



Universidad de Valladolid



**ESCUELA DE INGENIERÍAS
INDUSTRIALES**

UNIVERSIDAD DE VALLADOLID

ESCUELA DE INGENIERIAS INDUSTRIALES

GRADO EN INGENIERÍA ELECTRÓNICA INDUSTRIAL Y AUTOMÁTICA

Estudio, diseño e integración de una estación industrial colaborativa

Autor:

Juan Mínguez Moretón

Tutor:

Alfonso Valentín Poncela Méndez

Valladolid, septiembre, 2020.





RESUMEN

En la actualidad, la industria está en continua evolución. La industria 4.0 supone un avance en las metodologías utilizadas en los procesos industriales hasta puntos que hace unos años serían tomados como mitos. Con la llegada del tratamiento de datos masivos (Big data), el internet de las cosas (IoT) y la robótica colaborativa se podrá llegar a desarrollar “fábricas inteligentes” con una gran capacidad de adaptarse según las necesidades y una mejor gestión de los recursos.

En el presente proyecto se hablará del desarrollo de una estación colaborativa para instalación de cubiertas plásticas en el bajo de los coches controlada por un autómatas programable. La robótica colaborativa está en auge. Es un hecho que la colaboración entre hombre y máquina en un proceso industrial logra una mayor flexibilidad a la hora de adaptarse a las necesidades y supone una mejora en términos de productividad.

Palabras clave: industria 4.0, fábricas inteligentes, robótica colaborativa, cubiertas plásticas, autómatas programable.

ABSTRACT

Nowadays, industry is constantly evolving. Industry 4.0 involves a breakthrough in current industrial processes methodologies that years ago would be taken as myths. With the advent of Big data, Internet of Things and collaborative robotics we'll be able to develop “Smart factories” with enhanced adaptative capabilities to requirements and better management of resources.

This document will discuss the development of a collaborative industrial station for the installation of plastic covers at the underside of cars controlled by automaton. Cobots are on the rise. It is a fact that the cooperation between man and machine achieves an improvement in productive capacity and more leeway.

Keywords: industry 4.0, smart factories, collaborative robotics, plastic covers, automaton.





ÍNDICE

CAPÍTULO 1: INTRODUCCIÓN Y OBJETIVOS	11
1.1.- Introducción: INDUSTRIA 4.0.....	12
1.2.-Descripción del proyecto.....	13
1.3.- Funcionamiento en automático.....	15
1.4.- Justificación	20
1.5.-Búsqueda bibliográfica dentro de la UVa.....	20
CAPÍTULO 2: DESARROLLO DEL PROYECTO	21
2.1.- Introducción	22
2.2.- Estado del arte.....	22
2.2.1.- Investigaciones internacionales	22
2.2.2.- Investigaciones nacionales.....	23
2.3.- Descripción de los elementos empleados.....	23
2.3.1.- Robot colaborativo	23
2.3.2.- Sistema de control.....	24
2.3.3.- Elementos de seguridad	24
2.4.- Estudio de los elementos	25
2.4.1.- Estudio de valores mecánicos del robot colaborativo.....	25
2.4.2.-Estudio de la zona de trabajo del robot colaborativo	26
2.4.3.- Estudio de valores de agarre de la pinza colaborativa	27
2.4.4.- Diseño de esquemas neumáticos para accionamiento de la pinza.....	27
2.4.5.- Diseño de esquemas eléctricos de la estación.....	28
2.4.6.- Estudio de puntos de agarre sobre cubiertas plásticas	28
2.4.7.- Estudio de elementos de seguridad necesarios para asegurar la categoría 4.....	29
2.5.-Conexión de los elementos	29
2.6.- Simulación de esquemas neumáticos	31
2.7.- Programación y simulación del programa de control del PLC.....	35
2.7.1.- Descripción de red Hardware	35
2.7.2.-Diseño del programa de control del PLC con TIA Portal.....	39
2.7.3.- Simulación del programa del autómatas	62
2.8.- Programación y simulación de la rutina del robot.....	71
2.9.- Estudio económico.....	81
CAPÍTULO 3: CONCLUSIONES	85
3.1.- Conclusiones etapa de estudio	86



3.2.-Conclusiones etapa de desarrollo	86
3.3.- Conclusiones generales	88
CAPÍTULO 4: LÍNEAS FUTURAS	89
4.1.- Líneas futuras a corto plazo	90
4.2.- Líneas futuras a medio y largo plazo	90
BIBLIOGRAFÍA/WEBGRAFÍA.....	91
ANEXO I: PLANOS	95
ANEXO II: COMUNICACIÓN PROFINET DEL CONTROLADOR: ESTRUCTURA DE DATOS	103
ANEXO III: COMPONENTES DE LA ESTACIÓN	109
ANEXO IV: HARDWARE DEL SISTEMA DE CONTROL DE LA ESTACIÓN	117
ANEXO V: CÓDIGO DE POLYSCOPE: RUTINA DEL ROBOT	123
ANEXO VI: PANEL DE CONTROL DEL USUARIO	131
ANEXO VII: MANUAL DEL USUARIO.....	133



INDICE DE FIGURAS

Figura 1: Desarrollo de las revoluciones industriales [1]	12
Figura 2: Colaboracion hombre-máquina [5]	13
Figura 3: Cubierta plástica (Modelado aproximado)	13
Figura 4: Diagrama de alcance del robot UR10 CB Series [15]	26
Figura 5: Esquema neumático de control de funcionamiento de la pinza adaptativa	27
Figura 6: Esquema eléctrico	28
Figura 7: Zona de agarre de la cubierta plástica	29
Figura 8: Conexionado de los elementos	30
Figura 9: Entorno de diseño de esquemas de FluidSim	31
Figura 10: Esquema neumático definido.....	32
Figura 11: Vástago desplegado.....	33
Figura 12: Vástago recogido	34
Figura 13: Posición fija del vástago.....	34
Figura 14: Ruta para importar el archivo GSDML	36
Figura 15: Menú Administrar archivos de descripción de dispositivos.....	36
Figura 16: Esquema de configuración del hardware de la estación en TIA Portal.....	37
Figura 17: ruta para importar el archivo .udt	38
Figura 18: Tabla variables FB2	42
Figura 19: Segmento 1 de FB2	42
Figura 20: Segmento 2 de FB2	42
Figura 21: Segmento 3 de FB2	43
Figura 22: Segmento 4 de FB2	43
Figura 23: Tabla de variables de FB3.....	43
Figura 24: Segmento 1 de FB3	44
Figura 25: Segmento 2 de FB3	44
Figura 26: Segmento 3 de FB3	44
Figura 27: Segmento 4 de FB3	45
Figura 28: Tabla de variables de la FB4	45
Figura 29: Segmento 1 de FB4	46
Figura 30: Segmento 2 de FB4	46
Figura 31: Segmento 3 de FB4	47
Figura 32: Tabla de variables de la FB5	47
Figura 33: Segmento 1 de FB5	48
Figura 34: Segmento 2 de FB5	48
Figura 35: Tabla de variables de FB50.....	49
Figura 36: Segmento 1 de FB50	50
Figura 37: Segmento 2 de FB50	50
Figura 38: Segmento 3 de FB50	50
Figura 39: Segmento 4 de FB50	51
Figura 40: Segmento 5 de FB50	51
Figura 41: Segmento 6 de FB50	52
Figura 42: Tabla de variables de FB100.....	52
Figura 43: Segmento 1 de FB100	53
Figura 44: Segmento 2 de FB100	53



Figura 45: Segmento 3 de FB100	54
Figura 46: Segmento 4 de FB100	54
Figura 47: Segmento 5 de FB100	54
Figura 48: Segmento 6 de FB100	55
Figura 49: Segmento 7 de FB100	55
Figura 50: Tabla de variables de la FB150	55
Figura 51: Segmento 1 de FB150	56
Figura 52: Segmento 2 de FB150	56
Figura 53: Segmento 1 de la OB1.....	57
Figura 54: Segmento 2 de la OB1.....	57
Figura 55: Segmento 3 de la OB1.....	58
Figura 56: Segmento 4 de la OB1.....	59
Figura 57: Segmento 5 de la OB1.....	60
Figura 58: Segmento 6 de la OB1.....	61
Figura 59: Segmento 7 de la OB1.....	62
Figura 60: Menú principal de la aplicación.....	63
Figura 61: Entorno de programación de TIA Portal	63
Figura 62: Vista general de la estación a simular.....	64
Figura 63: Panel de control.....	65
Figura 64: Menu principal de TIA Portal. Selección Siemens.	66
Figura 65: Selección de autómatas en la página web de Factory IO	67
Figura 66: Página de selección de plantilla [26].....	67
Figura 67: Función de comunicación con el Factory IO.....	68
Figura 68: Menú de inicio de simulación.....	68
Figura 69: Menú Driver de Factory IO	69
Figura 70: Menú Configuration de Factory IO	69
Figura 71: Menú driver con las variables asignadas y el simulador corriendo + ventana de PLCSIM	70
Figura 72: Simulación de la estación en funcionamiento.....	70
Figura 73: Menú principal del PolyScope	72
Figura 74: Menú habilitar PROFINET.....	73
Figura 75: PROFINET habilitado y activo	74
Figura 76: Menú de inicialización del robot	75
Figura 77: Robot en estado ocioso.....	75
Figura 78: Menú “New Program”	76
Figura 79: Entorno de programación de PolyScope.....	77
Figura 80: Segmento 1 del programa	78
Figura 81: Segmento 2 del programa.....	78
Figura 82: Segmento 3 del programa.....	79
Figura 83: Segmento 4 del programa.....	80
Figura 84: Segmento 5 del programa.....	80
Figura 85: Tabla de variables definidas por sus siglas [28].....	103
Figura 86: Estructura de la variable tipo “UR_T20”	104
Figura 87: Estructura de datos T20 bits 0 a 255 [28].....	105
Figura 88: Estructura de datos T20 bits 256-799 [28]	105
Figura 89: Estructura de datos T20 bits 800-1631 [28]	105



Figura 90: Estructura de datos T2O bit 1632-2239 [28]	105
Figura 91: Estructura de datos T2O bits 2240-2303 [28]	106
Figura 92: Estructura de datos T2O bits 2304-3071 [28]	106
Figura 93: Estructura de datos T2O bits 3072-3839 [28]	106
Figura 94: Estructura de datos O2T bits 0-191 [28]	106
Figura 95: Estructura de datos O2T bits 192-991 [28]	107
Figura 96: Estructura de datos O2T bits 992-1791 [28]	107
Figura 97: Robot colaborativo UR10 [29]	109
Figura 98: PLC Siemens S7-1516F [30]	110
Figura 99 : Columna de señalización [32].....	112
Figura 100: sensor de presencia [33]	113
Figura 101: Similitud de la pinza con lengua de un camaleón [16].....	113
Figura 102: Ejemplo de soft gripper [34].....	113
Figura 103: Pinza adaptativa [16]	114
Figura 104: Seta de emergencia [35]	115
Figura 105: Módulo ET200SP [36].....	116
Figura 106: Sistema de control principal.....	117
Figura 107: Periferia descentralizada.....	118
Figura 108 : TeachPendant de Universal Robors [13].....	119
Figura 109: Armario de controladora del UR10 [38].....	120
Figura 110: Diagrama de los puertos E/S del controlador [15]	120
Figura 111: Programa de Polyscope.....	123
Figura 112: Configuración punto de rebuclaje	124
Figura 113: Configuración punto de aproximación a P10	125
Figura 114: Configuración punto P10.....	126
Figura 115: Configuración punto de paso P10-P20.....	127
Figura 116: Configuración punto de aproximación a P20	128
Figura 117: Configuración punto P20.....	129
Figura 118: Pupitre de control, vista 1	131
Figura 119: Pupitre de contro, vista 2	131
Figura 120: Pupitre de control, vista de panel de botonera.....	132
Figura 121: Cargar programa en dispositivo TIA Portal	133



INDICE DE TABLAS

Tabla 1: Variables del programa	40
Tabla 2: Registros E/S utilizados para la comunicación Robot-Autómata	59
Tabla 3: Precio del sistema principal de seguridad.....	81
Tabla 4 Precio del sistema de periferia descentralizada.....	81
Tabla 5: Precio de Robot +pedestal + pupitre	81
Tabla 6: Precios de las herramientas de programación.....	82
Tabla 7: Precio de elementos de seguridad	82
Tabla 8: Precio de los elementos neumáticos	82
Tabla 9: Precio de los elementos eléctricos	82
Tabla 10: Precio de los procedimientos de implementación del prototipo.....	82
Tabla 11: Sumatorio de los subtotales para obtener el precio final.....	83



CAPÍTULO 1: INTRODUCCIÓN Y OBJETIVOS



1.1.- Introducción: INDUSTRIA 4.0

La cuarta revolución industrial, también conocida como industria 4.0 [1], está cambiando la forma en que los negocios operan y, por lo tanto, los entornos en los que se ven obligados a competir.

La promesa de esta cuarta revolución industrial consiste en la combinación de técnicas avanzadas de producción y de operaciones con tecnologías inteligentes que se integrarán en las organizaciones, las personas y los activos.

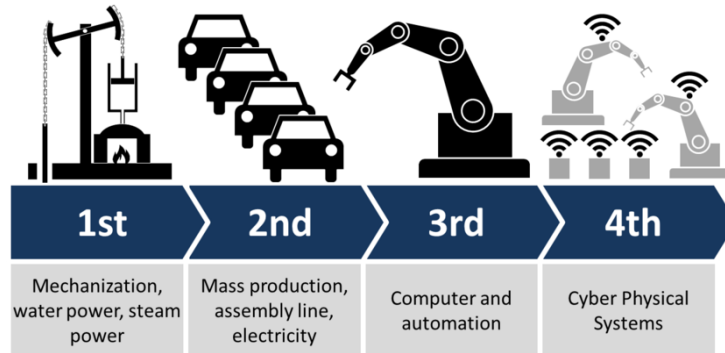


Figura 1: Desarrollo de las revoluciones industriales [1]

La inteligencia artificial está señalada como el elemento central de esta transformación debido al auge del tratamiento masivo de datos (“big data”), el uso de algoritmos para procesarlos y la interconexión masiva de los sistemas y dispositivos digitales [2]. No obstante, esta revolución está también marcada por la aparición de nuevas tecnologías como la robótica colaborativa [3], las tecnologías cognitivas, la nanotecnología y el Internet de las cosas (“IoT”), entre otros.

Las organizaciones deben identificar las tecnologías que mejor satisfacen sus necesidades para invertir en ellas. Si estas organizaciones no son capaces de comprender los cambios y oportunidades que trae consigo esta Industria 4.0, corren el riesgo de perder cuota de mercado.

Al ser la cuarta revolución, antes ha debido haber otras tres grandes revoluciones (Figura 1) [4].

- La primera revolución industrial tuvo lugar en el siglo XVIII. Con esta revolución se sustituyó el trabajo físico, ya fuera del hombre o del animal, por máquinas movidas por el vapor.
- La segunda revolución industrial, aproximadamente un siglo después de la primera. Su mayor representante es Henry Ford. Con esta segunda revolución se hizo accesible el consumo y producción en masa. Surgió el concepto “línea de montaje”.
- La tercera revolución industrial, se desarrolló a mediados del siglo XX. Se empezó a incluir las computadoras y los PLC (Programmable Logic Controller) para control de los procesos industriales, así como el uso de robots y sensores de forma masiva.

Con la aparición masiva de robots para las tareas de producción, era necesario implementar sofisticados sistemas de protección: numerosas setas de emergencia en diferentes lugares de la célula, vallado de las estaciones, cerrojos de seguridad, sensores de presencia, etcétera. Estos robots no son capaces de identificar colisiones con trabajadores y pueden llegar a causar

accidentes muy graves. Por ello, deben trabajar de forma aislada dentro de células industriales.

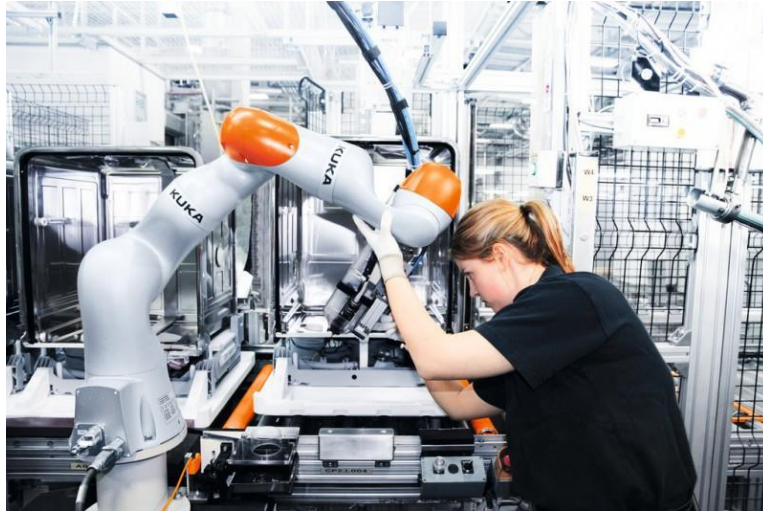


Figura 2: Colaboración hombre-máquina [5]

En la primera década del siglo XXI, debido a la evolución de los sensores, aparecen los primeros robots colaborativos (también conocidos como “cobots”). Estos robots colaborativos, son capaces de sentir la presencia de un operario cerca, de detener su movimiento en caso de que haya contacto con dicho operario de forma segura, en definitiva, de poder trabajar hombre y máquina de forma colaborativa y segura (Figura 2).

