm2 &= XF4 \cdot XF1 \cdot XF3 & pm10 &= XF5 + XF0 + XF2 \\
 pm3 &= XF6 \cdot XF1 \cdot XF3 \cdot XF5 & pm11 &= XF7 + XF0 + XF2 + XF4 \\
 pm4 &= XF8 \cdot XF1 \cdot XF3 \cdot XF5 \cdot XF7 & pm12 &= XF9 + XF0 + XF2 + XF4 + XF6 \\
 pm5 &= XF10 \cdot XF1 \cdot XF3 \cdot XF5 \cdot XF7 \cdot XF9 & pm13 &= XF11 + XF0 + XF2 + XF4 + XF6 + XF8 \\
 pm6 &= XF12 \cdot XF1 \cdot XF3 \cdot XF5 \cdot XF7 \cdot XF9 \cdot XF11 & pm14 &= XF13 + XF0 + XF2 + XF4 + XF6 + XF8 + XF10 \\
 pm7 &= XF14 \cdot XF1 \cdot XF3 \cdot XF5 \cdot XF7 \cdot XF9 \cdot XF11 \cdot XF13 & pm15 &= XF15 + XF0 + XF2 + XF4 + XF6 + XF8 + XF10 + XF12 \\
 pm8 &= XF16 \cdot XF1 \cdot XF3 \cdot XF5 \cdot XF7 \cdot XF9 \cdot XF11 \cdot XF13 \cdot XF15 & pm16 &= XF17 + XF0 + XF2 + XF4 + XF6 + XF8 + XF10 + XF12 + XF14
 \end{aligned}$$

Existe un método para meter piezas al que se le ha denominado primera celda disponible y consiste en que las piezas se irán guardando siempre en el orden de la Figura 44 independientemente de su color (más detallado en el Anexo Manual de Usuario). Esto quiere decir que: si por ejemplo las celdas 1,2 y 3 están ocupadas, la celda 4 es la siguiente donde hay que almacenar (siempre que ese modo de ordenación permanezca activado). Sin embargo, si el usuario decide extraer la pieza de la celda número 2, la prioridad para meter también cambia de la 4 a la 2.

Esto se consigue con el siguiente esquema gráfico en el que la condición de activación a cada una de las 9 etapas (una para cada celda) es que todas las celdas con mayor prioridad estén ocupadas y la propia celda libre y de desactivación que alguna de las celdas con mayor prioridad se libere o que la misma celda se ocupe. En las hojas del Anexo anterior, la terminología "pm" se sustituye por "prefM" junto con el mismo número.

2.9 Preferencia Sacar (S)



$$\begin{aligned}
 ps1 &= XF3 \cdot XF0 & ps9 &= XF2 + XF1 \\
 ps2 &= XF5 \cdot XF0 \cdot XF2 & ps10 &= XF4 + XF1 + XF3 \\
 ps3 &= XF7 \cdot XF0 \cdot XF2 \cdot XF4 & ps11 &= XF6 + XF1 + XF3 + XF5 \\
 ps4 &= XF9 \cdot XF0 \cdot XF2 \cdot XF4 \cdot XF6 & ps12 &= XF8 + XF1 + XF3 + XF5 + XF7 \\
 ps5 &= XF11 \cdot XF0 \cdot XF2 \cdot XF4 \cdot XF6 \cdot XF8 & ps13 &= XF10 + XF1 + XF3 + XF5 + XF7 + XF9 \\
 ps6 &= XF13 \cdot XF0 \cdot XF2 \cdot XF4 \cdot XF6 \cdot XF8 \cdot XF10 & ps14 &= XF12 + XF1 + XF3 + XF5 + XF7 + XF9 + XF11 \\
 ps7 &= XF15 \cdot XF0 \cdot XF2 \cdot XF4 \cdot XF6 \cdot XF8 \cdot XF10 \cdot XF12 & ps15 &= XF14 + XF1 + XF3 + XF5 + XF7 + XF9 + XF11 + XF13 \\
 ps8 &= XF17 \cdot XF0 \cdot XF2 \cdot XF4 \cdot XF6 \cdot XF8 \cdot XF10 \cdot XF12 \cdot XF14 & ps16 &= XF16 + XF1 + XF3 + XF5 + XF7 + XF9 + XF11 + XF13 + XF15
 \end{aligned}$$

Este esquema contiene las instrucciones necesarias para el vaciado automático de piezas. Esta opción, vacía el almacén con el mismo orden que el del apartado anterior Preferencia Meter y por tanto el que aparece en la Figura 44.

Para obtener este resultado se ha seguido una lógica muy similar a la anterior y consiste en que para activarse una celda y que por tanto sea la elegida para extraerle la pieza, todas las celdas con mayor prioridad deben estar vacías y la propia ocupada, en cambio, para desactivarse basta con que una de ellas se vuelva a llenar o que la celda en cuestión haya sido ya vaciada. En las hojas del Anexo anterior la terminología "ps" se sustituye por "prefS" junto con el mismo número.

2.10 Modos de funcionamiento y emergencia.

No se realizó forzado o encapsulación como la norma indica ya que no se dispone de este recurso en el programa. Se simuló esta organización jerárquica mediante funciones parecidas y de mayor utilidad como en el caso de la Figura 51 y cuya explicación detallada queda explicada en la memoria: Apartado 4.3.1 Programación del Proceso, Almacén Vertical - Unity Pro.

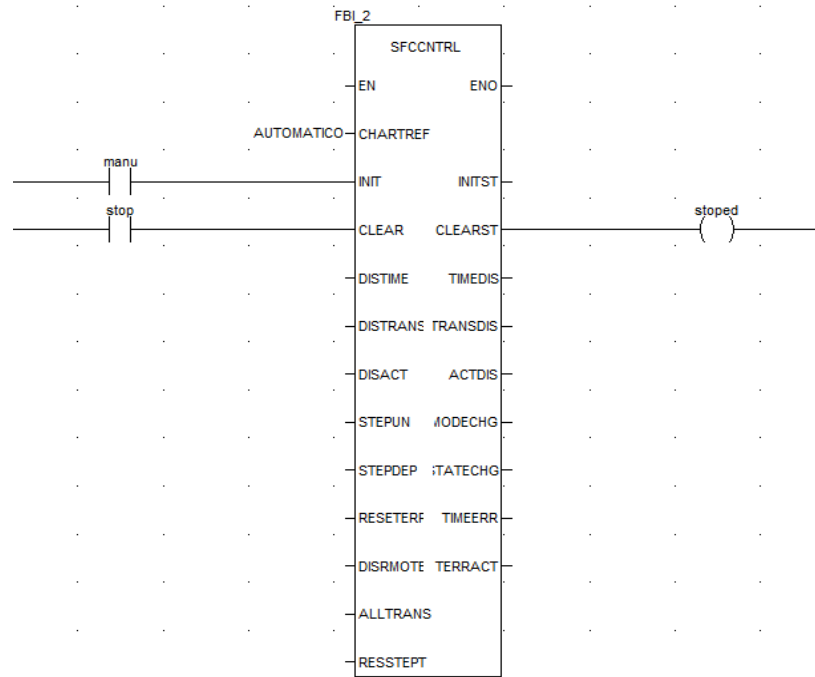


Figura 51. Bloque de funciones SFCCTRL

Otro recurso que se utilizó fue que en la sección de acciones, donde se encuentran todas las salidas del robot con los estados que las activan, condicionar a que ciertos modos estuvieran activos como muestra la Figura 52. Los botones que activan el modo manual necesitan que este esté activado y el "Stop" desactivado (esta última variable corresponde a la seta de emergencia) y aquellos estados del Grafset Principal, por tanto, que el modo manual se encuentre desactivado.