1.2.-Descripción del proyecto

El proyecto abarca el estudio, desarrollo e implementación de una célula industrial de trabajo colaborativo para instalación de cubiertas plásticas en el bajo de coches (Figura 3). En esta memoria se irá describiendo cada uno de los aspectos importantes dentro del desarrollo del proyecto. El éxito de este proyecto no se basa únicamente en la implementación final de la estación dentro de la línea de producción, sino que ese éxito está basado en los resultados obtenidos sobre el estudio de implementación de la pinza colaborativa y el impacto que supone el uso de células colaborativas con respecto a los puestos manuales.

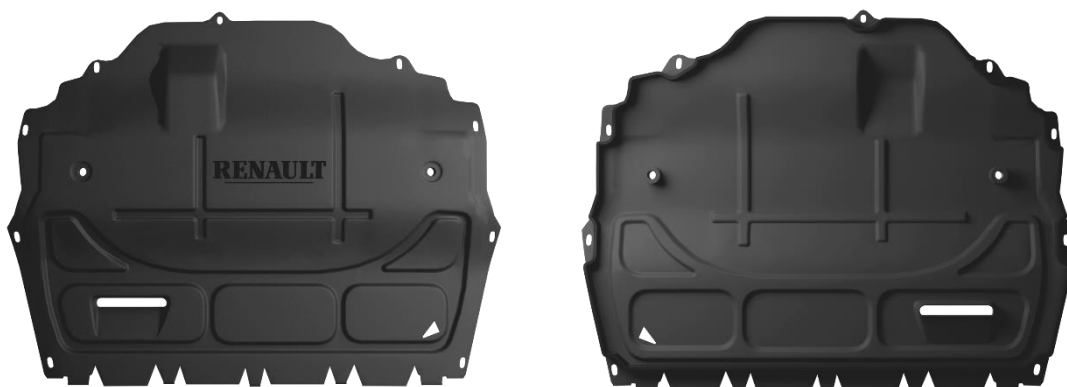


Figura 3: Cubierta plástica (Modelado aproximado)

El elemento principal de trabajo es un robot colaborativo de Universal Robot (para ser más exactos, el modelo UR10 de la serie CB. Anexo III: Componentes de la estación) cuya herramienta



de trabajo será la pinza adaptativa neumática DHEF, diseñada para trabajos colaborativos (Anexo III: Componentes de la estación). Al ser una estación colaborativa de categoría 4^{1*}, ésta contará con todos los elementos de seguridad necesarios para asegurar que el operario no pueda sufrir ningún daño. El elemento principal del sistema de control de esta estación será un autómatas de la serie S7-1500F (PLC de seguridad de Siemens). Con este autómatas, seremos capaces de controlar todos los elementos de seguridad (setas de emergencia, sensor de proximidad, rutinas de tratamiento de fallo en caso de error del robot, etc.), el control de las rutinas del robot y el control del circuito neumático para el funcionamiento de la pinza. Para hacer un sistema de control más eficiente, utilizaremos un módulo de E/S descentralizado (ET 200SP de Siemens, conectado por PROFINET. Anexo III: Componentes de la estación).

Esta célula colaborativa está siendo diseñada para la Factoría de motores del Grupo Renault de Valladolid, por lo que el diseño final de la célula debe seguir las normas y estándares referentes a estructura de proyecto de la empresa.

Debido a la pandemia que ha ocurrido en este año 2020, las prácticas de empresa que estaba realizando en Renault Valladolid se vieron suspendidas el día 13 de marzo y como consecuencia, la construcción de un prototipo ha sido sustituida por simulaciones. Con la reanudación de las prácticas se comenzará la construcción del prototipo con el cual podemos comprobar que todos los aspectos definidos son correctos.

A lo largo de todo el proyecto se lleva a cabo un estudio de los diferentes elementos que constituirán la estación, a saber:

- Estudio de valores mecánicos del robot colaborativo.
- Estudio de la zona de trabajo eficaz del robot colaborativo.
- Estudio de valores de agarre de la pinza colaborativa.
- Diseño de esquemas neumáticos para accionamiento de la pinza.
- Diseño de esquemas eléctricos de la estación.
- Estudio de puntos de agarre sobre cubiertas plásticas.
- Estudio de elementos de seguridad necesarios para asegurar la categoría 4.

Una vez realizado el estudio, pasaríamos al desarrollo de la estación fuera del entorno industrial. Como se ha indicado en la página anterior, el objetivo inicial de esta etapa era la construcción y programación de la estación siguiendo los resultados de la etapa de estudio, así como el ajuste de los errores que se hayan provocado en la etapa previa. No obstante, en esta etapa se realizarán simulaciones de los diferentes elementos que componen la estación para comprobar el correcto planteamiento de la estación y la correcta programación de los diferentes elementos de la misma. La finalidad de esta etapa es confirmar que todos los aspectos de la estación sean satisfactorios antes de llevarlo al entorno industrial.

^{1*}Los circuitos de categoría 4, según la norma UNE-EN ISO 13849-1 (“Seguridad de las máquinas”), son aquellos circuitos que pueden garantizar que la función de seguridad siempre se encuentra disponible incluso en el caso de uno o más fallos, normalmente empleando la redundancia de entrada y de salida (mediante el uso de doble canal), junto a un bucle de retorno para la supervisión continua de las salidas [6].



A lo largo del desarrollo se llevarán a cabo las siguientes funciones:

- Simulación del circuito neumático diseñado en la etapa de estudio y análisis de los resultados.
- Montaje del circuito de comunicación PROFINET de todos los elementos de control en el entorno de TIA Portal.
- Diseño y programación de la rutina del robot.
- Diseño y programación de la rutina de control del PLC con TIA Portal.
- Comprobación de correcto funcionamiento de elementos de seguridad.
- Solución y/o ajuste de fallos, en caso de que hubiera.

La siguiente etapa sería la construcción de un prototipo que se realizaría en la zona de I+D+i con la cual se comprobaría el correcto funcionamiento en su totalidad fuera del ámbito industrial. Como se indicó anteriormente, las prácticas de empresa fueron suspendidas y no será posible el montaje de un prototipo.

1.3.- Funcionamiento en automático

La estación colaborativa de este proyecto sirve para la instalación de cubiertas plásticas en la parte del bajo de los coches (su objetivo es cubrir el cárter). Esta estación cuenta con dos mantenciones (cintas transportadoras por dónde se mueven las diferentes piezas por toda la fábrica), una para la llegada de las cubiertas plásticas y otra para el transporte de los coches. Esta segunda es la manutención principal. El movimiento de estas piezas por las mantenciones está controlado por diferentes toques situados en lugares específicos a lo largo de estas.

La estación puede encontrarse en 5 posibles estados. Identificados mediante una columna de señalización (Anexo III: Componentes de la estación):

- **Estado normal:** La estación está funcionando correctamente, no hay ninguna parada de error o emergencia y no se ha solicitado una parada segura.
- **Estado de parada de emergencia por emergencia:** La estación está parada debido a que ha saltado algún elemento de seguridad de la estación (seta de emergencia, intrusión en la zona no permitida)
- **Estado de parada de emergencia por error:** La estación está en parada de emergencia debido a un fallo en el robot (colisión o se han superado los límites de seguridad establecidos)
- **Estado de parada segura:** La estación está en parada segura debido a que el trabajador ha solicitado una parada al final del ciclo de trabajo a través del panel del usuario (Anexo VI: Panel de control del usuario)
- **Saturación:** se ha producido un almacenamiento de coches en la manutención principal que provoca un retraso en el ciclo de trabajo global.

Con el funcionamiento normal de la estación, las piezas se mueven a través de sus mantenciones hasta llegar a los toques. Por una parte, los coches se mueven hasta su toque (si no hay ya un coche en el toque). Este toque sirve para indicar al sistema la presencia de un coche que se encuentra a la espera de que se termine el ciclo de trabajo anterior (si lo hay) y a la espera de que haya una cubierta también en su respectivo toque, dándose así las condiciones para que se inicie el ciclo de trabajo.



En caso de que haya un coche en el tope, el nuevo coche se moverá por la manutención principal hasta el pretope. Este pretope se encuentra antes del tope y está destinado a dar pulmón al sistema. Con la existencia de coches en el tope y en el pretope se considera que el sistema está saturado y se enciende la luz correspondiente (parpadeo de luz azul) en la columna de señalización.

Por su parte, la manutención de las cubiertas plásticas solo cuenta con un tope al final y esta manutención se mueve hasta que se detecte la presencia de pieza en el tope (este funcionamiento puede semejarse al de las cintas transportadoras de un supermercado).

Una vez hay un coche y una cubierta en sus respectivos topes, se declara el inicio de ciclo de trabajo. Se transporta el coche desde el tope hasta la zona de trabajo y el sistema de control envía una señal al robot indicando que puede empezar la rutina de trabajo. La rutina de trabajo del robot consiste en la recogida de la cubierta en su manutención y el transporte hasta dejarla en las manos del trabajador.

Tras haber instalado la pieza en el coche, el operario debe pulsar el botón "Fin de trabajo" del pupitre situado en la zona de trabajo del operario. Con el ciclo de trabajo ya finalizado, el coche procesado se mueve por la manutención hasta la salida de la estación y si hay coche y cubierta en sus topes se inicia el siguiente ciclo. Este ciclo de trabajo se puede parar de manera segura mediante los pulsadores que tiene el operario en su pupitre. Una vez se ha pulsado el botón, la estación se pausará cuando el robot haya terminado su ciclo de trabajo y se encuentre en rebuclaje (posición segura en la que se sitúa un robot cuando termina su ciclo de trabajo).

A lo largo de todo el ciclo de funcionamiento de la estación, se producirá una parada inmediata de categoría 0^{2*} en toda la estación si se pulsa alguna de las setas de emergencia de la estación o se produce algún error por parte del robot y una parada inmediata de categoría 1^{3*} en caso de que haya una intrusión en la zona restringida durante el funcionamiento normal de la estación.

Cualquiera de los defectos nombrados en el párrafo anterior se considera defecto grave y es necesario acudir al armario de control para configurar un rearme pulsando el botón destinado a ello (para que se produzca correctamente este rearme es necesario que se haya resuelto el defecto) tras haber rearmado la estación es necesario pulsar el botón de reanudación de ciclo de trabajo para que la estación vuelva a su funcionamiento normal

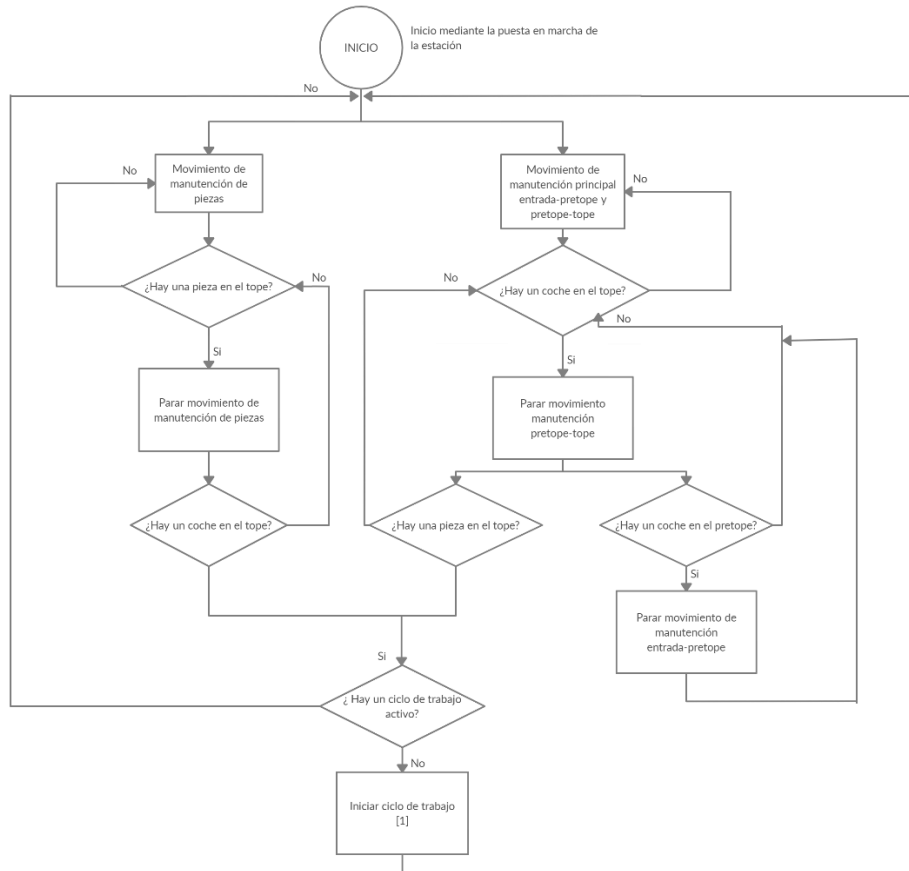
Para facilitar la comprensión del funcionamiento de la estación, en las siguientes páginas se pueden observar varios diagramas de flujo que hacen referencia al funcionamiento de los diferentes procesos que funcionan en paralelo.

^{2*} Categoría de parada 0: el movimiento del robot se detiene mediante la interrupción inmediata de la potencia. Se trata de una parada no controlada, con la que el robot puede desviarse de la trayectoria programada al frenar cada junta lo más rápido posible [7].

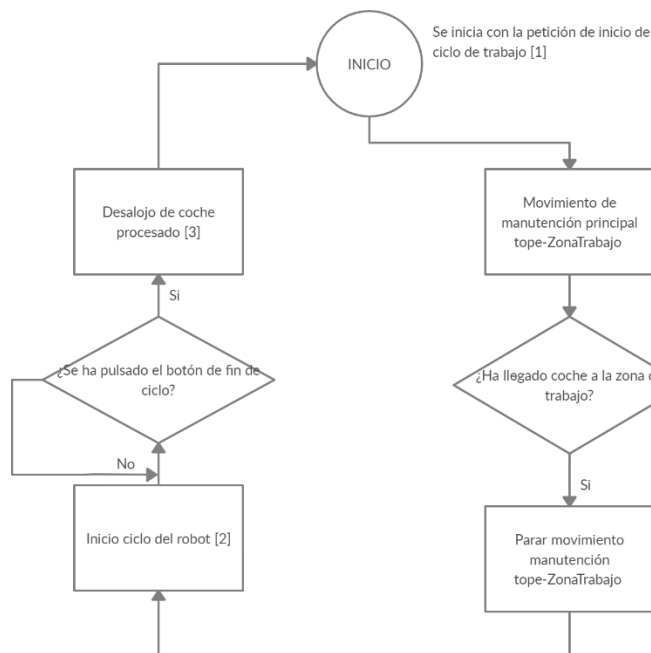
^{3*} Categoría de parada 1: el movimiento del robot se detiene sin cortar la potencia disponible hasta que se detenga, para a continuación, cortar la potencia. Se trata de una parada controlada, ya que el robot mantiene su trayectoria programada. La potencia se corta una vez el robot se detiene [7].