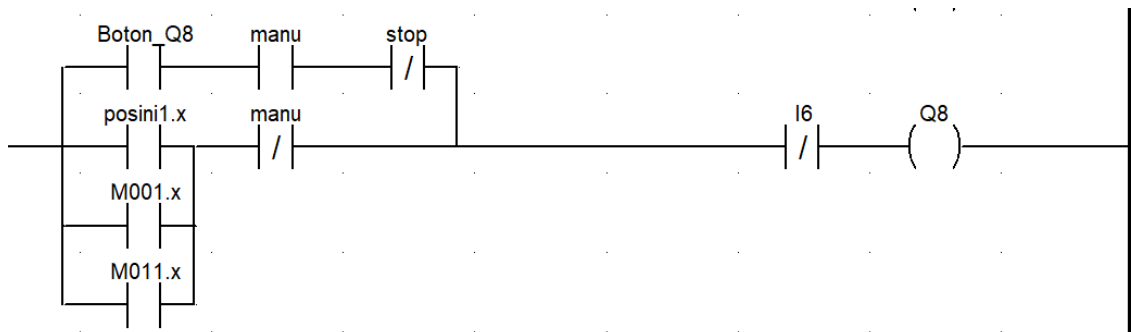


Figura 52. Condición para las salidas del sistema

3. SOMACHINE - BRAZO VENTOSA

La secuencia principal del brazo, como se dijo, aparece dividida en 2 Grafcets: uno que contiene los movimientos del robot y otro con las activaciones del compresor y la ventosa para atrapar las piezas.

3.1 Grafcet Principal de movimientos (B)

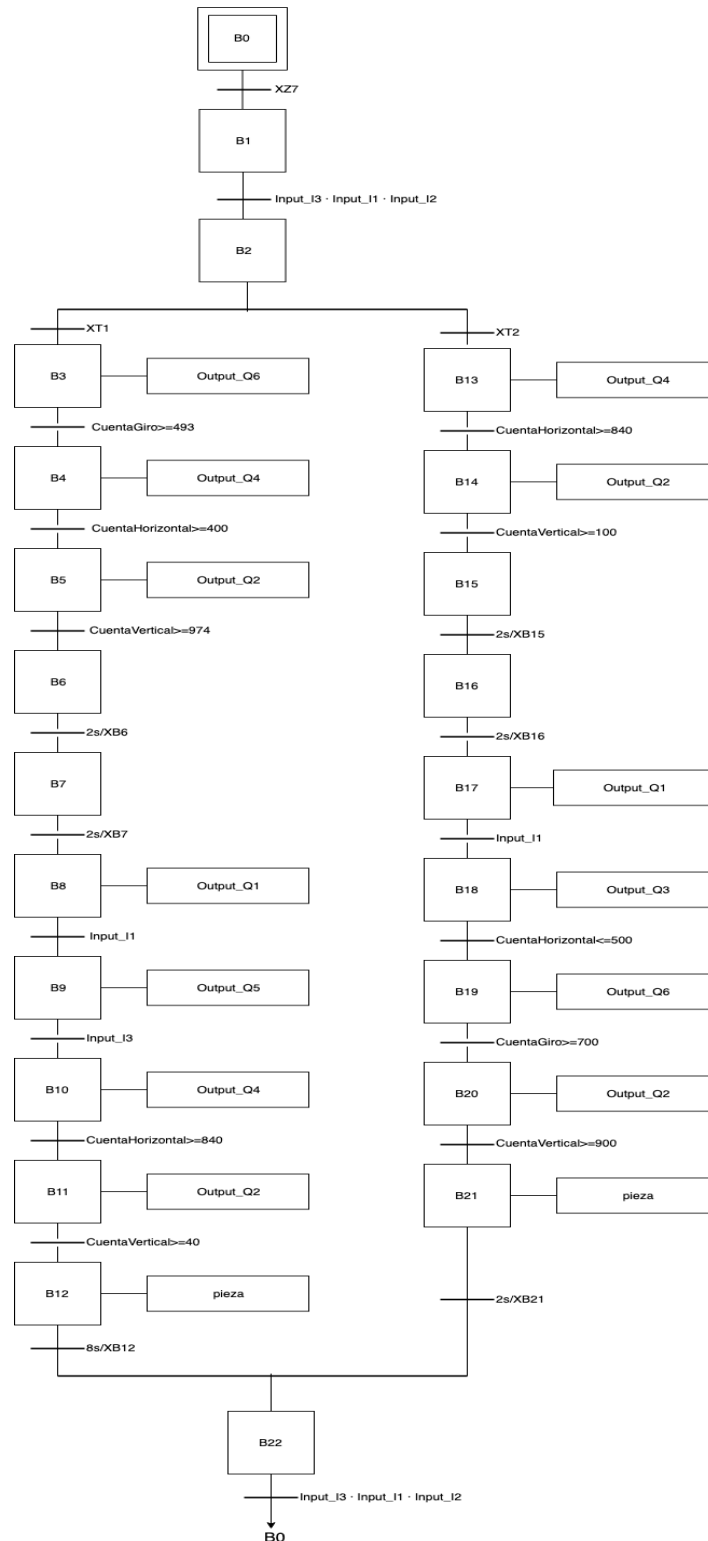
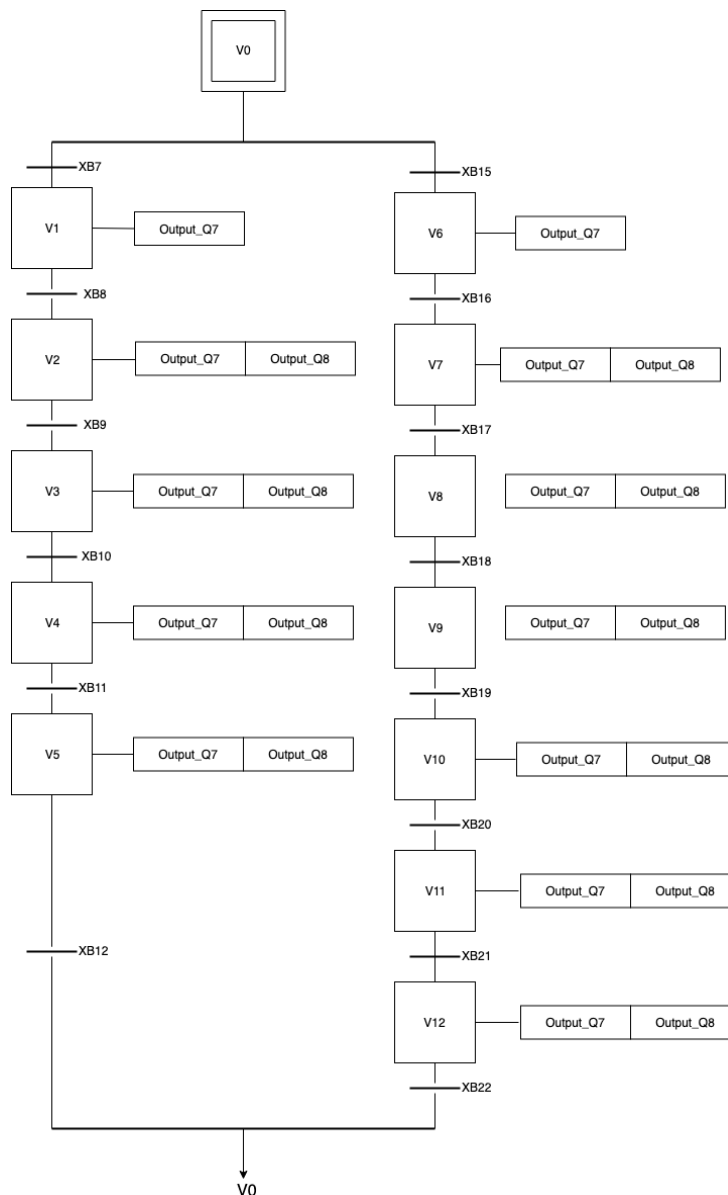




Figura 53. Transición de Inicio del Grafset de movimientos

Este Grafset se pone en marcha si hay una caja esperando en el almacén (Hay caja) y el stop está desactivado como muestra la Figura 53. Después, vuelve a la posición inicial a través del Grafset del apartado 3.3. Hay dos ramificaciones: la de la izquierda para meter piezas y la de la derecha para sacarlas, ambas posibilidades son variables compartidas por el almacén. El funcionamiento de los movimientos es igual que en el automatismo del almacén donde se compara el valor del encoder almacenado en las variables "Cuenta" con el valor que se obtuvo durante la calibración. Finalmente se vuelven a unir para volver a la posición inicial.

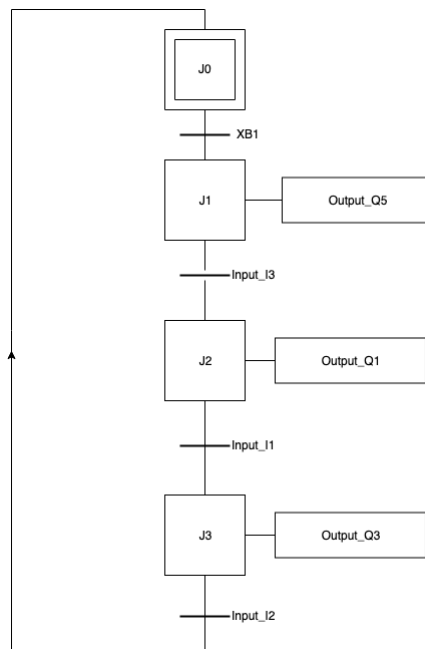
3.2 Grafset compresor y ventosa (V)



Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

Este segundo esquema es similar en estructura al primero y, como se observa, sus transiciones dependen del Grafset de movimientos lo que hace que ambos funcionen a la vez. Se recuerda que esto se creó para que en caso de emergencia donde se tiene que detener el robot, si este tiene una pieza, esta no caiga de golpe, sino que solo cesen los movimientos, congelando el Grafset que los contiene mientras que el compresor y la ventosa permanecen activados a la espera que los movimientos se reanuden o que de manera manual la pieza sea liberada.

3.3 Volver a posición inicial (J)



3.4 Modos de Funcionamiento y Emergencia

Similar a lo que ocurría con el Almacén, no se utilizó forzado o encapsulado, sino que se recurrió a las funciones que ofrecía el programa, en este caso, el Somachine. Las funciones programadas en Ladder son las que aparecen en la Figura 54 y su explicación queda recogida en: Documento Memoria, Apartado 4.3 Programación del Proceso, Subapartado 4.3.2 Somachine - Brazo ventosa.

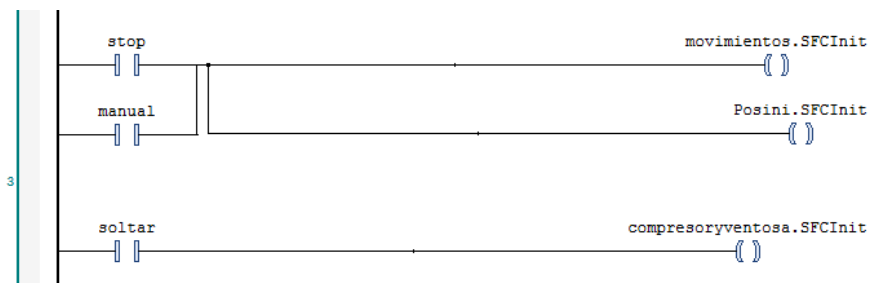


Figura 54. Funciones en Ladder para el control de SFC's

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

En el POU de acciones ocurre también lo mismo que en el otro robot, condicionando los estados y salidas pero aquí, no hace falta negar el modo manual en el desarrollo del modo automático ya que las funciones anteriores cubren esta tarea.

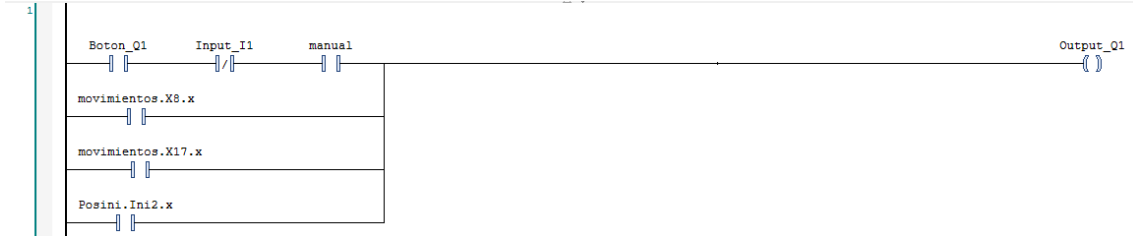


Figura 55. POU de acciones

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



ESCUELA TÉCNICA
SUPERIOR INGENIEROS
INDUSTRIALES VALENCIA

TRABAJO FIN DE GRADO EN INGENIERÍA EN TECNOLOGÍAS INDUSTRIALES

**PROYECTO DE AUTOMATIZACIÓN DE UN
SISTEMA DE ALMACENAMIENTO DE
PIEZAS CON PLC TSX-PREMIUM Y M241 DE
SCHNEIDER Y SCADA MEDIANTE
APLICACIÓN LABVIEW.**

MANUAL DE USUARIO

AUTOR: PABLO GÓMEZ GALLARDO

TUTOR: RAÚL SIMARRO FERNÁNDEZ

Curso Académico: 2019-20

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

1. INTRODUCCIÓN

Este Anexo tiene como objetivo proporcionar al usuario/cliente toda la información que necesita conocer, sin entrar en los detalles del desarrollo del proyecto, para poder poner en funcionamiento el automatismo y una vez en marcha saber cómo utilizar el SCADA y todas las prestaciones que este incluye.

2. PUESTA EN MARCHA

Los programas que se deben arrancar se encuentran configurados con las entradas y salidas de las maquetas FischerTechnik y en caso de querer instalarlos en una industria real habría que configurarlos acorde a la maquinaria real y con las conexiones correspondientes. Pero debido a que eso no se considera competencia de este trabajo se explicará la puesta en marcha que se tiene que realizar con los simuladores y que se asemeja en gran medida a la futura puesta en marcha en una industria.

2.1. Hardware

Los elementos Hardware que se tienen que instalar son los dos autómatas: TSX-Premium y M241 con sus respectivos cables a la red Ethernet y la conexión a los robots mediante un cable D-sub.