Proceso de control de las mantenencias de la estación

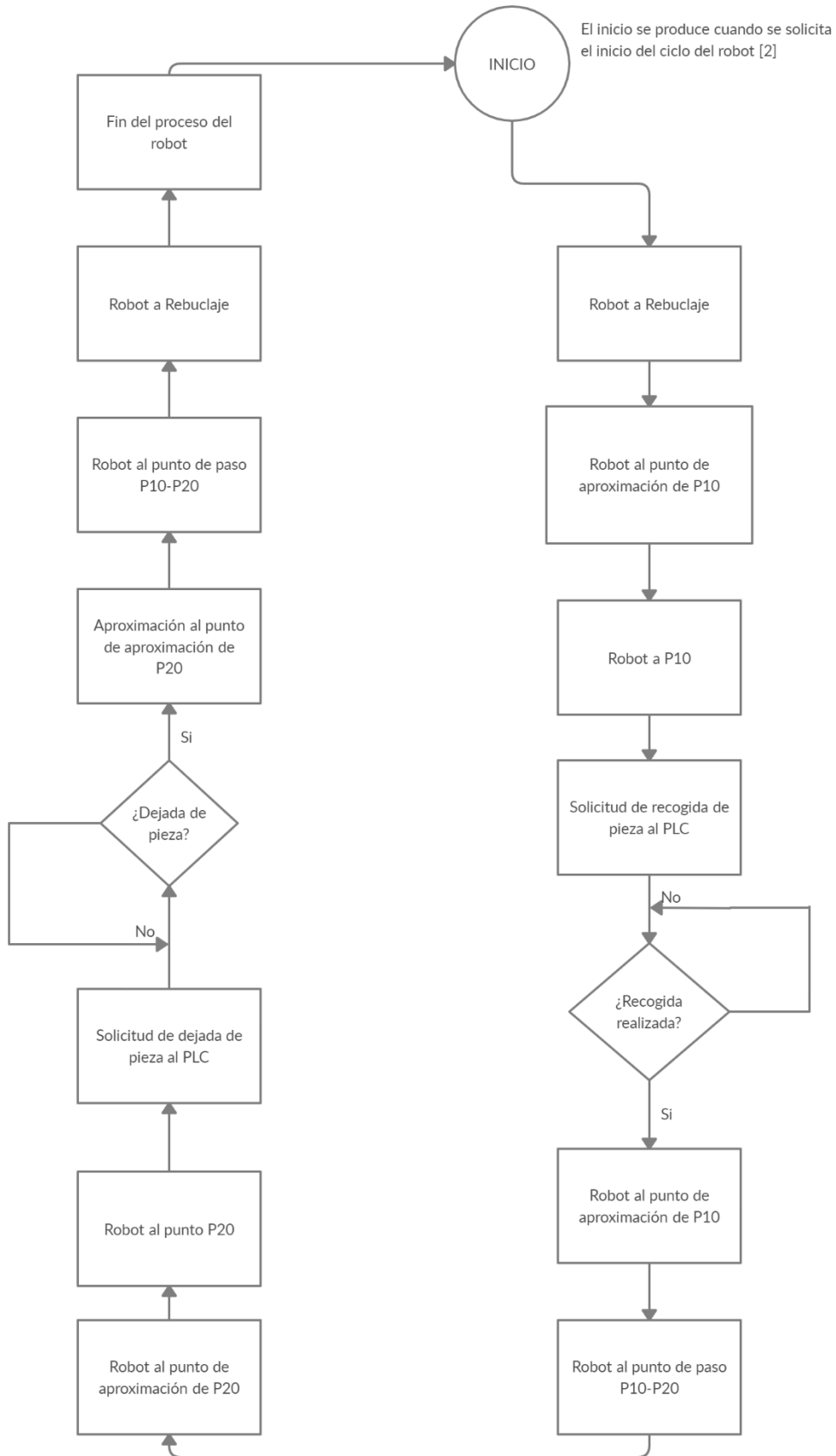


Ciclo de trabajo de la estación



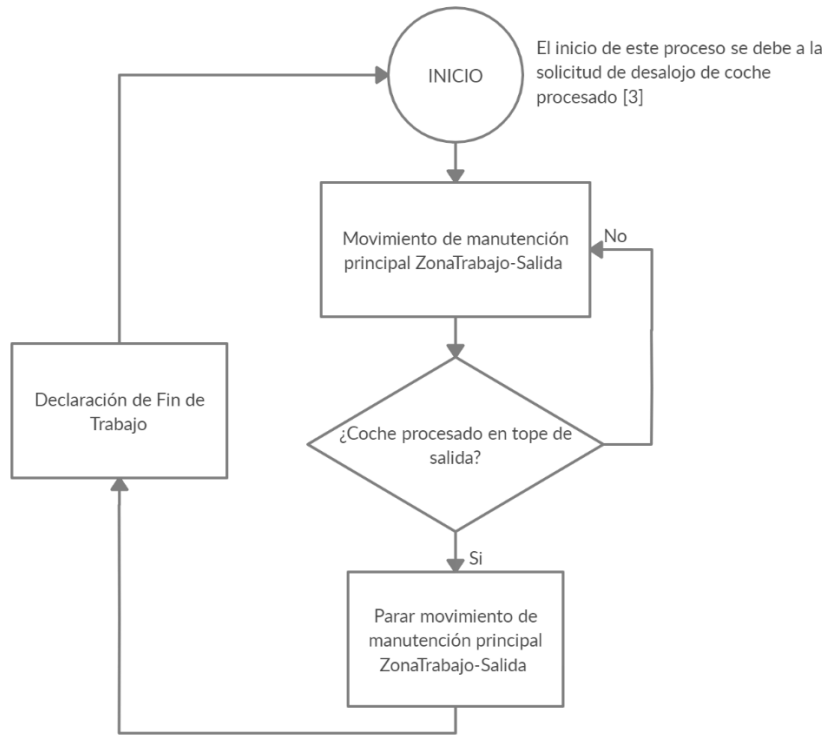


Ciclo de trabajo del robot

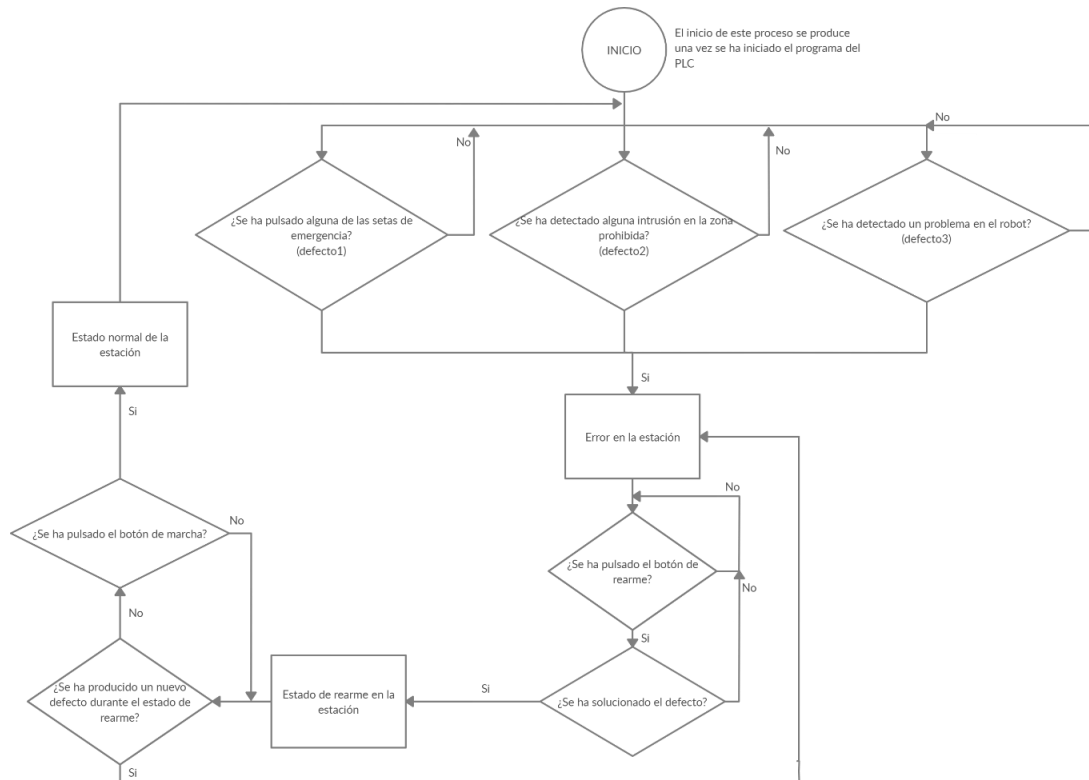




Desalojo de coche procesado



Sistema de control de seguridad





1.4.- Justificación

Este proyecto está basado en el estudio y desarrollo de una nueva metodología para la instalación de cubiertas plásticas en los bajos del coche. Con esta metodología colaborativa se busca una mayor productividad a corto/medio plazo. Y si los resultados son satisfactorios, en el futuro se buscará una mayor flexibilidad de esta estación mediante la entrada de diferentes coches y cubiertas incluyendo un sistema de detección de modelos.

Tal y como hemos indicado anteriormente, el principal objetivo de este proyecto es el estudio de nuevas tecnologías, en este caso la, pinza colaborativa mencionada; para una posible implementación en fábrica que suponga una mejora en eficiencia. La instalación de esta estación sería la primera dentro de la línea de montaje y puede suponer una renovación de numerosos puestos manuales.

Como segundo objetivo, en este proyecto busca el estudio de funcionamiento del último modelo de autómatas de Siemens, el S7-1500. Debido a que actualmente la factoría de motores de Renault se está planificando el uso de este nuevo modelo en sus próximas estaciones, sustituyendo a la serie S7-1200, que es junto a la serie S7-300, los autómatas predominantes en la fábrica.

1.5.-Búsqueda bibliográfica dentro de la UVa

Antes de empezar con el desarrollo del proyecto, se ha buscado bibliografía relacionada con la robótica colaborativa y su implementación en entornos industriales. Entre varios documentos, podemos destacar el Trabajo Fin de Grado presentado en 2017 por el estudiante de Ingeniería en Organización Industria de la Universidad de Valladolid Diego Tabuenca Alcusón. [8]

El título de su trabajo es "IMPLANTACIÓN DE ROBOTS COLABORATIVOS EN LINEA DE PRODUCCIÓN". A lo largo de su memoria, Tabuenca explica los pasos a seguir en su proyecto de implementación de un robot colaborativo en una línea de producción para la empresa Valeo Térmico. A diferencia del proyecto de Tabuenca, dónde se explica la implementación de un cobot en la línea de montaje, repasa la historia de la robótica y hace una descripción del entorno de programación del robot, nuestro proyecto está más centrado en la investigación de nuevos métodos de trabajo.

En cuanto a los otros aspectos del proyecto, como puede ser la implementación de una conexión PROFINET entre el controlador del robot y el autómatas, o la implementación y uso de una pinza neumática; no se ha encontrado una documentación bibliográfica dentro Universidad de Valladolid que reseñable.



CAPÍTULO 2: DESARROLLO DEL PROYECTO



2.1.- Introducción

En este capítulo, vamos a describir el desarrollo del proyecto a lo largo de todas sus fases, también se justificará el uso de cada uno de los componentes de la estación. Una vez se han definido los objetivos a alcanzar (descritos en el apartado 1.4.- Justificación y las condiciones a cumplir, debemos organizar y definir las fases por las que debe pasar este.

En el apartado 1.2.- Descripción del proyecto ya se habló de la organización del proyecto. En resumidas palabras, el proyecto se dividirá en 2 etapas principales (etapa de estudio y etapa de desarrollo), y en cada una de ellas se han establecido una serie de objetivos que debemos completar para poder desarrollar el proyecto de manera adecuada.

2.2.- Estado del arte

El análisis del estado del arte que aquí se realiza se dividirá en dos grupos: documentación internacional y documentación nacional. En este apartado se recogerán aquellos escritos relacionados con el tema de esta memoria que hayan sido publicados a lo largo de los últimos 7 años.

Debido a que el enfoque de este proyecto está dentro del ámbito industrial, los desarrollos referentes a este son llevados por empresas privadas y por lo tanto son de carácter confidencial. Es por esto, que la cantidad de documentación científica al respecto es limitada.

2.2.1.- Investigaciones internacionales

METZNER, Maximilian, et al. "Human-In-The-Loop"-Virtual Commissioning of Human-Robot Collaboration Systems. En *Tagungsband des 3. Kongresses Montage Handhabung Industrieroboter*. Springer Vieweg, Berlin, Heidelberg, 2018. p. 131-138. [9]

- **Resumen:** En este documento se presenta la situación actual respecto al uso de células colaborativas para procesos industriales, mencionando la gran limitación que existe para su desarrollo debido a las complejas consideraciones de seguridad y falta de herramientas diseñadas para trabajar de manera colaborativa con los seres humanos. A lo largo del escrito, se propone una metodología sustentada en la visualización del entorno de trabajo vía realidad virtual para mejorar las condiciones de seguridad.
- **Conclusiones:** Este documento supone un acercamiento al entorno de la robótica colaborativa dentro del ámbito industrial, presentando las diferentes formas de interacción posibles entre humano y robot en estos procesos. Se expone también los puntos críticos en materia de seguridad para poder establecer ese trabajo colaborativo sin riesgos. Por último, nos muestra la dirección en la que se mueve el desarrollo de las estaciones colaborativas para optimizar su funcionalidad.

DUC BINH, Vo. Robotic Tiling System. 2019. [10]

- **Resumen:** A lo largo de este documento se describe el desarrollo de un equipo basado en un robot colaborativo conectado mediante PROFINET a un autómatas de Siemens y a una pantalla HMI cuyo objetivo es establecer una interfaz entre hombre y máquina.
- **Conclusiones:** Pese a que el objetivo de este desarrollo no tiene carácter puramente colaborativo ni está enfocado en la aplicación de estos cobot en procesos de fabricación industrial (el robot del documento está centrado en el ámbito de la construcción), se



muestra una guía para la conexión PROFINET entre autómeta y robot que nos explica una de las muchas maneras de transmisión de datos entre ambos elementos.

DYER, Catherine, et al. Universal Robot Gripper. 2020.[11]

- **Resumen:** Este documento tiene un doble objetivo, la investigación y desarrollo de un tipo de *soft gripper* (un tipo de pinza diseñado para tomar objetos delicados) con una gran versatilidad a la hora de agarrar objetos.
- **Conclusiones:** La pinza desarrollada en el documento citado tiene el mismo fundamento teórico que la que va a ser utilizada en nuestra estación colaborativa, ambas se basan en una cámara de aire capaz de adaptarse a la forma del objeto a recoger y debido a dicha deformación en la cámara se provoca una variación en la presión que tiene como resultado un efecto ventosa sobre el objeto, que se queda fijo a la pinza. Pese a que el objetivo de esta pinza es su implementación en un ABB IRB 140 (robot no colaborativo), sus estudios y desarrollos permiten una mayor comprensión del funcionamiento de nuestra pinza adaptativa (Festo DHEF-20-A)

2.2.2.- Investigaciones nacionales

TABUENCA ALCUSÓN, Diego, et al. Implantación de robots colaborativos en línea de producción. 2017. [8]

- **Resumen:** Este documento se describe el desarrollo de la implementación de un robot colaborativo en una línea de producción, desde los cuadernos de carga y el fundamento teórico hasta la legislación relacionada con la seguridad y condiciones de implementación.
- **Conclusiones:** Este documento llega a ser muy útil para tener un primer contacto con la robótica colaborativa en el ámbito industrial. En este documento podemos encontrar algunos procedimientos a seguir para cumplir con la normativa vigente sobre la construcción de células industriales, se realiza una explicación detallada para diseñar un programa en el robot, entre otros.

2.3.- Descripción de los elementos empleados

En este apartado se realizará un desarrollo de cada uno de los objetivos y los aspectos más importantes de los diferentes elementos de la estación. Para una información más general, podemos consultar el Anexo III: Componentes de la estación.

2.3.1.- Robot colaborativo

Para el desarrollo de una célula colaborativa, se necesita un robot colaborativo [3]. Estos robots colaborativos (también conocidos como 'cobots') cuentan con una serie de sensores específicos diseñados para que este robot pueda trabajar con un operario de forma segura. La mayoría de los robots industriales de la actualidad no son colaborativos y estar cerca de su campo de trabajo durante la ejecución de su rutina es un peligro que puede llegar a provocar heridas graves (atrapamientos, amputaciones...) o incluso la muerte. Esto se debe a la falta de sensores que aporten una realimentación del entorno en el que trabajan estos robots industriales.



Para este proyecto, en lugar de elegir el robot colaborativo que mejor se adapte a las necesidades de la estación, se ha establecido un modelo en específico (el UR10 de la serie CB) por lo que en lugar de estudiar qué robot es el óptimo para nuestra estación, estudiaremos si las características de este modelo son válidas para nuestra estación. El estudio de viabilidad del robot se realiza en el apartado 2.4.- Estudio de los elementos.

2.3.2.- Sistema de control

En toda célula industrial, se deben cumplir las normas UNE-EN ISO 13849-1 [6] e UNE- EN ISO 11200:2014 [12] referentes a la seguridad de las células industriales. Para administrar estos elementos de seguridad, debemos tener un sistema de control. Actualmente el sistema de control más eficaz y el más utilizado en el mundo industrial son los autómatas (o PLC, Programmable Logic Controller).

En Renault Valladolid la amplia mayoría de estaciones en las fábricas de motores están controladas por autómatas Siemens, en especial de la serie S7-300 y S7-1200. Con el fin de unificar los modelos de autómatas de las nuevas estaciones se ha decidido utilizar la nueva serie S7-1500 para aquellas estaciones que se van a instalar en la actualidad y en adelante. Una ventaja que nos ofrece este modelo sobre sus antecesoras es que, debido a su mayor potencia, este nuevo modelo permite el control de varias pantallas IHM además del control de la estación (con modelos anteriores era necesario un PLC para el control de la estación y otro para el control de las pantallas IHM, todos los PLC se conectaban entre si).

Al igual que con el robot, este modelo de PLC ya está determinado y lo que debemos estudiar es la configuración de los módulos que acompañaran a la CPU principal, así como la necesidad de una periferia descentralizada.

2.3.3.- Elementos de seguridad

En una estación de estas características es necesario la implementación de medidas de seguridad que garanticen el correcto funcionamiento de la misma sin accidentes que puedan causar lesiones tanto al trabajador como a cualquier operario dentro de las inmediaciones de la estación colaborativa. En el Anexo III: Componentes de la estación podemos encontrar una información más detallada.

El elemento de seguridad imprescindible en cualquier célula industrial son las setas de emergencia. Estas setas de emergencia sirven para provocar una parada inmediata de toda la estación cuando esta es pulsada. Para asegurar el funcionamiento de esta seta se utiliza lógica negativa, la cual nos permite observar si la seta está desconectada o no funciona correctamente. Por otra parte, al ser una estación de categoría 4, es necesario que la señal de la seta de emergencia sea de doble canal simétrico (ambos canales llevan la misma señal al sistema de seguridad). Ya que un fallo en uno de los canales no supondría la inhibición de este elemento de seguridad.

El otro elemento de seguridad de esta estación es un sensor laser de proximidad. Pese a que se trata de un robot colaborativo, este puede causar lesiones de diferentes niveles de gravedad. La zona de mayor peligrosidad es la zona de recogida de las piezas plásticas. Es por ello que, para asegurar la seguridad, se ha incluido un sensor de proximidad láser capaz de detectar la



presencia de un usuario en dicha zona peligrosa. Al igual que con la seta de emergencia, la señal del sensor llega al sistema de control mediante el uso de doble canal simétrico para garantizar que, en caso de fallo de alguno de los dos canales, esta medida de seguridad no es inhibida.

2.4.- Estudio de los elementos

Esta es la primera etapa que todo proyecto debe tener. El objetivo de esta etapa es buscar las soluciones que permitan la realización del proyecto de forma óptima. Para ello, debemos estudiar en primer lugar, aquellos elementos que, de acuerdo con las condiciones del proyecto, se deben implementar, como puede ser el cobot UR10 y la pinza neumática adaptativa.

2.4.1.- Estudio de valores mecánicos del robot colaborativo

Todos los robots industriales tienen unos valores mecánicos que indican las condiciones en las que pueden trabajar de forma óptima y los valores máximos con los que pueden asegurar un funcionamiento correcto sin que se produzcan desgastes o daños en el robot de forma prematura. Los valores de estas características mecánicas se pueden encontrar en la ficha técnica del robot (Anexo III: Componentes de la estación).