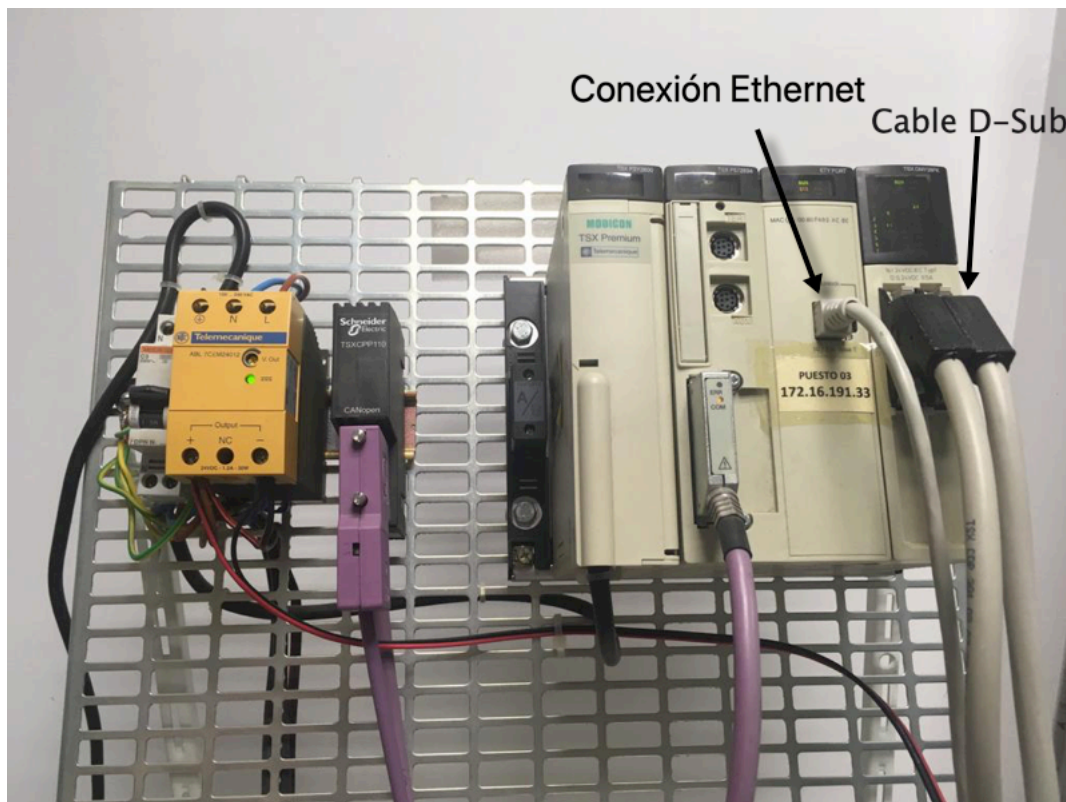


Figura 56. PLC TSX-Premium y sus conexiones

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

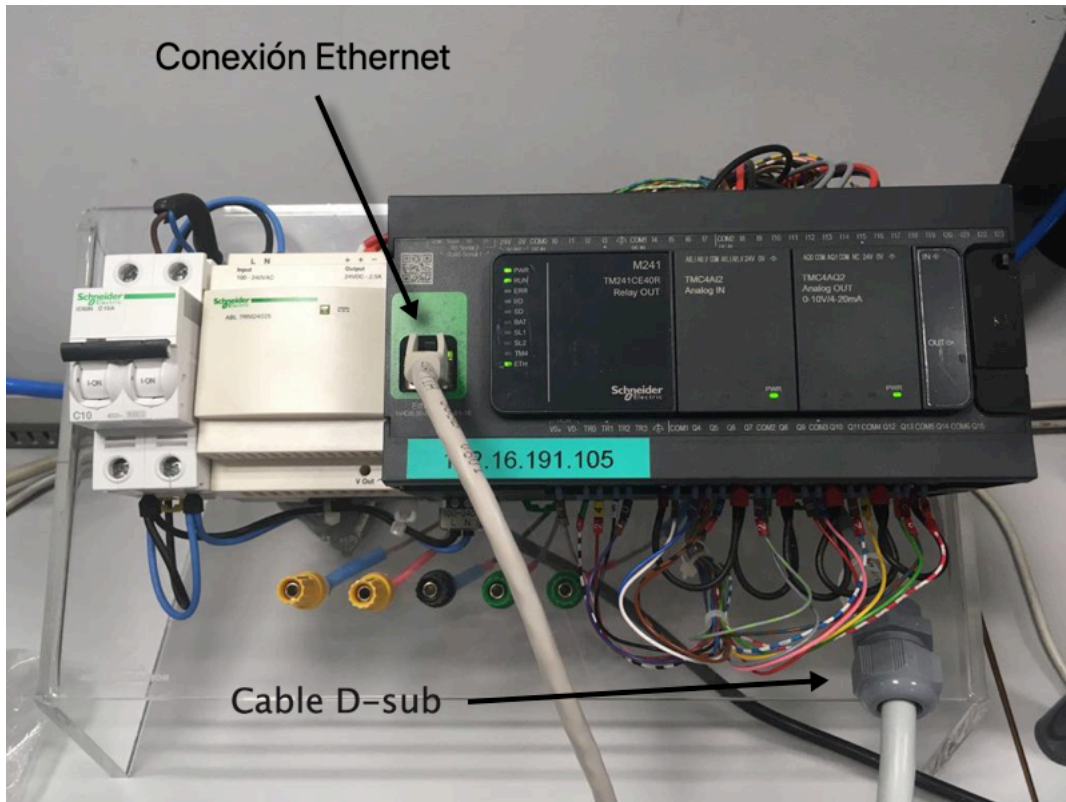


Figura 57. PLC Modicon M241 y sus conexiones

2.2 Software

Todos los archivos del proyecto están contenidos en la carpeta que tiene el nombre de "proyectoTFG" e incluye algunas librerías que se usaron en la programación. Los archivos que el cliente debe utilizar son:

- AlmacenVertical archivo de Unity Pro.
- Brazoventosa archivo de Somachine.
- OpcTFG archivo de KepserverEX.
- ProyectoTFG archivo de LabVIEW.

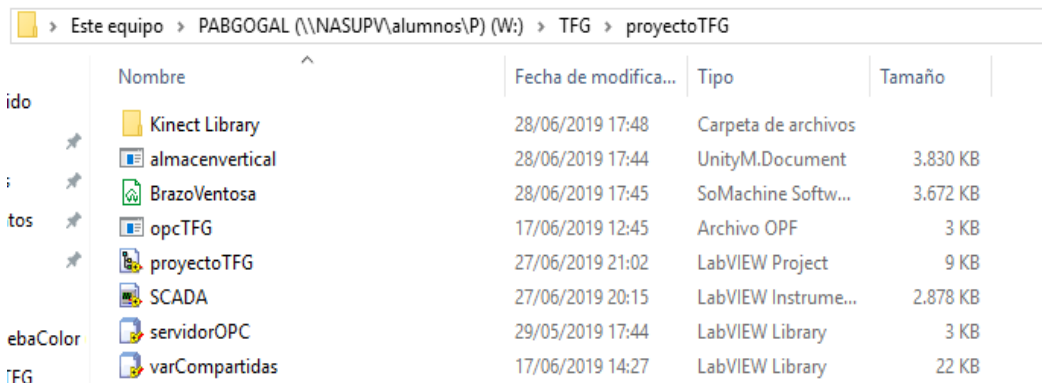


Figura 58 Carpeta ProyectoTFG y sus archivos

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

El orden de apertura que se siguió y con el que se consiguió evitar problemas de comunicación del servidor causados por solapamiento y sobrecarga de datos consistía en: conectar primero el servidor OPC y posteriormente ejecutar todos los programas que deben comunicarse con él, es decir, conectar el centro de las comunicaciones y luego, sus periféricos. Aquí se recoge en realidad la puesta en marcha de un programador que cuenta con todos los programas utilizados, instalados en su ordenador. A un cliente se le pasarían en otro formato o ya instalados en el PLC, aquellos que contienen la automatización.

2.2.1 Servidor OPC- KepServerEX

La única opción que debe configurarse es la dirección IP de los autómatas a los que se conecta el almacén y el brazo. Para ello, se hace doble clic en los dispositivos y desde sus propiedades se cambia el ID como muestra la Figura 59.

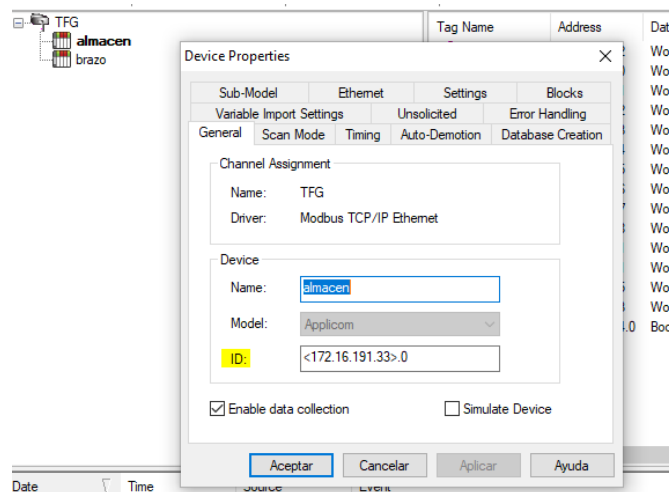


Figura 59 Configuración IP de PLC's

Una vez configuradas las IP se le da a "Runtime" y dentro de este a "Connect" en la barra superior. Posteriormente se activa el "QuickClient" ubicado en la Figura 60. Se cierra la pantalla emergente que aparece sin guardar nada.



Figura 60. Quick Client en KepServerEX

A partir de este momento el servidor ya está conectado a las variables que se comparten y, si se quiere saber si se están compartiendo correctamente se puede abrir otra vez el Quick Client y ver si las variables compartidas aparecen en estado "Good", si se accede a esa dirección, o "Bad", si no lo hace.

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

2.2.2 Almacén Vertical- Unity Pro

Una vez abierto, lo primero que se tiene que hacer es analizar el proyecto que se encuentra en la barra superior, dentro de la carpeta "Generar". Aunque el proyecto se proporciona analizado y sin ningún error, resulta útil conocer esto por si, en el futuro, alguna persona realiza cambios ya que, en ese caso, se debería Analizar. El siguiente paso es señalar la opción de "Regenerar el Proyecto" aunque, como se dice, el proyecto ya está generado. Cuando se hayan completado estos pasos iniciales se pincha en la ventana PLC, "establecer conexión", y se escribe la IP del autómatas, en este caso del TSX que gobierna el almacén.