Es muy común que, dentro de una misma serie de robots, haya varios modelos con diferentes valores mecánicos. El objetivo de esta variedad de modelos es alcanzar una mayor flexibilidad a la hora de elegir un robot para llevar a cabo un proceso industrial y la empresa elija dentro de la serie, el modelo que mejor se adapte a sus necesidades.

Como se ha indicado con anterioridad, en este proyecto, partimos con un robot en concreto. En lugar de estudiar el entorno en el que va a trabajar y las cargas con las que va a operar, estudiaremos si con este cobot podemos llevar a cabo dichas operaciones estando dentro de los valores mecánicos del mismo.

Para ver los valores mecánicos del UR10, podemos acceder a su ficha técnica [13] (Anexo III: Componentes de la estación). Esta ficha puede ser encontrada en la página oficial de 'Universal Robot' de forma rápida y sencilla gracias a la política que tiene la empresa sobre la distribución de documentación y software de sus robots.

En caso de no tener acceso a la ficha técnica, el nombre del modelo hace referencia al valor de carga máxima que puede soportar el robot. De esa manera, nuestro cobot, que es el modelo 'UR10', tendrá una carga máxima en la última articulación (muñeca 3) de 10kg(22lbs).

Otra característica a tener en cuenta es que el controlador del robot cuenta con una protección IP44. El grado de protección IP indica con las dos cifras que le suceden el nivel de protección frente al ingreso de objetos sólidos y el ingreso de líquidos respectivamente [14]. El primer 4, indica que tiene un nivel de protección frente al ingreso de partículas de hasta 1mm de diámetro. El segundo 4, indica que su nivel de protección frente al ingreso de agua permite que no entre agua arrojada desde cualquier ángulo a un promedio de 10 litros por minuto y a una presión de 80-100kN/m² durante un tiempo que no sea menor a 5 minutos.

Por su parte, el brazo robótico y la consola de programación cuentan con una protección IP54, es decir, una protección frente al ingreso de polvo en una cantidad que pueda interferir con el correcto funcionamiento.



Nuestra estación estará en la zona de montaje motor, por lo que no hay peligro de ingreso de virutas metálicas en las articulaciones del robot, como si pasase en mecanizado. La pinza con la que trabaja el motor pesa 0.475Kg y las cubiertas plásticas que tomará tienen un peso inferior a 2Kg. Es así, que los valores mecánicos del robot se adaptan satisfactoriamente al desempeño que deseamos que realice.

2.4.2.-Estudio de la zona de trabajo del robot colaborativo

Una vez hemos estudiado los valores mecánicos del robot, dentro de la misma ficha técnica debemos observar otras características técnicas, a destacar el alcance y área de trabajo de nuestro robot.

Los cobots de 'Universal Robots' están diseñados para que puedan ser instalados en cualquier orientación (empotrados a la pared, colgados del techo o en un ángulo específico). En nuestra estación, el robot estará anclado paralelo al suelo y ligeramente elevado gracias a una base de indexado. El peso total del robot es de 33.5kg (73.9lb).

Por otra parte, el robot cuenta con una huella de 190mm (7.48in) de diámetro con 4 taladros M6 para su anclaje a una superficie y así evitar que se realicen modificaciones (otros taladros) en la base que modifiquen los valores mecánicos del brazo robótico.

El alcance máximo del brazo robótico es de 1300mm (51.2in) sin herramienta (Figura 4), logrando hasta 1525mm (60.03in) con la pinza instalada y con el émbolo desplegado.

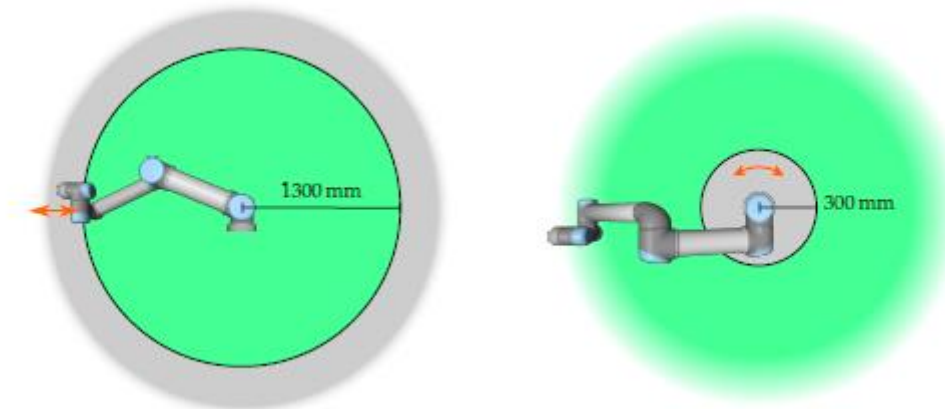


Figura 4: Diagrama de alcance del robot UR10 CB Series [15]

Con los valores de alcance conocidos, debemos estudiar la disposición de las manutenciones y el resto de los elementos de la estación de forma que el robot, con la pinza instalada, pueda llegar a cada uno de los puntos de la estación sin mayor problema. En caso de que no se encontrara una disposición de los elementos de la estación que permita el funcionamiento óptimo del robot, se debería estudiar la implementación de otro modelo.



2.4.3.- Estudio de valores de agarre de la pinza colaborativa

La pinza colaborativa consta de un cilindro de doble efecto. El extremo del vástago está anclado, por la parte interior, al extremo de la cámara de aire y le permite alterar la forma de esta.

El rango de presión de funcionamiento del actuador oscila entre 1 y 8 bares, teniendo como valor nominal 6 bares. A esa presión, obtenemos una fuerza del actuador de 158N y 189N en retroceso y avance respectivamente y una fuerza de agarre de 26N en el eje paralelo a la pinza y de 45N en el eje perpendicular a la pinza [16]. Con 6 bares, podemos lograr la sujeción de un objeto con una masa máxima de 2.65kg, con una masa superior, la fuerza del peso provocado por la gravedad sería superior a la fuerza de sujeción y no sería posible el agarre. Nuestra instalación cuenta con una red de aire comprimido no lubricado de 6 bares de presión.

Los objetos que esta pinza va a tomar son cubiertas plásticas para motores, cuyo peso es inferior a 2kg, por lo que la pinza sería capaz de sujetarla sin problema. Más adelante hará falta el cálculo de fuerzas para ajustar la velocidad a la que se puede mover el robot asegurando que la pieza no se suelte debido a la inercia.

2.4.4.- Diseño de esquemas neumáticos para accionamiento de la pinza

Esta pinza es un actuador y no cuenta con ninguna válvula interna capaz de controlar el flujo de aire para hacer que el vástago avance y retroceda, por lo que es necesario diseñar un circuito neumático para poder controlar el funcionamiento de la pinza. A continuación (Figura 5), podemos ver el esquema eléctrico. No obstante, en el Anexo I: "Planos" podremos encontrar el plano correspondiente a una escala mayor.

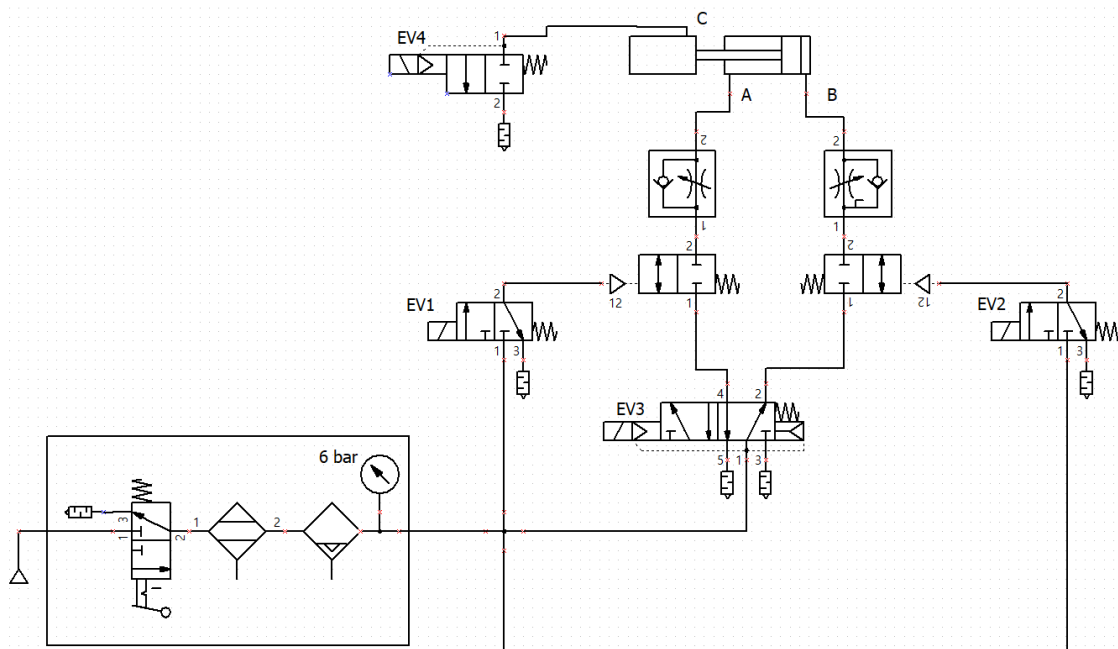


Figura 5: Esquema neumático de control de funcionamiento de la pinza adaptativa



2.4.5.- Diseño de esquemas eléctricos de la estación

A parte del esquema neumático, es necesario diseñar un esquema eléctrico. Tanto el controlador del robot, como el autómatas y sus módulos, pasando por las electroválvulas y los elementos de seguridad necesitan una alimentación para funcionar, esta fuente de energía es diferente para cada uno de los elementos, por lo que es necesario el uso de transformadores y otros medios con los que podemos suplir la energía de alimentación de cada componente.

A continuación, podemos ver una figura con el esquema eléctrico, no obstante, en el Anexo I: 'Planos' podremos encontrar el plano correspondiente a una escala mayor y con las descripciones necesarias.

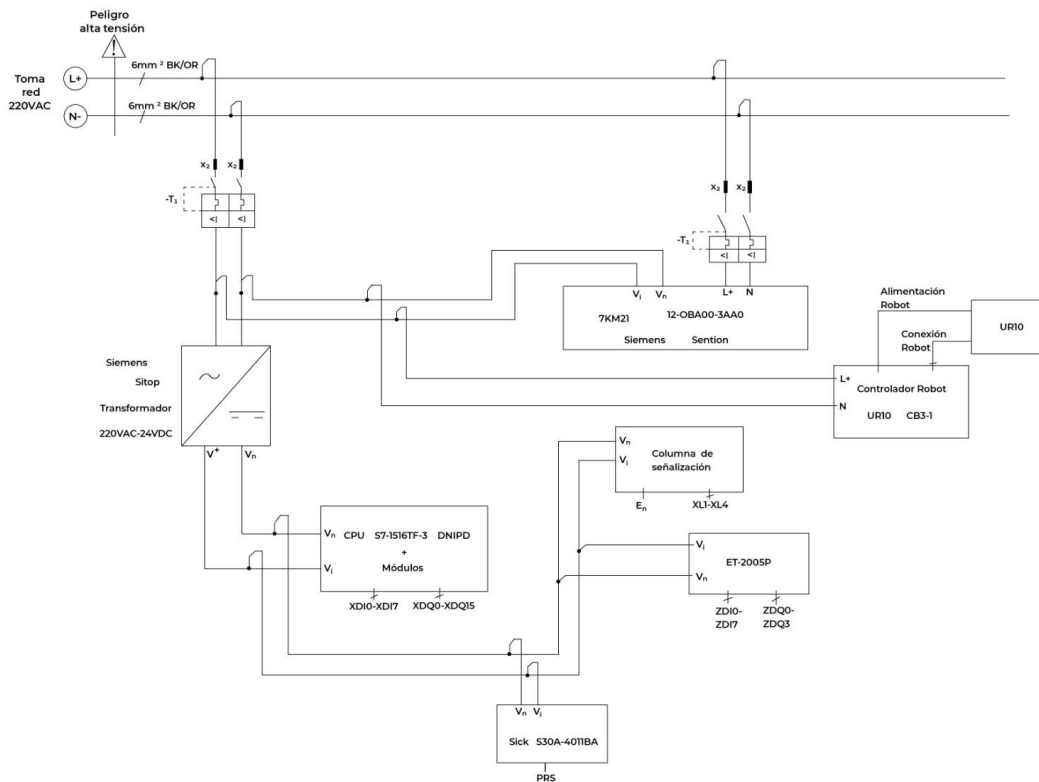


Figura 6: Esquema eléctrico

2.4.6.- Estudio de puntos de agarre sobre cubiertas plásticas

El punto de la cubierta sobre el cual el robot debe atacar es de principal importancia. Si este punto se encuentra muy alejado del centro de masa de la cubierta, se generarán momentos de inercia elevados que pueden causar que la pinza se suelte al menor movimiento. Por otro lado, si el punto sobre el que la pinza ataca es una superficie con una geometría complicada, es posible que no se pueda hacer un efecto ventosa óptimo, provocando el problema que hemos descrito anteriormente.

A partir del plano y las características de la cubierta, debemos hacer un estudio con el fin de obtener uno o varios puntos con los cuales se puede producir un agarre correcto. Es así, que en



la figura 7 podemos observar la zona que se ha designado como “óptima” para realizar el agarre de la pieza minimizando la probabilidad de que se suelte o que haya otros problemas.

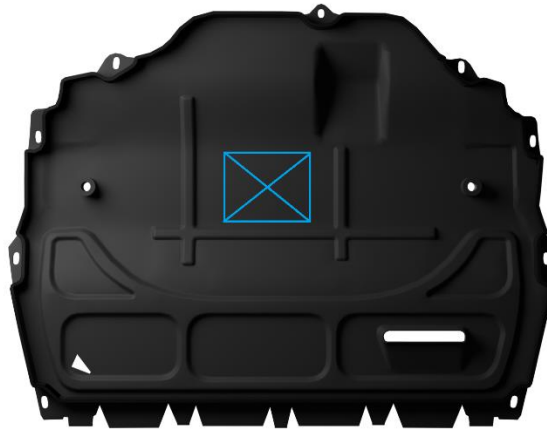


Figura 7: Zona de agarre de la cubierta plástica

2.4.7.- Estudio de elementos de seguridad necesarios para asegurar la categoría 4

Una vez más, los circuitos de categoría 4 son aquellos circuitos que garantizan que la función de seguridad siempre se encuentra disponible incluso en el caso de uno o más fallos, normalmente empleando la redundancia de entrada y de salida, junto a un bucle de retorno para la supervisión continua de las salidas. Para que nuestra estación recoja las características necesarias para ser de categoría 4 es necesario añadir diferentes elementos de seguridad y realizar una programación redundante sobre estos elementos.

En primer lugar, uno de los elementos principales son las setas de emergencia. Se añadirán un total de 3 setas de emergencia. Una al alcance del operario, otra en la zona dónde se encuentra el armario eléctrico y la TeachPendant (por si es necesario realizar alguna intervención) y una tercera en una de las paredes de la estación.

El cobot, cuenta con sensores de fuerza en cada una de sus articulaciones. Estos sensores le permiten detectar un esfuerzo superior al permitido (según la configuración de seguridad que se ha implementado) en alguna de sus articulaciones y parar el movimiento del robot en caso de que se produzca. Este error es interno al controlador del robot, pero mediante la conexión PROFINET con el autómatas logramos que este sea consciente de este error y pueda realizar una serie de tareas en respuesta.

2.5.-Conexión de los elementos

Es imprescindible definir las conexiones entre todos los elementos de la estación. La conexión de los diferentes elementos se realiza de tres posibles maneras.

Los diferentes elementos de control (el PLC, la periferia descentralizada y el controlador del robot) están conectados entre sí mediante los protocolos PROFINET [17] para señales ordinarias y PROFISAFE para señales de seguridad. Es necesario crear una red dónde el elemento principal sea la CPU S7-1500F que actuará como maestro de la comunicación, mientras que el resto de los elementos actuarán como esclavos.



Para la conexión de otros elementos que no son de seguridad (columna de señalización, topes de la manutención o algunos de los botones del armario eléctrico) se usa lógica cableada de canal simple. Es decir, un cable de cobre es el encargado de enviar la señal del elemento al PLC y otro cable diferente la señal del PLC al elemento de seguridad.

Por último, para aquellos elementos de seguridad (seta de emergencia y sensor de proximidad) que lo necesitan, es necesario el uso de señales redundantes, es decir, cada uno de los elementos de seguridad utiliza dos cables para enviar la misma señal por cada uno de ellos al PLC.