Figura 61. Configuración IP en Unity Pro

Acto seguido se conecta con la opción "Conectar". Si la conexión se ha realizado correctamente debe aparecer en la parte inferior de la pantalla un recuadro con el mensaje la Figura 62.

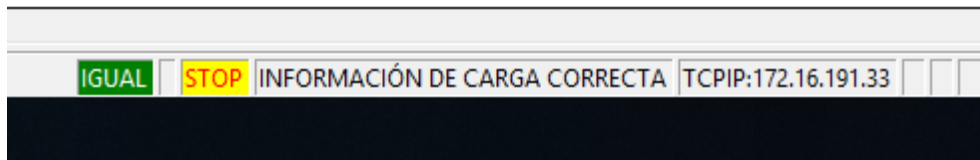


Figura 62. Conexión realizada en Unity Pro

Ahora es turno de transferir el programa al PLC. Esto se consigue con la opción de "Transferir Proyecto a PLC" que se encuentra en la misma carpeta que "Conectar" y "Establecer conexión". Tras una breve carga, el programa ya se encontrará instalado en el PLC. Sólo queda pulsar la tecla "Run".



Figura 63. Botón de RUN

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

Si el mensaje de STOP de la Figura 62 ha cambiado por un recuadro verde con la palabra RUN, quiere decir que el programa ya está ejecutándose.

Para desconectarlo el proceso es a la inversa. A la derecha de "Run" tenemos la opción de "Stop" que detiene la ejecución del programa y, posteriormente, se desconecta desde el mismo panel en que se conectó. Aunque se puede desconectar directamente es muy recomendable que se sigan estos pasos ya que, en ocasiones, el programa almacenado en el PLC sigue ejecutándose, y desde el Unity Pro no se tiene la posibilidad de detenerlo. En ese caso, es necesario apagar el PLC manualmente.

2.2.3 Brazo Ventosa - Somachine

Cuando se abre el programa en Somachine aparece una interfaz con unos módulos conectados por flechas. Se debe ir a la parte superior y elegir la opción "Logic Builder". Entonces, se abrirá una pantalla gris, y una barra a la izquierda con los 3 navegadores que hay: Dispositivos, Aplicaciones y Herramientas.

En la parte de Aplicaciones se encuentran los POU's con toda la programación. La parte que el usuario debe configurar es la sección de Dispositivos. Una vez abierta, dentro del módulo de "MyController", aparecerán las distintas partes y redes que configuran el PLC como muestra la Figura 64.

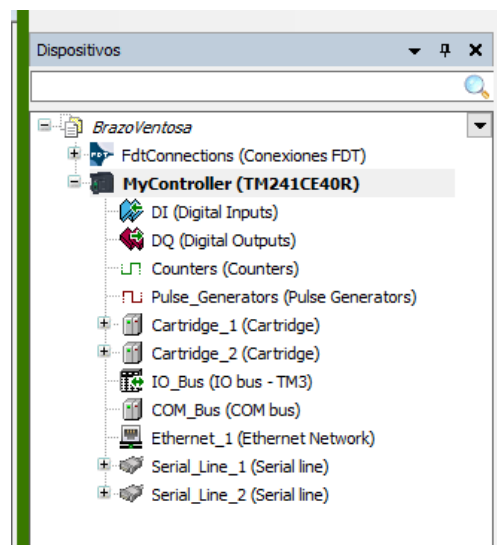


Figura 64. Aspecto del navegador Dispositivos

Las entradas y salidas digitales del robot real se tienen que configurar desde este panel con las casillas de DI (Digital Inputs) y DQ (Digital Outputs). En el Proyecto aparecen configuradas las entradas y salidas de los simuladores que se detallan en el Anexo Definición de Variables. La conexión al autómatas se realiza, primeramente, desde la opción Ethernet_1 (Ethernet Network), donde se tiene que marcar Dirección IP fija y rellenar con los datos del PLC Schneider M241, que gobernará el brazo.

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

En la simulación que se hace en la Universidad se marcaba, sin embargo, la opción Dirección IP de DHCP que rastrea las direcciones de aquellos autómatas que se conectan al Bus de campo de la Universidad. Para acceder a esa búsqueda se selecciona "MyController", y a la derecha aparecerán todos los dispositivos que están conectados. Se busca el dispositivo IP al que se quiere conectar y se comprueba que su conexión sea ETH (Ethernet). Una vez se elige este dispositivo, debe aparecer su dirección en la parte inferior. De no ser así asegurarse de que la Modalidad de conexión sea Dirección IP.

De cualquiera de las maneras, el siguiente paso es elegir la opción "En línea" de la barra superior y de ahí se selecciona el primer elemento, "Iniciar la sesión", que tiene la forma de un engranaje con un punto verde característico.

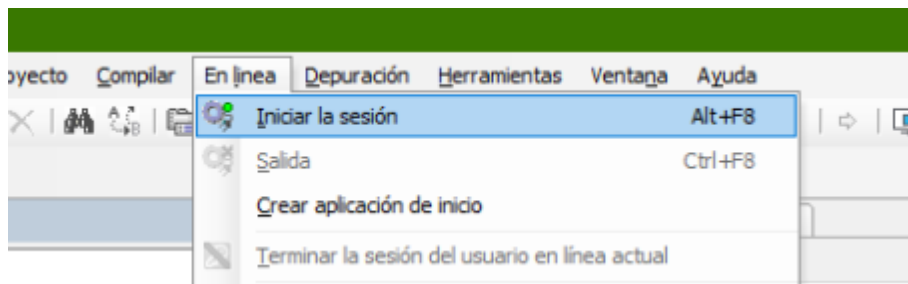


Figura 65 Iniciar sesión en Somachine

Después de esto se abrirán unas advertencias que nos avisarán de un funcionamiento inesperado. No es que haya ningún tipo de error, sino que ocurre siempre. Lo que se ha de hacer es: presionar Alt+F en todas estas advertencias y pulsar que sí o aceptar a cualquier ventana emergente que pudiera aparecer. Por último, se pulsa el triángulo tumbado de inicio que equivale al "Run" de Unity Pro. Para desconectarlo ocurre lo mismo que en el programa anterior, donde se recomienda enormemente que primero se detenga y luego se desconecte para evitar movimientos descontrolados.

2.2.4 LabVIEW

El aspecto que tiene el proyecto de LabVIEW es el de la Figura 66 y lo que el usuario debe hacer es seleccionar el archivo .vi que se llama SCADA. Se abrirá el panel frontal del SCADA que es la interfaz con la que el usuario debe trabajar. Para ponerla en funcionamiento se debe apretar el icono de "Run" que tiene forma de flecha blanca. Inmediatamente después las imágenes de la cámara saldrán por pantalla y ya se puede interactuar con el proceso. Darle al botón de Stop, bastaría para desconectar el SCADA.

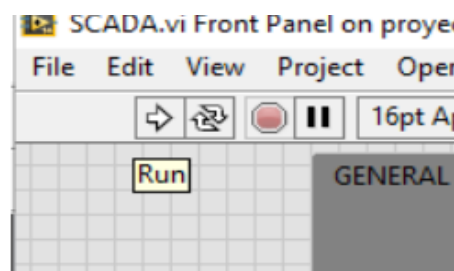


Figura 66. Run en LabVIEW

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

3. FUNCIONAMIENTO DEL SCADA

El SCADA, está compuesto por 2 pestañas: una del funcionamiento general y otra con los controles manuales para cuando se elija ese modo de funcionamiento. Primero de todo se explicarán los elementos de estas dos pantallas y las prestaciones que ofrece en relación a las Especificaciones del Usuario y posteriormente, los protocolos de actuación en situaciones de emergencia.