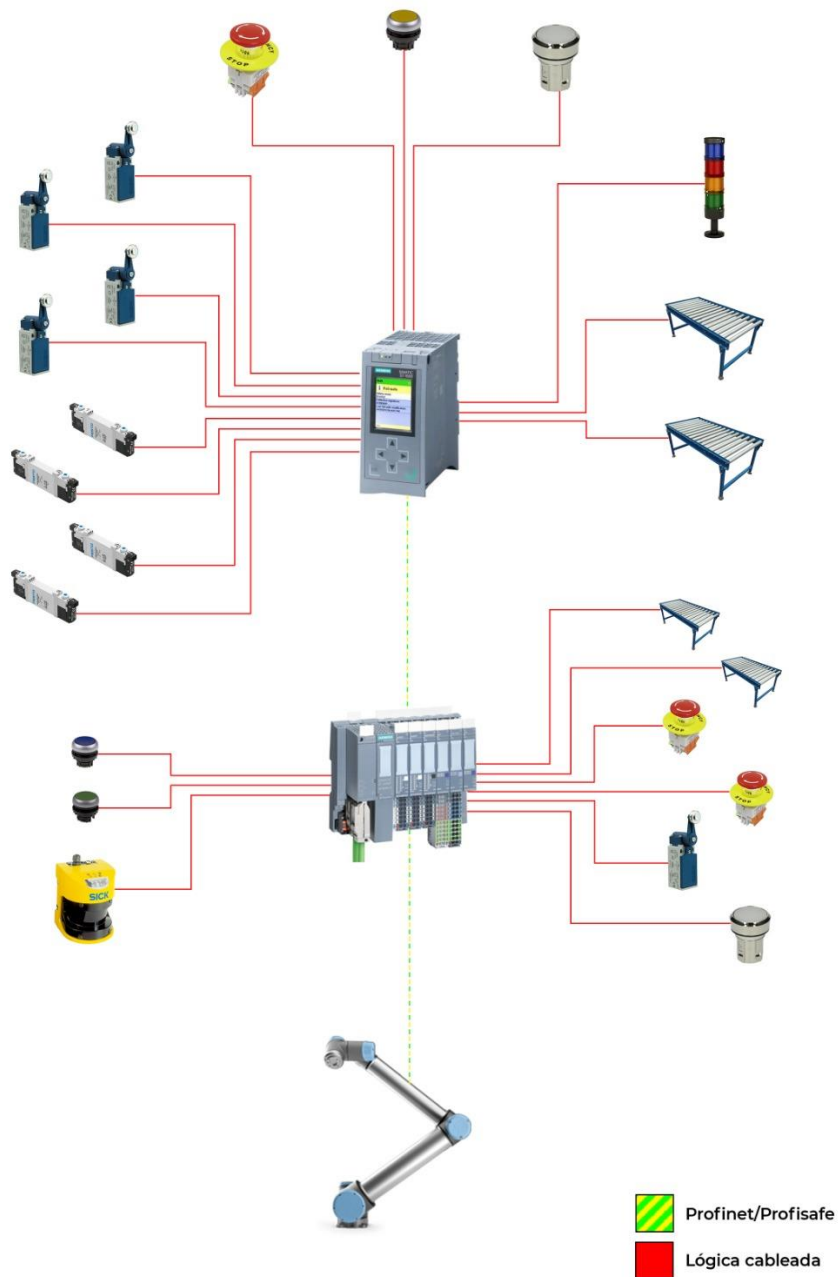


Figura 8: Conexión de los elementos



2.6.- Simulación de esquemas neumáticos

En este apartado hablaremos del desarrollo de la simulación del circuito neumático encargado del funcionamiento de la pinza y las conclusiones derivadas de esta. El objetivo de esta simulación es el estudio del correcto funcionamiento del cilindro de doble efecto de la pinza. La cámara de aire es un elemento trivial cuyo único elemento neumático es una electroválvula 2/2 bidireccional que permite la salida y entrada del aire o que no haya flujo de aire.

El software utilizado para realizar la simulación es el FluidSim de la marca de elementos neumáticos y electrónicos Festo. Se utilizará una versión de prueba que nos proporciona 30 días de licencia didáctica. Esta licencia didáctica nos permite trabajar únicamente durante 30 minutos (tras ese tiempo el programa se cierra), no se podrá guardar el archivo con el que estamos trabajando y en el espacio de trabajo aparecerá una marca de agua con la palabra “Demo”. FluidSim se puede descargar desde la página oficial de Festo. [18]

En primer lugar, hablaremos del programa FluidSim. Este programa es un entorno que nos permite realizar esquemas de circuitos neumáticos, eléctricos e hidráulicos y después simularlos (No obstante, nuestro interés se centra únicamente en el apartado neumático). El programa cuenta con una gran variedad de bloques predefinidos con los que podemos crear de manera sencilla nuestros circuitos. Dentro de la simulación, podemos actuar en tiempo real sobre las diferentes válvulas para cambiar su posición y observar los cambios que se producen al cambiar el estado de las válvulas.

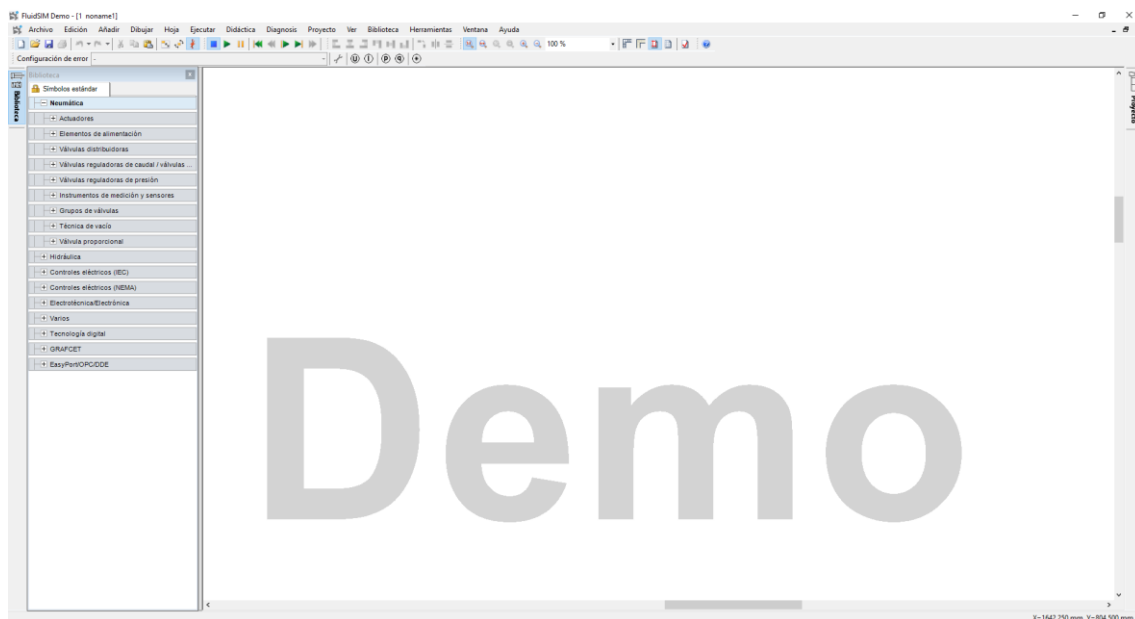


Figura 9: Entorno de diseño de esquemas de FluidSim

En la figura 9 observamos el entorno de creación del programa. En la parte izquierda podemos apreciar una serie de pestañas dónde encontraremos los bloques predefinidos. Dentro de cada grupo principal (Neumática, Hidráulica, Controles Eléctricos, ...) encontramos las diferentes categorías en las que se pueden dividir los elementos de dicha categoría (Actuadores, Elementos de alimentación, Válvulas distribuidoras, ...).



A la derecha de estas pestañas encontramos el entorno de desarrollo, una ventana blanca con la marca de agua “Demo” (en caso de ser versión de prueba) dónde posicionaremos los bloques predefinidos que encontramos en la zona de pestañas. Al iniciar la construcción del sistema nos aparece un rectángulo rojo referenciando el tamaño de una hoja de papel DIN-A4.

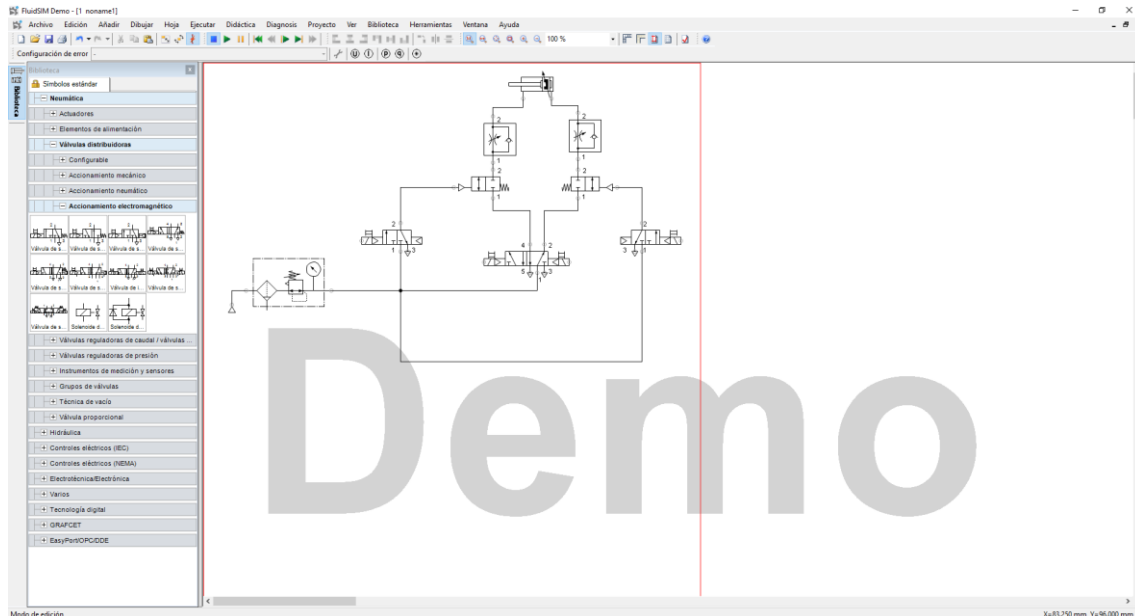


Figura 10: Esquema neumático definido

En la figura 10 podemos apreciar el circuito neumático a simular ya definido. Tal y como vimos en el capítulo de la etapa de estudio, el sistema toma el aire comprimido no lubricado de la red general de aire y lo pasamos por la estación de acondicionamiento para asegurarnos de que el aire se encuentra seco y a la presión correcta (6 bar).

El sistema simulado cuenta con 3 electroválvulas, siendo la más importante la electroválvula 5/2 encargada de dirigir el flujo en un sentido determinado hacia el cilindro de doble efecto.

Un problema que encontramos en el programa es que sus bloques predefinidos son elementos básicos y no cuenta con bloques compuestos. En nuestra estación, las válvulas antirretornos con estrangulación de flujo de aire llevan incluidos también unas válvulas 2/2 con accionamiento neumático para control de dirección del flujo. En el programa se han introducido esos 2 elementos de manera independiente.

Con el esquema ya construido, explicado y la configuración de estrangulación de las válvulas elegido, pasamos a realizar la simulación. En la parte superior del programa, observamos los elementos principales para la simulación. Al no ser un esquema de gran complejidad, únicamente nos interesan los 3 primeros elementos: Stop, Play y Pause.

NOTA: De cara a facilitar la simulación se ha sustituido el retorno por muelle de las electroválvulas 2/2 por un accionamiento manual. Con el accionamiento por muelle al soltar la electroválvula vuelve al estado de reposo y no podemos activar sendas electroválvulas de manera simultánea con esa configuración. Es un cambio que no afecta al devenir de la simulación.



Al pulsar el botón de inicio (Play) comienza inmediatamente nuestra simulación. Al iniciar la simulación podemos observar la presión detectada por el manómetro en cada momento. Las vías neumáticas aparecerán con dos colores: azul oscuro para indicar el paso de aire y azul claro para indicar que no hay flujo de aire.

Para extender el vástago necesitamos posicionar la electroválvula 5/2 en la posición correcta y habilitar el flujo de aire por ambas electroválvulas 2/2 para permitir el flujo de aire en ambas direcciones. En la figura 11 podemos observar la configuración necesaria para extender el vástago.

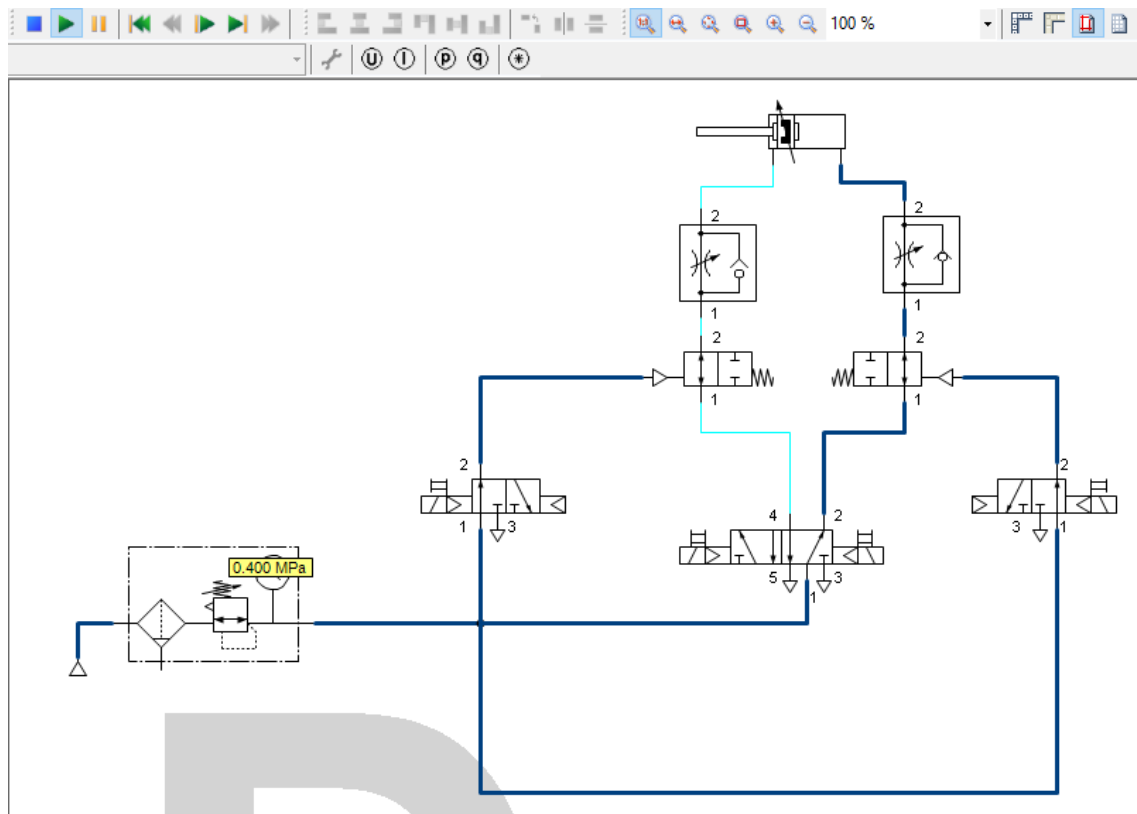


Figura 11: Vástago desplegado

Para recoger el vástago, seleccionamos el otro estado de la electroválvula 5/2 y mantenemos ambas electroválvulas 2/2 activadas para permitir el paso de flujo de aire comprimido. En la figura 12 (siguiente página) podemos observar la configuración y el sentido del flujo de aire.

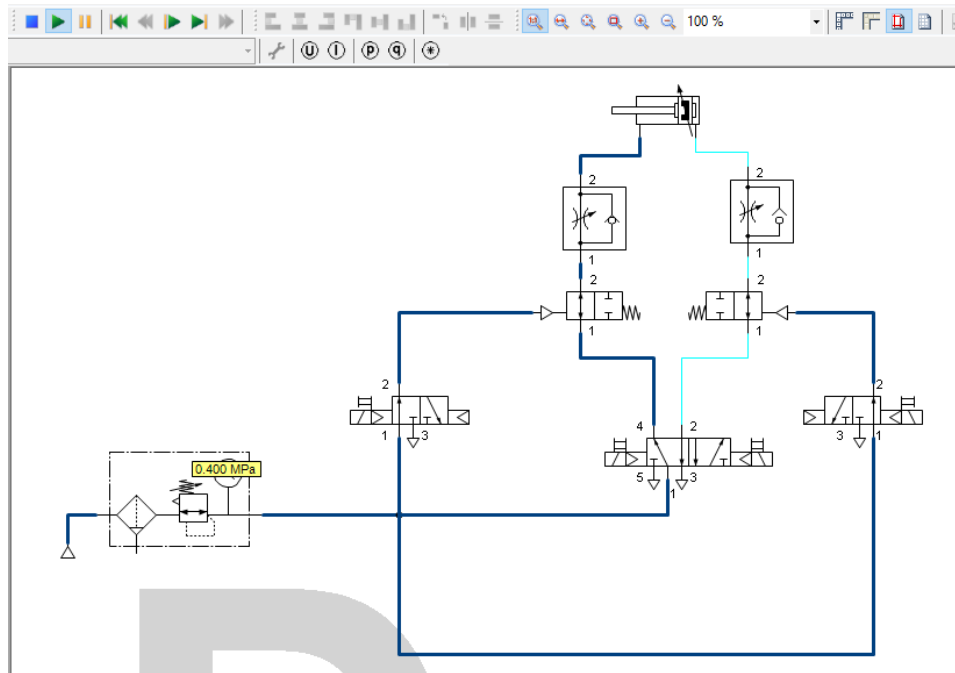


Figura 12: Vástago recogido

Otro estado en el que podemos encontrarnos es mantener la posición del vástago fija eliminando el flujo de corriente por el circuito. Para conseguir mantener la posición del vástago fija debemos cerrar las dos electroválvulas 2/2 y la electroválvula 5/2 puede estar en cualquier posición. Si cerramos una electroválvula 2/2 y dejamos la otra abierta, el vástago tendrá cierto movimiento, por lo que debemos cerrar ambas electroválvulas 2/2 para mantener fija la posición del émbolo (Figura 13).