3.1 Meter Pieza

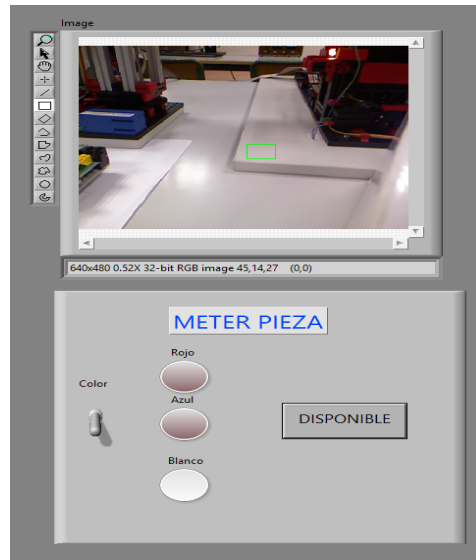


Figura 67 Panel Meter Pieza

Este módulo se encarga del control de la introducción de piezas dentro del almacén. La imagen superior de la Figura 67 capta las imágenes de la cámara de reconocimiento, encargada de distinguir entre los 3 colores de la pieza. El elemento encargado de reconocer el color es el recuadro verde del centro de la pantalla. Por tanto, la pieza debe estar en esa misma posición. Cuando el color es detectado, su led correspondiente se iluminará con el color correspondiente.

En la Figura 68 se recuerda el orden con que se ha seguido para numerar las celdas y que como se verá, está relacionado con los métodos de ordenación.

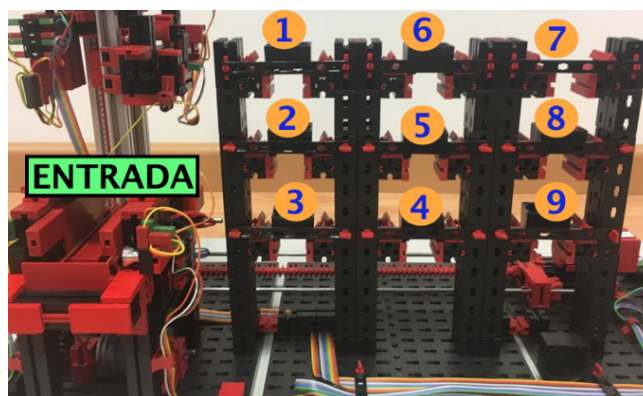


Figura 68 Orden numérico de las celdas

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

Mientras el interruptor Color este activado, las piezas que se depositen en la estructura de entrada serán reconocidas por su color y almacenadas según el siguiente criterio de ordenación.

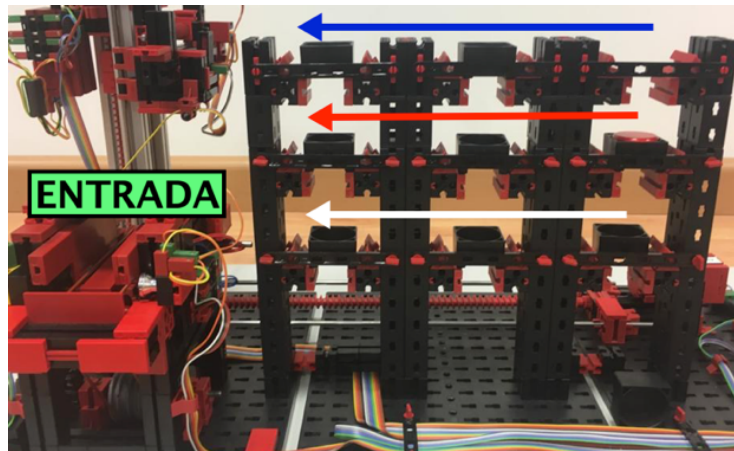


Figura 69 Ordenación por Color

Si una fila que se destina a un color de pieza se completa y se vuelve a introducir una pieza en el depósito de entrada, no se recolocaría en otro lado. Para la recolocación haría falta desactivarlo y presionar el pulsador Disponible. Eso también se puede hacer sin que la fila este completa es decir en cualquier momento pasando de un modo de ordenación por Color a uno por celdas disponibles. Este criterio va rellenando el almacén de la manera en que se refleja en la Figura 70. Esto quiere decir que si por este orden tocaría meter en la celda 4 y, sin embargo, la celda número 2 se queda libre se daría prioridad, como indica la flecha verde, a la número 2.

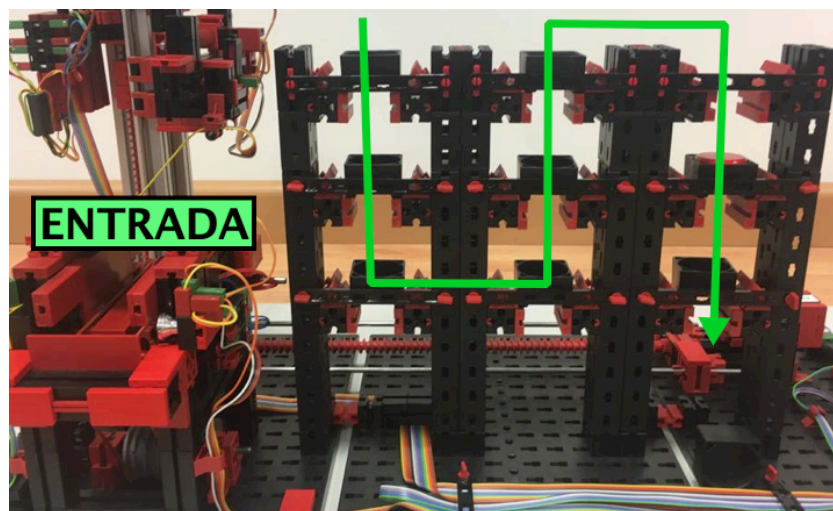


Figura 70 Ordenación por celda Disponible

Combinando estos 2 métodos la idea es que se puedan ordenar las piezas por su tipología, en este caso por su color, sin que esto limite la capacidad del almacén en caso de que filas se completen o se quiera almacenar en las posiciones más cercanas a aquellos productos que por su urgencia a la hora de distribuir necesiten colocarse los primeros.

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

3.2 Sacar Pieza

El modulo del SCADA encargado del control de la extracción de las piezas es el que muestra la Figura 71.

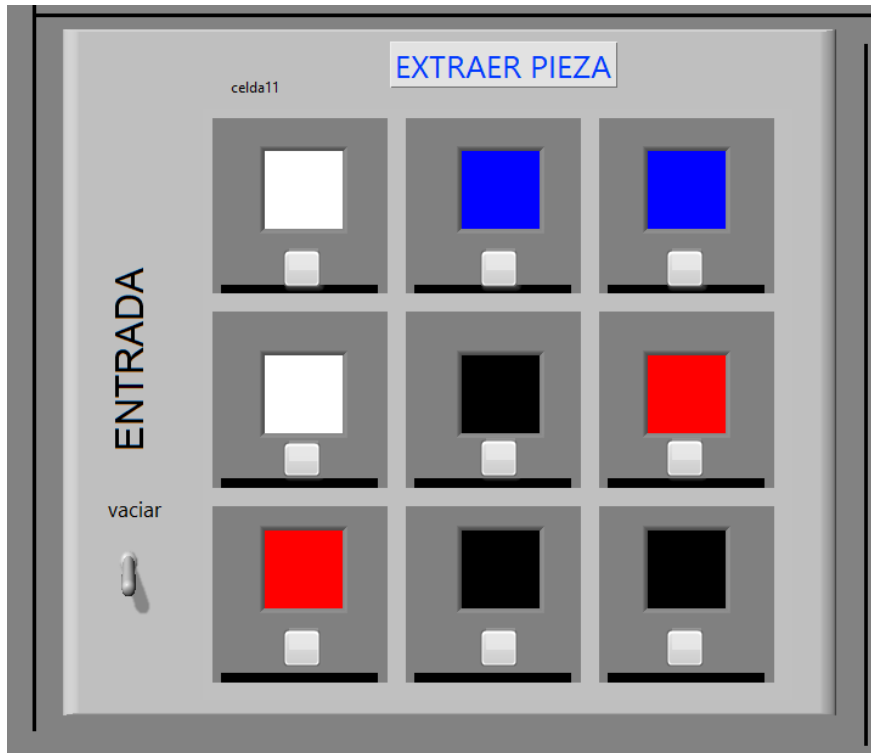


Figura 71. Aspecto del panel para Extraer Pieza.