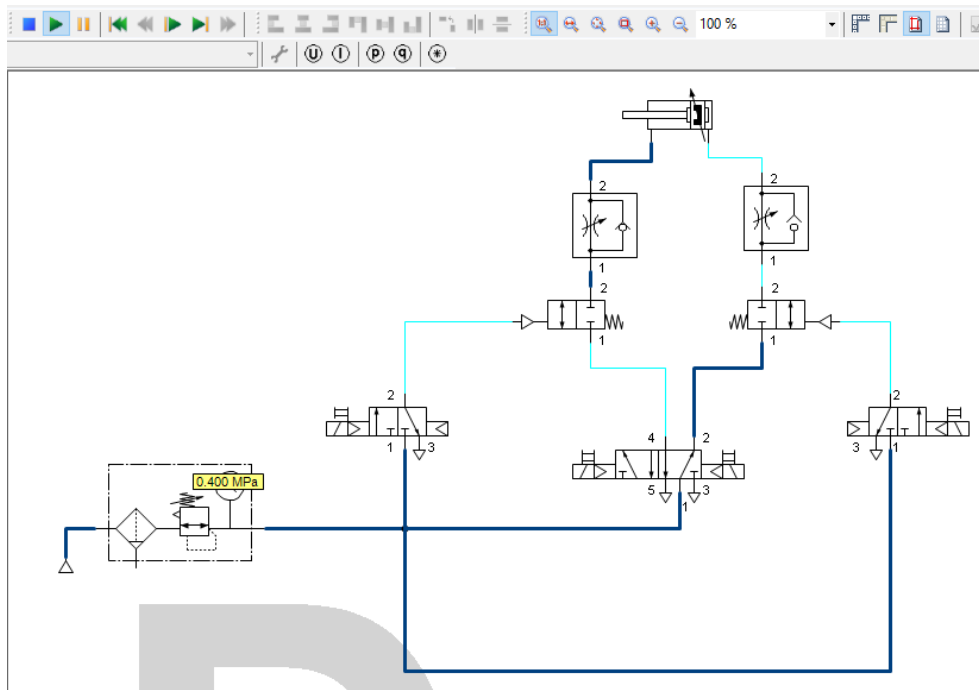


Figura 13: Posición fija del vástago



Tanto la simulación de los robots colaborativos, cómo el modelo de pinza que vamos a utilizar en esta estación es muy novedoso y no se ha desarrollado en la actualidad un software específico para realizar simulaciones de manera fidedigna a la realidad. Para poder hacer simulaciones de estos elementos, en algunos casos, es necesario realizar aproximaciones como las que hemos hecho en este apartado.

La simulación que hemos realizado sobre el circuito neumático nos da la información necesaria y objetiva para asegurar el correcto funcionamiento del émbolo de doble efecto de la pinza, y el estudio teórico realizado sobre la pinza nos da información para asegurar el correcto funcionamiento de la cámara de aire de la pinza. Podemos confirmar que este modelo es lo suficientemente confiable y los resultados suficientemente satisfactorios como para pasarlo a la etapa de pruebas.

2.7.- Programación y simulación del programa de control del PLC

2.7.1.- Descripción de red Hardware

En cuanto a la programación del PLC, usaremos el entorno de programación de SIEMENS, el TIA Portal. Lo primero que debemos hacer es la descripción de hardware de nuestro sistema.

En primer lugar, debemos decir que nuestro hardware de control cuenta con 3 agrupaciones principales:

- **El bloque del PLC-1516F:** que se encuentra en el armario de control y cuya descripción de hardware se encuentra en el Anexo IV: “Hardware del sistema de control”.
- **El bloque de la periferia descentralizada:** controlado por el ET 200SP, acoplado a la base del robot UR10 y cuya descripción de hardware se encuentra en el Anexo IV: “Hardware del sistema de control”.
- **El bloque de controlador del cobot UR10:** situado al lado del armario de control, debido a que este controlador tiene conectado una TeachPendant con la que podemos modificar la rutina del robot y rearmar el mismo en caso de fallo. La descripción del hardware de este controlador la podemos encontrar en el Anexo IV: “Hardware del sistema de control”.

Este último bloque, definido como ‘URIODev1’ no está incluido en el catálogo por defecto del programa, por lo que es necesario descargar de la página oficial de Universal Robots en fichero GSDML para importarlo [19]

Con el archivo descargado, abrimos nuestro proyecto y en la barra de herramientas pinchamos sobre la ventana de “Opciones” y dentro de ese menú en “Administrar archivos de descripción de dispositivos”, tal y como podemos ver en la figura 14 (siguiente página).

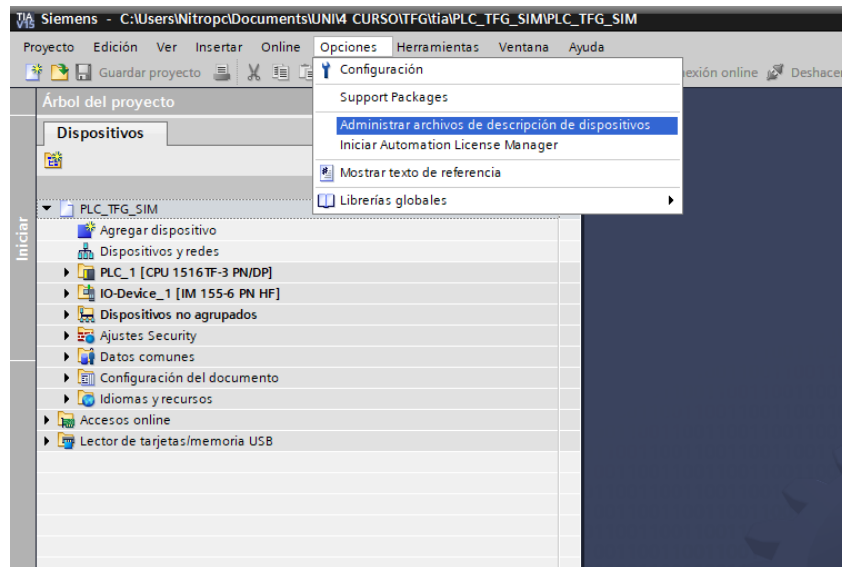


Figura 14: Ruta para importar el archivo GSDML

Una vez hayamos seleccionado esa opción nos aparecerá un menú como el de la figura 15. En la ruta de origen debemos seleccionar la dirección dónde hemos guardado la descarga del fichero GSDML y en la parte inferior, en “Contenido de la ruta importada” nos debería aparecer el fichero que queremos instalar. Lo último que queda en este menú es seleccionar el archivo y pulsar en “Instalar”.

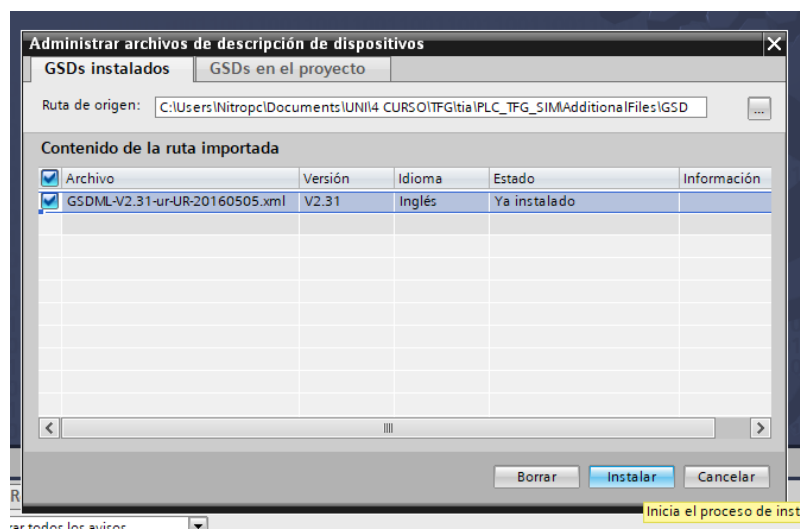


Figura 15: Menú Administrar archivos de descripción de dispositivos

Si el fichero se ha instalado correctamente, nos aparecerá en el catálogo de hardware cuando entremos al menú de “dispositivos y redes”.

En la guía profinet de Universal Robots [19] se puede ver un tutorial oficial para la implementación del controlador en el entorno TIA Portal junto con una demo y los archivos necesarios para llevar a cabo la implementación.



Una vez se han definido los 3 bloques que componen nuestra red de control, debemos establecer la conexión PROFINET entre ellos.

Según el centro de formación técnica para la industria [20]: “PROFINET es un protocolo de comunicación Ethernet industrial basado en los estándares abiertos TCP/IP y desarrollado con un enfoque de semejanza a PROFIBUS DP [21]. Así mismo, es un mecanismo para intercambiar datos entre controladores y dispositivos.”

Esto supone una conexión física entre los controladores mediante cables Ethernet de tipo RJ-45, dónde cada uno de los elementos de la conexión cuenta con una dirección IP propia y única dentro de la red.

El S7-1500 cuenta con un total de 3 puertos divididos en 2 grupos cada uno con una dirección IP diferente (en este caso 192.168.1.1 y 192.168.0.1 respectivamente) lo que permite a este PLC estar conectado de forma simultánea a dos redes diferentes. En nuestra estación utilizaremos una única red, por lo que dentro del S7-1500, usaremos el grupo que cuenta con dos puertos RJ-45 y tiene la IP asociada 192.168.0.1. El otro puerto restante con la dirección IP 192.168.1.1 quedará libre.

El PLC será el maestro de la conexión, mientras que los otros dos bloques de control serán los esclavos con unas direcciones IP de 192.168.0.2 para el ET-200SP y 192.168.0.3 para el controlador del UR. En la figura 16 podemos ver el esquema de conexión hardware que nos proporciona TIA Portal.

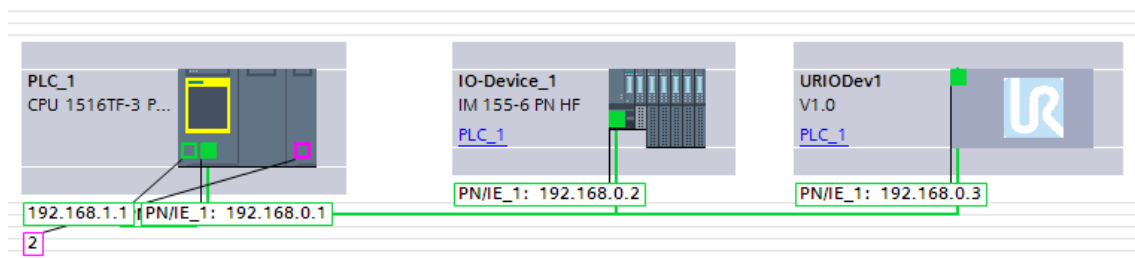


Figura 16: Esquema de configuración del hardware de la estación en TIA Portal

Otro punto importante en la conexión de los diferentes elementos al PLC es la asignación de las posiciones de memoria dentro del PLC. Para la carta de entradas digitales del ET200SP (DI 8x24VDC ST) usaremos el byte 52 (%I52.0 ... %I52.7) y para la carta de salidas digitales (DQ4x24VDC ST) del mismo, usaremos el byte 50 (%Q50.0...%Q50.3).

Por su parte, las variables del controlador del UR son específicas (“UR_T2O” y “UR_O2T”) y no se encuentran dentro de las variables preinstaladas en el TIA Portal, por lo que accediendo de nuevo a la guía de PROFINET [19] encontramos el fichero “UR_datastruct.udt”. Para importar este archivo debemos ir a la pestaña “Fuentes externas” y dentro de la pestaña seleccionamos la opción “Agregar nuevo archivo externo” (Figura 17, siguiente página) [22].

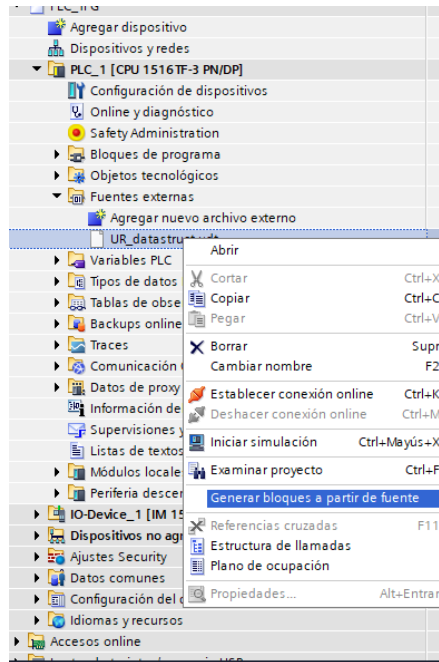


Figura 17: ruta para importar el archivo .udt

Una vez hemos importado el fichero nos aparecerá justo debajo. Para finalizar la importación, debemos pinchar en el archivo con el clic derecho y seleccionar “Generar bloques a partir de fuente” (Figura 17).

Dentro del apartado de soporte técnico de Siemens podemos encontrar el tutorial original para importar los ‘data structs’ a TIA Portal. [22]



2.7.2.-Diseño del programa de control del PLC con TIA Portal

Una vez diseñada la red PROFINET de los diferentes controladores, se debe crear el programa capaz de controlar el correcto funcionamiento de la estación [23]. Para la programación del PLC seguiremos utilizando TIA Portal y utilizaremos el lenguaje de programación por defecto del TIA Portal, el KOP, el cual está basado en "LADDER" [24].

A lo largo de este apartado, se explicarán todos los elementos que componen el programa de control, desde las variables utilizadas hasta el diseño de las sentencias de las funciones.

Variables del sistema

En primer lugar, debemos hablar de las variables utilizadas. En la siguiente tabla podemos ver todas las variables de entrada, salida y variables de memoria utilizadas en el programa, junto con su dirección de memoria y una breve descripción de su propósito (aunque se ha buscado siempre que las variables sean auto explicativas)

Nombre	Tipo de dato	Dirección de memoria	Descripción
Pretope_Coche	Bool	%I0.0	Pretope de la cinta de coches
Cinta_EntradaCoches	Bool	%Q0.0	Motor de la cinta de entrada de coches
Tope_Coche	Bool	%I0.1	Tope de la cinta de coches
Tope_ZonaTrabajo	Bool	%I0.2	Tope de ZT de la cinta de coches
Tope_Salida	Bool	%I0.3	Tope de salida de la cinta de coches
Seta_EM1	Bool	%I0.4	Seta de Emergencia 1 (Armario de control)
Boton Rearme	Bool	%I0.6	Botón de rearme (Armario de control)
EV1	Bool	%Q0.1	Electroválvula 1
EV2	Bool	%Q0.2	Electroválvula 2
EV3	Bool	%Q0.3	Electroválvula 3
EV4	Bool	%Q0.4	Electroválvula 4
Eneable_sema	Bool	%I0.7	Señal "Eneable" del semáforo
Rojo_sema	Bool	%Q0.6	Control luz roja del semáforo
Azul_sema	Bool	%Q0.7	Control luz azul del semáforo
Verde_sema	Bool	%Q1.0	Control luz verde del semáforo
Ambar_sema	Bool	%Q1.1	Control luz amarilla del semáforo
Seta_EM2	Bool	%I52.0	Seta de Emergencia 2 (Mesa de control)
Seta_EM3	Bool	%I52.1	Seta de Emergencia 3 (Entorno de la estación)
Clock_2Hz	Bool	%M333.3	Marca reloj 2Hz
Clock_1.25Hz	Bool	%M333.4	Marca reloj 1,25Hz
Clock_1Hz	Bool	%M333.5	Marca reloj 1Hz
Clock_0.625Hz	Bool	%M333.6	Marca reloj 0.625Hz
Clock_0.5Hz	Bool	%M333.7	Marca reloj 0.5Hz
URI	"UR_T2O"	%I53.0	Comunicación Robot to PLC



Nombre	Tipo de dato	dirección de memoria	Descripción
URO	"UR_O2T"	%Q51.0	Comunicación PLC to Robot
Boton Fin de Trabajo	Bool	%I52.3	Botón Fin de Trabajo (Mesa de control)
Tope_Cubierta	Bool	%I52.5	Tope cinta de cubiertas
Boton Pausa	Bool	%I52.2	Botón de Pausa (Mesa de control)
Boton Reanudar	Bool	%I52.6	Botón de Reanudar (Mesa de control)
LED_Marcha	Bool	%Q1.5	LED de botón de Marcha
LED_FinCiclo	Bool	%Q50.2	LED de botón de Fin de trabajo
Boton Marcha	Bool	%I1.0	Botón de Marcha (Armario de control)
Cinta_salida_coche	Bool	%Q0.5	Motor de la cinta de salida de coches
Sensor_proximidad	Bool	%Q50.3	Señal del sensor de proximidad
Cinta_Pretope_to_Toipe	Bool	%Q50.4	Motor de cinta del pretope al tope de la cinta de coches
Cinta_Toipe_to_ZT	Bool	%Q50.5	Motor de la cinta del tope a la ZT de la cinta de coches
Estado Normal	Bool	%M100.0	Variable de memoria del estado normal
Estado Emergencia	Bool	%M100.1	Variable de memoria del estado emergencia
Estado Error	Bool	%M100.2	Variable de memoria del estado error
Estado saturacion	Bool	%M100.3	Variable de memoria del estado saturación
Estado Rearme	Bool	%M100.4	Variable de memoria del estado rearme
Estado Pausa	Bool	%M100.5	Variable de memoria del estado pausa
Pieza cogida	Bool	%M100.6	Variable de memoria que indica pieza cogida
Pieza dejada	Bool	%M100.7	Variable de memoria que indica pieza dejada
Embolo libre	Bool	%M101.0	Variable de memoria que indica libre émbolo
Comienzo CT	Bool	%M101.1	Variable de memoria de inicio de ciclo de trabajo
Fin CT	Bool	%M101.2	Variable de memoria de fin de ciclo de trabajo
Coche procesado	Bool	%M101.3	Variable de memoria de coche procesado
Fin de ciclo del Robot	Bool	%M101.5	Variable de memoria de fin de ciclo de trabajo del robot
M_Coche_en_Toipe	Bool	%M101.6	Variable de memoria coche en tope
M_Coche_en_Pretope	Bool	%M101.7	Variable de memoria coche en pretope
M_Coche_en_ZT	Bool	%M102.0	Variable de memoria coche en tope de ZT
M_Coche_en_salida	Bool	%M102.1	Variable de memoria coche en tope de salida
M_Pieza_en_Toipe	Bool	%M102.2	Variable de memoria pieza en tope
Solic. Recoger Pieza	Bool	%M102.3	Solicitud de recoger pieza
Solic. Dejar Pieza	Bool	%M102.4	Solicitud de dejada de pieza
Solic. Libre Embolo	Bool	%M102.5	Solicitud de libertad de movimiento de émbolo
Solic Reanudar	Bool	%M102.6	Solicitud de reanudar ciclo de trabajo
Solic. Parada de Emergencia	Bool	%M102.7	Solicitud de realizar parada de emergencia
Solic. Parada de Seguridad	Bool	%M103.0	Solicitud de realizar parada de seguridad
Inicio de ciclo del Robot	Bool	%M101.4	Solicitud de inicio de ciclo del robot

Tabla 1: Variables del programa



Descripción del programa

Con las variables ya definidas, pasaremos a explicar el programa de control.