Se observa un esquema gráfico del almacén con la referencia de la entrada como en las figuras empleadas para explicar cómo se meten las piezas. Como se observa en cada celda hay un cuadrado que representará el color de la pieza que se haya dentro y de color negro para las posiciones vacías. Debajo de cada indicador se encuentra un botón que nos permite extraer piezas siempre que se encuentren ocupadas por alguna posición, sino no pasará nada

El interruptor de vaciar realiza un vaciado automático del almacén (mientras esté activado) y sigue un orden de prioridad igual que el de la Figura 70 para meter piezas por posición disponible. De manera que sea cual sea el estado del almacén si se deja activado, el almacén se vaciará por completo. Como información extra, el elevador encargado de coger las piezas solo irá a aquellas posiciones que están ocupadas por una pieza.

3.3 Estados del almacén



Figura 72. Indicadores del almacén.

Este módulo nos permite conocer los estados generales del almacén y el tipo de proceso que se está llevando a cabo, si meter o sacar pieza. Con esta información, el usuario, puede conocer de manera clara y sencilla como está trabajando en ese momento el automatismo.

3.4 Modos de funcionamiento y parada de emergencia



Figura 73 Módulo de emergencia y de modos del SCADA

De izquierda a derecha, el primer elemento que aparece es la ruleta de modos. Esta ruleta tiene su posición inicial en Paro por tanto el arranque se realiza desde Paro hacia el modo en que se quiera trabajar. Para realizar todas aquellas funciones descritas en los 2 subapartados anteriores de Meter y Sacar pieza es necesario estar en modo automático. Una vez en el modo automático, si se quiere parar de procesar piezas, bastaría con volver a la posición de paro y solo cuando acabase de colocar la pieza que se esté manipulando se detendría el almacén y el brazo en la posición inicial.

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

Si por el contrario no se estuvieran manipulando piezas, los elementos quedarían donde están sin la posibilidad de meter o sacar pieza hasta que se volviese a automático. En caso de querer pasar de automático a manual, habría que pasar por paro, pero a diferencia que este, sí que se detendría instantáneamente el almacén, sin esperar a que la última pieza se resuelva. Con el control puesto en manual se debe cambiar de pestaña a manual.

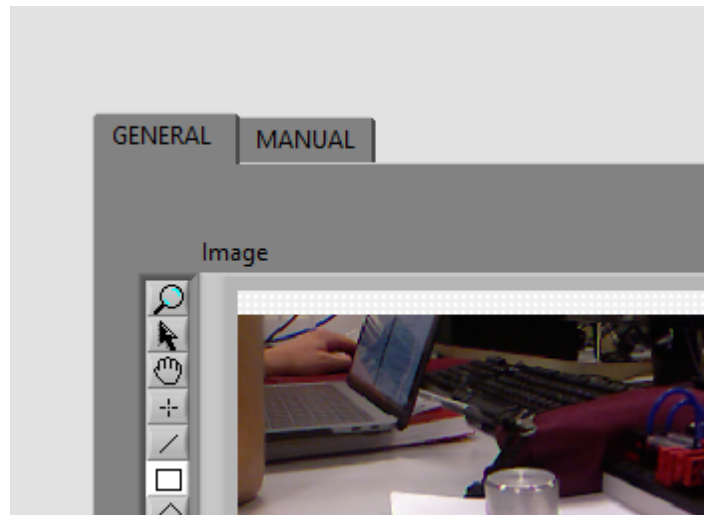


Figura 74 Pestañas en el SCADA

En esta pestaña se proyecta la interfaz gráfica de la Figura 75, la cual cuenta con todas las salidas del almacén y del brazo que se detallan en el Anexo Definición de Variables. Además, cuenta con un botón de soltar que se explicó su utilidad en el Apartado 4.2.2 de la Memoria, Somachine-Brazo ventosa.



Figura 75 Control manual en el SCADA.

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

En caso de emergencia (fallo de los robots, fallo de la caja/pieza, situación de peligro para persona, máquina o producto etc.) se distingue entre la parada del almacén o la del brazo. Esta distinción ocurre debido a que el tratamiento de la parada del brazo es diferente. La razón de esto queda explicada en el Apartado 4.2.2 de la Memoria, Somachine-Brazo ventosa, y en el Apartado 1.4 de la Memoria, Especificaciones del Usuario.

En el caso del almacén, pulsar el botón de Stop detiene de manera instantánea el proceso y activa la iluminación de emergencia. Una vez se desactive la seta de emergencia, la luz se desactivará, pero para volver al proceso automático es necesario pasar por el modo manual para retirar el contenido que se transporta (solo si en ese momento se transporta). El modo manual, solo funciona si esta desactivado el Stop. Esta retirada, es necesaria porque cuando se vuelva a el modo automático, el proceso se reiniciará, por lo tanto, si en el momento de la parada, el transelevador transportaba una caja, en su reinicio entraría en conflicto con las paredes del almacén (se producen choques). En ese caso hay que resolver la caja y su pieza utilizando el modo manual para colocar esa caja en un sitio que permita la reanudación del proceso sin ningún tipo de obstáculo.

Para la explicación de estas soluciones se introducirán una serie de conceptos respecto al recorrido de las piezas. En la acción de meter, el almacén primeramente va a la celda, coge la caja vacía y la transporta a la cinta donde esperara en la entrada. Después la pieza es introducida y la caja llena devuelta al elevador que la dejará de nuevo en la celda. En esta descripción, se han observado tres etapas: 1ª parte, cinta y 2ª parte. En la salida de piezas ocurre lo mismo con la diferencia que en la 1ª parte la caja contiene una pieza y en la 2ª se encuentra vacía. Las soluciones dependen de en que parte se encuentre como indica la Tabla 3.

estado	tramo	Protocolo de actuación manual (una vez terminado, volver al modo automático)
metiendo	1er tramo	Llevar la caja vacía a la entrada
	cinta	Volver al modo automático
	2º tramo	Llevar la caja con la pieza a la entrada y, pulsar el botón de sacar en la celda que se estaba metiendo. Cuando la pieza llegue de nuevo al depósito de salida llevarla al contenedor de entrada y volver a elegir la opción de meter pieza previamente elegida.
sacando	1er tramo	Llevar la caja con la pieza a la entrada
	cinta	Volver al modo automático
	2º tramo	Llevar la caja vacía con precisión a la celda a la que se dirijía. Esta posición es aquella celda que no contiene caja

Tabla 9. Protocolos del paso por manual en una emergencia. Fuente: elaboración propia

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

Otra opción que se deja como elección del usuario, es la de retirar la caja directamente con la ayuda de un operario. Aunque en el simulador esto es fácil ya que con las manos se retira la caja y desaparece el problema, se desconoce la situación del almacén real y de si esto es posible pero una opción es que, desde el modo manual, el elevador, descienda a la altura del suelo y de ahí el operario, retire la caja y posteriormente se coloque en la entrada. En ese caso, no haría falta realizar todo lo que se indica en la Tabla 3. Otra opción es automatizar estos protocolos de emergencia sin tener que hacerlos de manera manual pero que, por falta de tiempo, aquí no se ha podido realizar.

En el caso del brazo, activar le seta, activa la luz de emergencia . Se detiene el proceso de manera inmediata. La particularidad del brazo reside en que si lleva cogida una pieza, al detenerse, el compresor y la ventosa permanecen activos y la pieza no cae, sigue succionada. Aquí el usuario elige si pasar por el modo manual o permanecer en el automático. Para utilizar cualquiera de estos 2 modos, es necesario que previamente se desactiva la emergencia. Si se elige permanecer en modo automático, el brazo que estaba detenido se reiniciara su proceso sin que entre en conflicto sus movimientos, con la pieza sujeta en todo momento. Si por el contrario se detecta que la pieza tiene algún defecto desde la parada se cambiaría a modo manual donde se llevaría el brazo a la posición deseada y una vez ahí, se presiona el botón de soltar que libera la pieza.