En TIA Portal, los programas se ejecutan desde las OB o Bloques de Organización, este bloque sería el equivalente a la función “main” del lenguaje C. Dentro de esta OB, se encuentra toda la programación necesaria para el correcto funcionamiento del PLC, ya sea mediante FB (Bloques de Función), mediante el uso de eventos o bien directamente con el código del programa.

Para aplicaciones industriales de estas dimensiones y dentro de la normativa de muchas empresas, es obligatorio el uso de FBs para la organización del código en diferentes bloques con el mismo propósito, incluso hay normas sobre la enumeración de estas.

Para el diseño del programa de esta estación se han utilizado un total de 7 FBs con una numeración determinada de tal manera que en el Bloque de Organización solo se encuentran llamadas a estos Bloques de Función y la asignación de variables globales necesarias a cada una de estas FB. A continuación, se hará una explicación detallada de cada una de las FB y el apartado finalizará con la explicación de la OB y la asignación de las variables mencionadas con anterioridad a cada una de las FB. Se ha buscado realizar una programación lo más sencilla posible para que una persona que no está familiarizada con este tipo de dispositivos sea capaz de comprender el funcionamiento de este y sea capaz de replicarlo.

Como norma general, en la descripción de cada una de las FB hablaremos primero de las variables locales de esta FB y posteriormente se hará una detallada explicación de cada uno de los segmentos que configuran este Bloque de Función.

FB2: CONTROL DEL SEMAFORO

Este primer FB del que vamos a hablar se llama “Control del Semáforo”, y como su nombre indica, se encargará de la enviar las señales a la columna de señalización de la estación con el fin de indicar el estado en el que se encuentra la estación.

En el Anexo III: “Componentes de la estación” Se encuentra una descripción más detallada de esta columna de señalización. Este semáforo cuenta con 4 colores (Rojo, Azul, Verde y Amarillo) con el siguiente código de colores:

- **Rojo:** Error del robot (Parpadeo a 2Hz)
- **Azul:** Saturación de la línea (Parpadeo a 1Hz)
- **Amarillo:** Parada de emergencia
- **Verde:** Funcionamiento correcto

El semáforo nos devuelve una señal de diagnóstico ‘Eneable’ que indica que la alimentación llega correctamente y que no hay ninguna luz fundida.

En la siguiente figura (Figura 18, siguiente página) se puede ver el conjunto de las variables locales utilizadas en el bloque. Están divididas en 3 grupos principales (Entradas, Salidas, Entrada-Salida).



Control_Semaforo				
	Nombre	Tipo de datos	Valor predet.	Remanencia
1	Input			
2	Error	Bool	false	No remane...
3	Emergencia	Bool	false	No remane...
4	Saturacion	Bool	false	No remane...
5	Eneable	Bool	false	No remane...
6	Output			
7	S_Rojo	Bool	false	No remane...
8	S_Azul	Bool	false	No remane...
9	S_Amarillo	Bool	false	No remane...
10	S_Verde	Bool	false	No remane...
11	InOut			
12	Normal	Bool	false	No remane...

Figura 18: Tabla variables FB2

En esta figura, podemos observar que cómo variables de entrada recibimos los posibles estados en los que se puede encontrar la estación.

En las variables de salida se encuentran los colores del semáforo.

Como variables de Entrada-Salida se encuentra el estado Normal ya que se utilizará tanto de entrada como de salida.

- **SEGMENTO 1: Control de la luz Roja. (Figura 19)**

Como su nombre indica, este segmento se encarga de controlar las condiciones que se deben dar para que se active la luz roja.

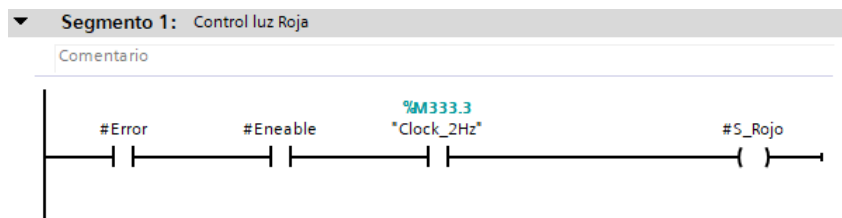


Figura 19: Segmento 1 de FB2

- **SEGMENTO 2: Control de la luz Amarilla. (Figura 20)**

En este segundo segmento, se controla las condiciones necesarias para activar la luz amarilla.

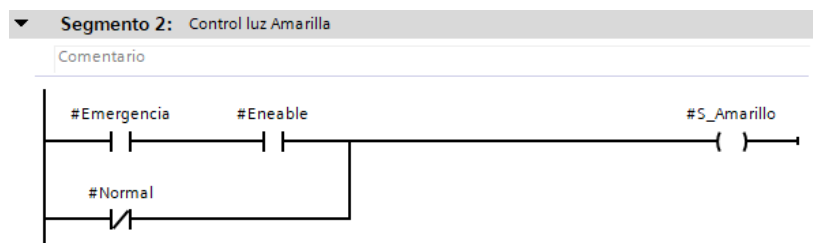


Figura 20: Segmento 2 de FB2



- **SEGMENTO 3: Control de luz Azul. (Figura 21)**

En este segmento se controla la condición necesaria para activar el parpadeo de la luz azul.

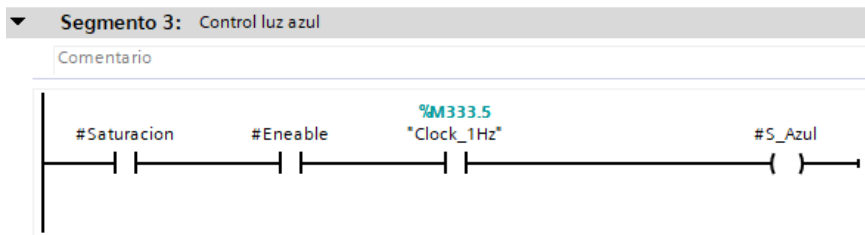


Figura 21: Segmento 3 de FB2

- **SEGMENTO4: Control de la luz Verde. (Figura 22)**

Por último, este segmento controla la condición necesaria para que se encienda la luz verde.

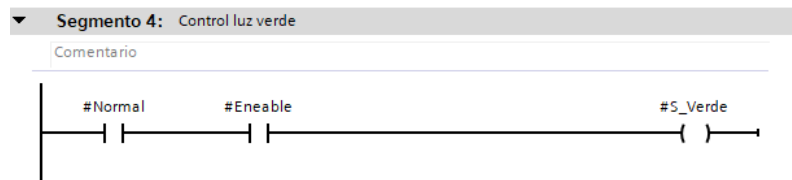


Figura 22: Segmento 4 de FB2

FB3: CONTROL DE ESTADO DE TRABAJO

Este Bloque de Función denominado “Control de estado de trabajo” se encarga de guardar las variables de los topes en las variables de memoria correspondientes. Este bloque ha sido diseñado para evitar el uso de lógica negativa en el resto de FBs y para que el PLC almacene en memoria el estado de todos los topes de la estación.

Este bloque utiliza únicamente variables de entrada y variables de salida (Figura 23).

Control de estado de trabajo				
	Nombre	Tipo de datos	Valor predet.	Remanencia
1	Input			
2	Tope_Coche	Bool	false	No remane...
3	Pretope_Coche	Bool	false	No remane...
4	Tope_ZonaTrabajo	Bool	false	No remane...
5	Tope_salida	Bool	false	No remane...
6	Tope_Pieza	Bool	false	No remane...
7	Output			
8	Saturacion	Bool	false	No remane...
9	Coche en Tope	Bool	false	No remane...
10	Coche en ZT	Bool	false	No remane...
11	Salida Coche	Bool	false	No remane...
12	Pieza en Tope	Bool	false	No remane...

Figura 23: Tabla de variables de FB3

Como podemos observar en la figura anterior, las variables de entrada hacen referencia a la activación de los diferentes topes de la estación, y las variables de salida, hacen referencia al



estado del ciclo de trabajo. Los topes funcionan estando “normalmente cerrados” (NC) y cuando detectan un objeto el circuito se abre. Por ello, debemos utilizar lógica negativa.

- **SEGMENTO 1: Control de coche en tope y pretope. (Figura 24)**

Como el nombre de este segmento indica, este segmento se encarga de observar la presencia de coches en el tope y en el pretope y señalar saturación en caso de que se den las condiciones.

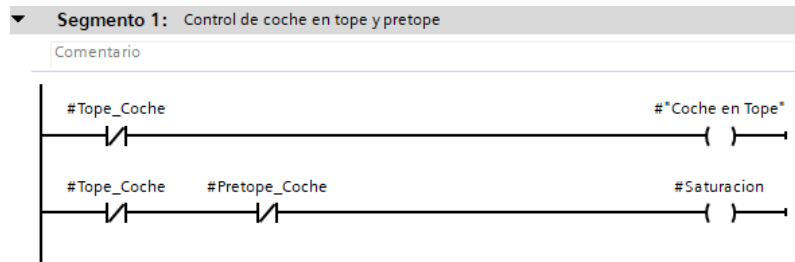


Figura 24: Segmento 1 de FB3

- **SEGMENTO 2: Control de coche en zona de trabajo. (Figura 25)**

Este segmento se encarga de indicar si hay un coche en la zona de trabajo, o no.

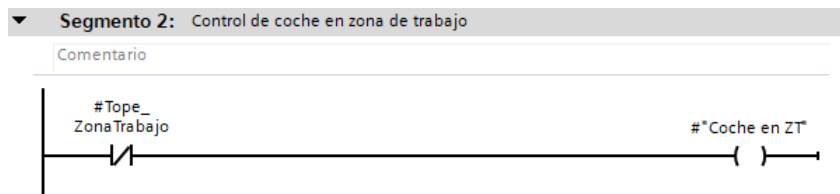


Figura 25: Segmento 2 de FB3

- **SEGMENTO 3: Control de salida de coche. (Figura 26)**

Este segmento se encarga de indicar si el coche ya procesado ha llegado al fin de la estación, rumbo a la siguiente estación de trabajo.

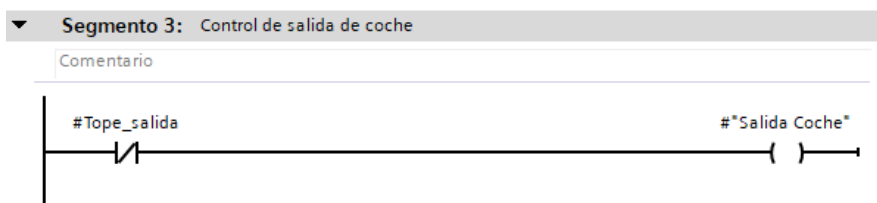


Figura 26: Segmento 3 de FB3



- **SEGMENTO 4: Control de cubierta en tope lista para empezar el Ciclo de Trabajo. (Figura 27)**

Este último segmento indica si hay una cubierta en el tope de las cubiertas.

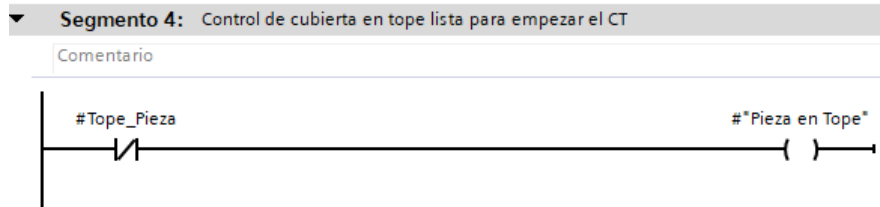


Figura 27: Segmento 4 de FB3

FB4: CONTROL NEUMÁTICO

Este bloque se encarga del control de las 4 electroválvulas que componen el sistema neumático en función del estado en el que deseamos que se encuentre la pinza.

Este bloque vuelve a tener los 3 tipos de variables utilizadas en la FB2. (Figura 28)

Control neumático					
	Nombre	Tipo de datos	Valor predet.	Remanencia	
1	Input				
2	Solic. Recogida de Pieza	Bool	false	No remane...	
3	Solic. Dejada de Pieza	Bool	false	No remane...	
4	Solic. libertad de embolo	Bool	false	No remane...	
5	Output				
6	EV1(A)	Bool	false	No remane...	
7	EV2(B)	Bool	false	No remane...	
8	EV3(C)	Bool	false	No remane...	
9	EV4(D)	Bool	false	No remane...	
10	InOut				
11	Pieza cogida	Bool	false	No remane...	
12	Pieza dejada	Bool	false	No remane...	
13	Embolo libre	Bool	false	No remane...	

Figura 28: Tabla de variables de la FB4

Como variables de entrada tenemos las solicitudes de tener la pinza en un estado determinado, como salida tenemos las electroválvulas y como variables de entrada-salida, los estados en los que se encuentra la pinza. Este último tipo de variable es el encargado de informar al resto de Bloques de Función del programa el estado de la pinza.



- **SEGMENTO 1: Solicitud de émbolo libre. (Figura 29)**

Este segmento gestiona la petición para dejar el émbolo libre con el fin de preparar el efecto ventosa de la pinza.

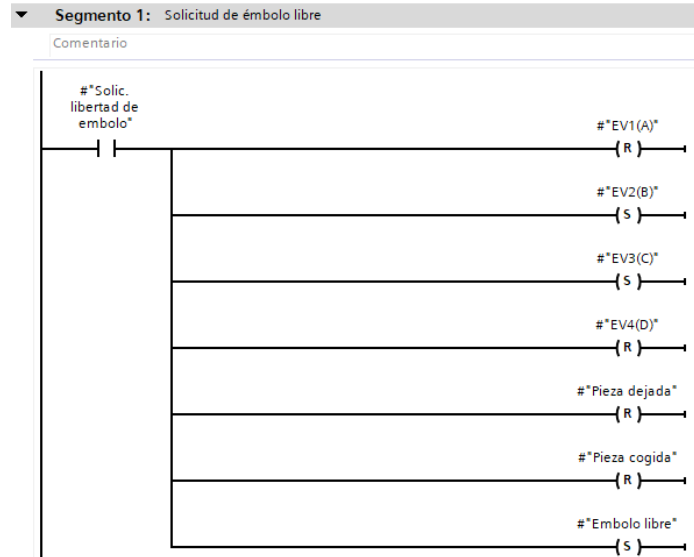


Figura 29: Segmento 1 de FB4

- **SEGMENTO 2: Solicitud de agarre de pieza. (Figura 30)**

Este segmento siempre irá después de la sentencia de libertad del émbolo que permite el agarre de la pieza. El objetivo es bloquear el émbolo para lograr el efecto ventosa y, por lo tanto, la cogida de la pieza.

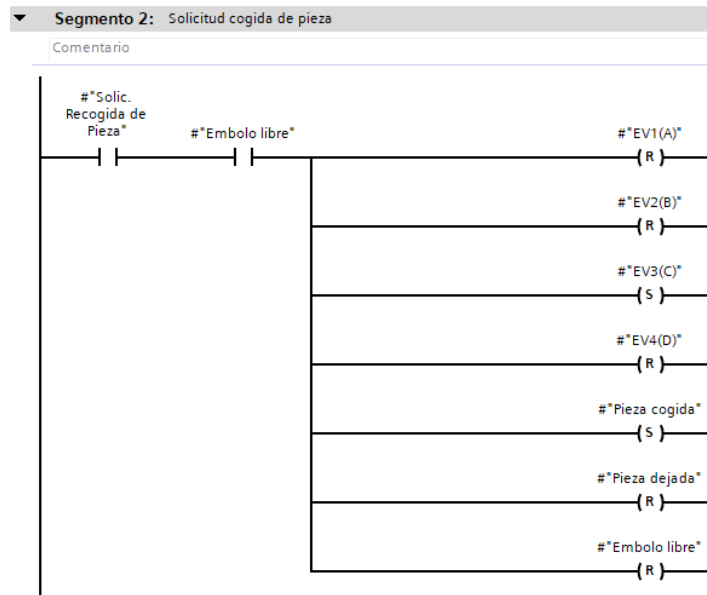


Figura 30: Segmento 2 de FB4



- **SEGMENTO3: Solicitud de dejada de pieza. (Figura 31)**

Este segmento se encarga de gestionar la dejada de pieza. Esta dejada de pieza se consigue abriendo la cámara de aire del extremo de la pinza, deshaciendo el efecto ventosa, y haciendo que el émbolo se expanda para que posteriormente se pueda volver a dejar el émbolo libre.