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



ESCUELA TÉCNICA
SUPERIOR INGENIEROS
INDUSTRIALES VALENCIA

TRABAJO FIN DE GRADO EN INGENIERÍA EN TECNOLOGÍAS INDUSTRIALES

**PROYECTO DE AUTOMATIZACIÓN DE UN
SISTEMA DE ALMACENAMIENTO DE
PIEZAS CON PLC TSX-PREMIUM Y M241 DE
SCHNEIDER Y SCADA MEDIANTE
APLICACIÓN LABVIEW.**

PRESUPUESTO

AUTOR: PABLO GÓMEZ GALLARDO

TUTOR: RAÚL SIMARRO FERNÁNDEZ

Curso Académico: 2019-20

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

1. INTRODUCCIÓN:

En el cálculo del presupuesto, no se tendrán en cuenta los costes de la maquinaria empleada como son el brazo y el almacén ya que se asume que el cliente, cuenta ya con ellos. Sí que se tendrán en cuenta por el contrario los gastos de la fase de diseño, implementación y adquisición y amortización del material.

2. CUADRO Nº1: MANO DE OBRA

Aunque el proyecto, llevado a un caso real, se realizaría por varios Ingenieros de manera que el proceso no fuera secuencial, sino que se ahorrara tiempo realizando ciertas tareas de manera simultánea, al ser estos actores, graduados en la titulación Ingeniería Industrial con el mismo salario, se simplificarán las tablas con el trabajo de un solo ingeniero que realiza las horas totales del proyecto.

Código	Unidad	Descripción	Salario (€/h)
MO01	h	Ingeniero Industrial	18

Tabla 10 Cuadro mano de obra.

3. CUADRO Nº2: PRECIO DE LOS MATERIALES

Hardware

En el cálculo del precio unitario de los ordenadores, se ha considerado que la vida útil de ambos es de 5 años. El portátil se ha empleado para la búsqueda de información sobre todo el análisis de alternativas y en la redacción del proyecto y se estima su uso medio anual en 3 horas diarias.

$$\text{Amortización portátil} = \frac{\text{base amortización(€)}}{\text{vida útil (h)}} = \frac{1200}{5 \cdot 365 \cdot 3} = 0.22$$

El ordenador de sobremesa necesario para la programación de los autómatas, servidor OPC y SCADA tiene un precio estimado de 900 € con un uso medio de 8 horas diarias que representa una jornada laboral, durante 225 días laborales al año.

$$\text{Amortización sobremesa} = \frac{\text{base amortización(€)}}{\text{vida útil (h)}} = \frac{900}{5 \cdot 225 \cdot 8} = 0.1$$

Los 2 autómatas también se incluyen en este concepto junto con los cables necesarios (cable Ethernet y D-sub) que aparecen contenidos dentro del precio de estos.

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

Código	Unidad	Descripción	Precio Unitario(€/h)	Rendimiento	Importe(€)
MH01	ud	TSX-Premium P-57 de Schneider	1.400	1	1.400
MH02	ud	Modicon M241 de Schneider	280	1	280
MH03	h	Apple MacBook Pro	0,22	80	17,6
MH04	h	Ordenador de sobremesa	0,1	184	18,4
				TOTAL	1.716

Tabla 11 Cuadro material Hardware

Software

Se considera que el ingeniero proporciona al cliente los programas instalados ya en el PLC o en un formato ejecutable de manera que el cliente puede hacer uso de ellos sin necesidad de tenerlos instalados en su industria. Esto último no se aplica al servidor OPC en KEPServerEX ya que se necesita contar con el programa instalado para que funcione el servidor. Salvo esta excepción, los programas se consideran costes de amortización. Somachine es un software libre que puede instalarse gratuitamente por tanto su coste es nulo. El uso que se ha hecho de estos programas se asume en jornadas laborales de 8 horas y 225 días laborales al año. Esto nos da un coeficiente de 1800 horas/anales que dividido por el coste de la licencia anual, resulta en el coeficiente de amortización o precio unitario.

Software	Precio anual licencia (€)
Unity Pro	1.184
Labview	399
KEPServerEX	397,44
Microsoft Office 365	69

Tabla 12. Programas y precio de licencia anual.

Código	Ud	Descripción	Precio unitario (€/h)	Rendimiento	Importe(€)
MS01	h	Unity Pro	0,66	80	52,8
MS02	h	Labview	0,22	40	8.8
MS03	ud	KEPServerEX	397,44	1	397,44
MS04	h	Microsoft Office 365	0,04	50	2
MS05	h	Somachine	0	56	0
				TOTAL	461,04

Tabla 13. Cuadro material Software

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

Total

El total en concepto de material, el importe asciende a 2.177,04 €.

Material	Importe(€)
Hardware	1.716
Software	461,04
TOTAL	2.177,04

Tabla 14. Precio total de la ejecución material

4. CUADRO N°3: PRECIOS UNITARIOS

Unidad de Obra	Descripción	Importe (€)
1	Estudio de alternativas	301,18
2	Diseño del automatismo	600,87
3	Programación del automatismo	2.612,25
4	Desarrollo e implementación del control	3.073,24
5	Elaboración del documento del Proyecto	1.063,21
TOTAL		7.650,75

Tabla 15 Presupuesto Precios Unitarios

5. CUADRO N°4: PRECIOS DESCOMPUESTOS

Estudio de alternativas

Código	Unidad	Descripción	Rendimiento	Precio unitario	Importe (€)
MO01	h	Graduado en Ingeniería Industrial	16	18	288
MH03	h	Apple MacBook Pro	16	0,22	3,52
	%	Costes directos complementarios	4		11,66
TOTAL					303,18

Tabla 16 Presupuesto descompuesto N°1

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

Diseño del automatismo

Código	Unidad	Descripción	Rendimiento	Precio unitario	Importe (€)
MO01	h	Graduado en Ingeniería Industrial	32	18	576
MH03	h	Apple MacBook Pro	8	0,22	1,76
	%	Costes directos complementarios	4		23,11
TOTAL					600,87

Tabla 17. Presupuesto descompuesto N°2.

Programación del automatismo

Código	Unidad	Descripción	Rendimiento	Precio unitario	Importe (€)
MO01	h	Graduado en Ingeniería Industrial	136	18	2.448
MH04	h	Ordenador de sobremesa	136	0,1	13.6
MS01	h	Unity Pro	80	0,66	52,8
MS05	h	Somachine	56	0	0
	%	Costes Directos complementarios	4		100,57
TOTAL					2.612,25

Tabla 18. Presupuesto descompuesto N°3.

Desarrollo e implementación del control

Código	Unidad	Descripción	Rendimiento	Precio unitario	Importe (€)
MO01	h	Graduado en Ingeniería Industrial	48	18	864
MH04	h	Ordenador de sobremesa	48	0,1	4,8
MS02	h	Labview	40	0,22	8,8
MS03	ud	KEPServerEX	397,44	1	397.44
MH01	ud	TSX-Premium P-57 de Schneider	1	1.400	1.400
MH02	ud	Modicon M241 de Schneider	1	280	280
	%	Costes Directos complementarios	4		118,20
TOTAL					3.073,24

Tabla 19. Presupuesto descompuesto N°4.

Proyecto de automatización de un sistema de almacenamiento de piezas con PLC TSX-Premium y M241 de Schneider y SCADA mediante aplicación LabVIEW.

Elaboración del documento del Proyecto

Código	Unidad	Descripción	Rendimiento	Precio unitario	Importe (€)
MO01	h	Graduado en Ingeniería Industrial	56	18	1008
MHO3	h	Apple MacBook Pro	56	0,22	12,32
MS04	h	Microsoft Office 365	50	0,04	2
	%	Costes directos complementarios	4		40,89
TOTAL					1.063,21

Tabla 20. Presupuesto descompuesto N°5.

6. PRESUPUESTO DE EJECUCIÓN Y POR CONTRATA

Presupuesto de Ejecución Material	7.650,75
13% Gastos Generales	994,59
6% Beneficio Industrial	459,05
Presupuesto de Ejecución por Contrata	9.104,39
21% IVA	1.911,92
PRESUPUESTO BASE DE LICITACIÓN	11.016,31

Tabla 21. Resumen del presupuesto