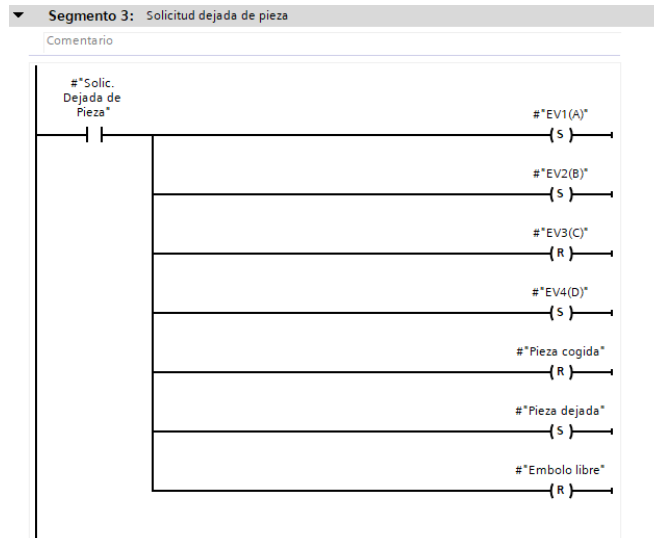


Figura 31: Segmento 3 de FB4

FB5: CONTROL DE ESTADO DE PAUSA Y REARME

Este bloque es el encargado de informar al sistema de control del estado de pausa de seguridad y los estados de emergencia y error. Una vez nos encontramos en este estado, este bloque también administra el rearme para que el sistema pueda volver a la normalidad.

En cuanto a las variables, tenemos variables distribuidas en los 3 grupos generales. (Figura 32)

Control de Estado de Pausa y Rearme				
	Nombre	Tipo de datos	Valor predet.	Remanencia
1	Input			
2	Boton rearme	Bool	false	No remane...
3	Estado error	Bool	false	No remane...
4	Boton Marcha	Bool	false	No remane...
5	Output			
6	Solic. reanudacion	Bool	false	No remane...
7	LED_Reanudar	Bool	false	No remane...
8	InOut			
9	Estado Rearme	Bool	false	No remane...
10	Estado Emergencia	Bool	false	No remane...
11	Estado Normal	Bool	false	No remane...

Figura 32: Tabla de variables de la FB5

En el grupo de entradas encontramos los 3 botones necesarios para el rearme y el control de la pausa de seguridad, y también disponemos de la entrada procedente de la comunicación con el UR (Bloque de función 50) que nos indica que hay un error en el robot.



En el grupo de salidas, tenemos las diferentes solicitudes para reanudar el estado normal de la estación y el LED que indica que el sistema está rearmado y listo para que se pulse el botón de marcha.

En el último bloque de entrada-salida encontramos los diferentes estados que se pueden encontrar la estación relacionados con la parada y reanudación de la estación.

- **SEGMENTO 1: Solicitud de parada. (Figura 33)**

Este segmento es el encargado de resetear el estado de rearme y el estado normal cuando se produce una anomalía (error en el robot o emergencia en la estación).

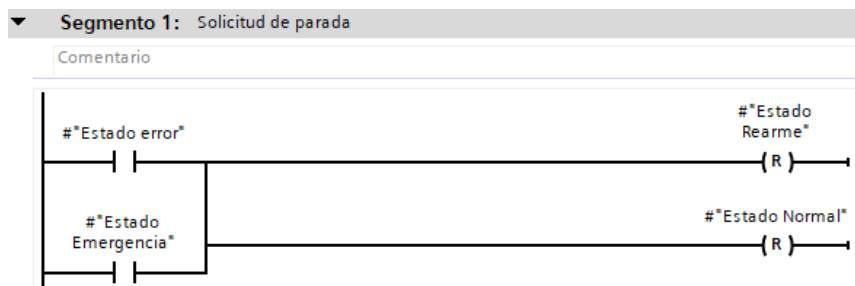


Figura 33: Segmento 1 de FB5

- **SEGMENTO 2: Control de rearme y reanudación de la estación. (Figura 34)**

Este segundo bloque es el encargado de llevar a cabo el rearme de la estación tras un error robot o emergencia y la reanudación de la rutina normal de la estación una vez esta ya haya sido rearmada.

También se encarga del parpadeo del LED cuando sea necesario pulsar el botón de marcha.

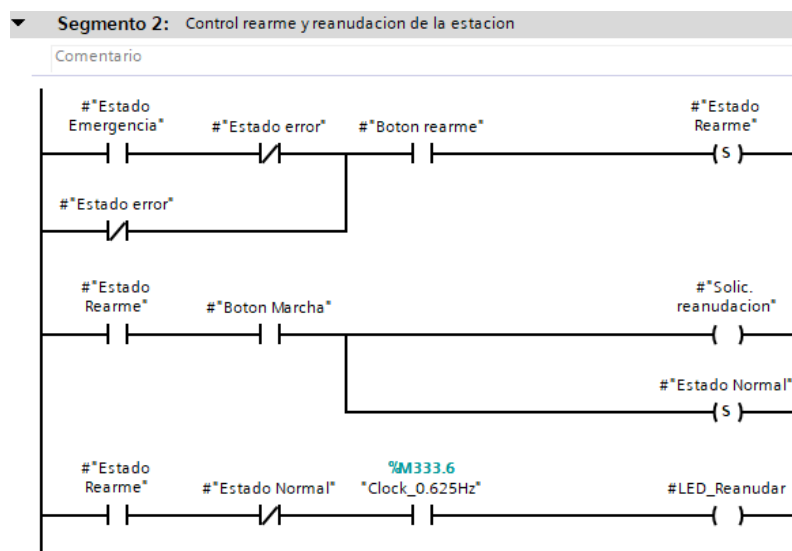


Figura 34: Segmento 2 de FB5



FB50: COMUNICACIÓN CON EL UR

Este segmento, ya con una numeración específica recoge todas las sentencias del programa dónde hay una comunicación directa con el controlador del cobot. Para esta comunicación se utilizará los registros de entrada de bits y los registros de salida de bits descritos en el Anexo II: “Comunicación PROFINET del controlador”

Los segmentos de este bloque sirven para organizar estas sentencias en grupos con el mismo objetivo, en lugar de diferenciar entre emisión y recepción de datos.

Las variables de este grupo son de los 3 tipos generales (Figura 35)

Comunicacion con UR				
	Nombre	Tipo de datos	Valor predet.	Remanencia
1	Input			
2	Llegada a aprox P10	Bool	false	No remane...
3	Posicion de Rebuclaje	Bool	false	No remane...
4	Llegada a P20	Bool	false	No remane...
5	Error del robot	Bool	false	No remane...
6	Llegada a P10	Bool	false	No remane...
7	Agarre completado	Bool	false	No remane...
8	Dejada completada	Bool	false	No remane...
9	Solicitud de parada se...	Bool	false	No remane...
10	Boton reanudar	Bool	false	No remane...
11	Boton de pausa	Bool	false	No remane...
12	Fin de ciclo programa	Bool	false	No remane...
13	Comienza ciclo robot	Bool	false	No remane...
14	Solic. reanudacion	Bool	false	No remane...
15	Output			
16	Pieza cogida	Bool	false	No remane...
17	Pieza dejada	Bool	false	No remane...
18	Solic. movimiento em...	Bool	false	No remane...
19	Solic. recoger pieza	Bool	false	No remane...
20	Solic. dejar pieza	Bool	false	No remane...
21	Solic. parada de emer...	Bool	false	No remane...
22	Solic. pausa Robot	Bool	false	No remane...
23	Inicio de ciclo	Bool	false	No remane...
24	Continua rutina	Bool	false	No remane...
25	InOut			
26	Estado pausa	Bool	false	No remane...
27	Estado Normal	Bool	false	No remane...
28	Estado error robot	Bool	false	No remane...
29	Estado de emergencia	Bool	false	No remane...
30	Solic. parada de segur...	Bool	false	No remane...
31	Fin de ciclo robot	Bool	false	No remane...

Figura 35: Tabla de variables de FB50

En la entrada tenemos tanto algunos bits de entrada del UR, como botones y valores recibidos de otros bloques.

Como salidas, encontramos bits de salida a los registros del UR y solicitudes y sentencias que se activan como respuesta a la llegada de ciertas señales de entrada del robot.

En el bloque de entrada-salida encontramos los estados de la estación y la solicitud de parada de seguridad.



- **SEGMENTO 1: Solicitud de inicio de rutina del robot. (Figura 36)**

En este primer segmento se notifica al robot que ya puede iniciar su rutina una vez se dan las condiciones necesarias en la estación.

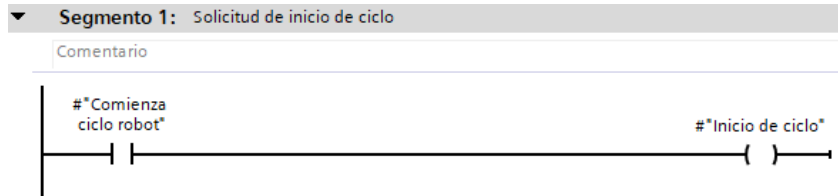


Figura 36: Segmento 1 de FB50

- **SEGMENTO 2: Comunicación para la recogida de pieza en el punto 10. (Figura 37)**

Este bloque cuenta con la comunicación que se realiza entre PLC y robot cuando se haya alcanzado las proximidades del punto 10 y se desee recoger la pieza.

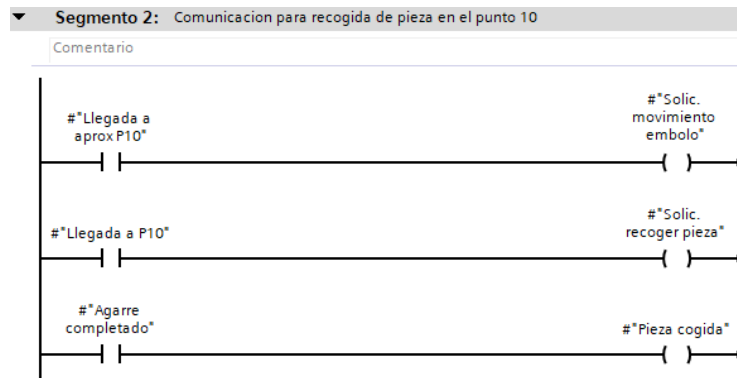


Figura 37: Segmento 2 de FB50

- **SEGMENTO 3: Comunicación para la dejada de pieza en el punto 20. (Figura 38)**

Este tercer bloque es semejante al primero, recoge toda la comunicación entre el PLC y el controlador del UR una vez se haya alcanzado el punto 20 y haya que dejar la pieza.

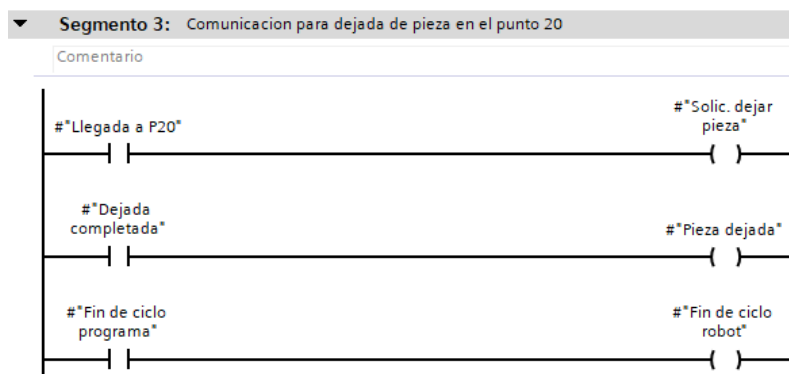


Figura 38: Segmento 3 de FB50



- **SEGMENTO 4: Comunicación para la parada segura: pausa. (Figura 39)**

En este segmento se recoge las sentencias de comunicación para solicitar una parada segura cuando se pulsa el botón de pausa y su posterior reanudación de la rutina mediante el botón de reanudar una vez la parada de seguridad se ha producido.

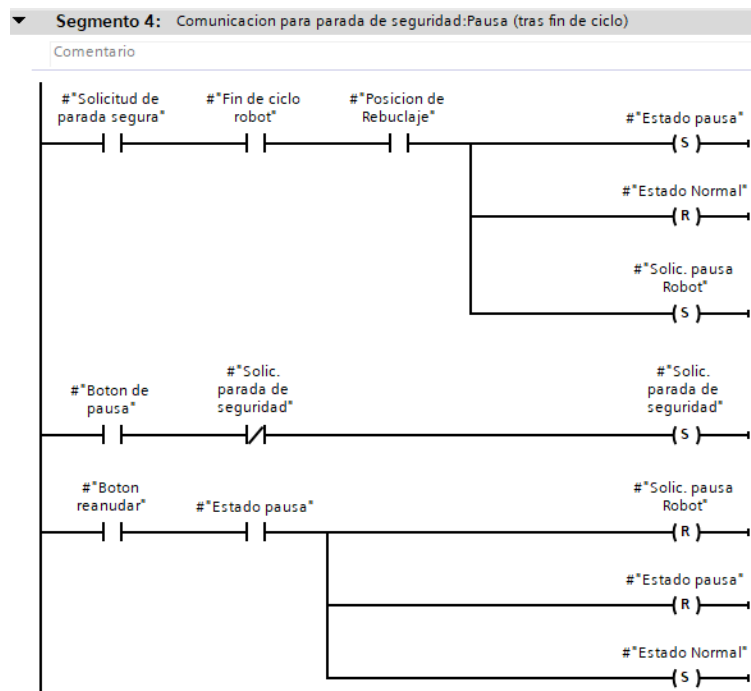


Figura 39: Segmento 4 de FB50

- **SEGMENTO 5: Comunicación en caso de fallo del robot. (Figura 40)**

Este sencillo segmento guarda en una variable de memoria el bit recibido por el controlador del UR sobre la existencia de algún error en el UR.

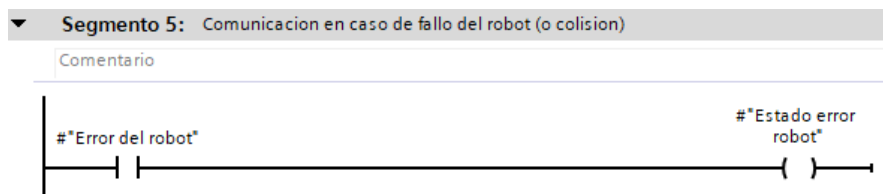


Figura 40: Segmento 5 de FB50

- **SEGMENTO 6: Comunicación en caso de parada de emergencia. (Figura 41)**

Este último segmento recoge la sentencia que obliga al UR a parar en caso de que haya un estado de emergencia (invasión detectada por el sensor de proximidad o se haya pulsado alguna seta



de emergencia) y su posterior solicitud de reanudar rutina cuando el sistema vuelve a la normalidad.

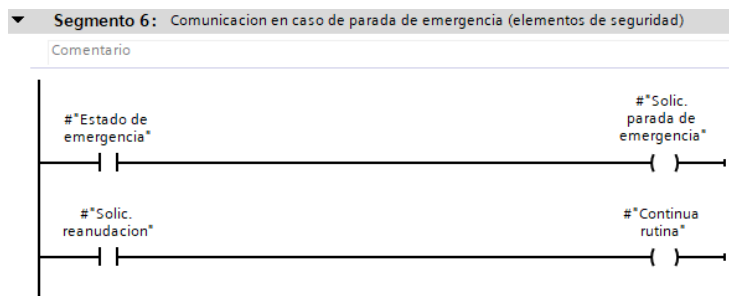


Figura 41: Segmento 6 de FB50

FB100: CONTROL RUTINA DE LA ESTACIÓN

Este bloque con la numeración específica de la normativa del grupo Renault con respecto a la numeración de FBs hace referencia a que es el Bloque de Función principal para la ejecución de la rutina. Este, se encarga de gestionar el resto de las FBs para llevar a cabo el ciclo de trabajo de la estación.

En cuanto a las variables, este bloque, aun siendo el más importante, no es el que más variables tiene; no obstante, cuenta con variables de 3 tipos. (Figura 42)

Control rutina de la estación				
	Nombre	Tipo de datos	Valor predet.	Remanencia
1	▼ Input			
2	Coche listo	Bool	false	No remane...
3	Pieza lista	Bool	false	No remane...
4	Coche en Pretope	Bool	false	No remane...
5	Coche en ZT	Bool	false	No remane...
6	Estado Normal	Bool	false	No remane...
7	Fin de Rutina del Robot	Bool	false	No remane...
8	Pulsador fin de Ciclo d...	Bool	false	No remane...
9	▼ Output			
10	Inicia Ciclo de Robot	Bool	false	No remane...
11	Mover_Cinta_Salida	Bool	false	No remane...
12	LED_Fin_Ciclo	Bool	false	No remane...
13	Mover_Cinta_Tope_ZT	Bool	false	No remane...
14	Mover_Cinta_Pretope...	Bool	false	No remane...
15	Mover_Cinta_Entrada...	Bool	false	No remane...
16	▼ InOut			
17	Comienzo CT	Bool	false	No remane...
18	Coche procesado	Bool	true	No remane...
19	Fin de Ciclo de Trabajo	Bool	false	No remane...
20	Coche salida	Bool	false	No remane...

Figura 42: Tabla de variables de FB100

En las variables de entrada, podemos observar que cuenta con muchas de las variables de la FB3 “Control de estado de ciclo de trabajo”, junto con el pulsador de fin de ciclo para indicar que la pieza ya ha sido instalada.

En cuanto al grupo de salidas, este Bloque de Función se encarga de mover la cinta transportadora de coches dentro del perímetro de la estación, también encontramos el LED que indica que se puede pulsar el botón de fin de ciclo y la solicitud de iniciar el ciclo de trabajo del robot.



En el último grupo, el de variables entrada-salida, tenemos el grupo de variables de la que necesitamos que actúen tanto como entrada, como salida para el correcto funcionamiento de este bloque.

- **SEGMENTO 1: Control de condiciones para iniciar el Ciclo de Trabajo. (Figura 43)**

Este segmento administra el movimiento de los coches a través de las cintas desde la entrada a la estación hasta la zona de los topes “pre-zona” de trabajo. El movimiento está regido por el estado de la estación y la presencia (o ausencia) de coches en el pretope y/o en el tope.

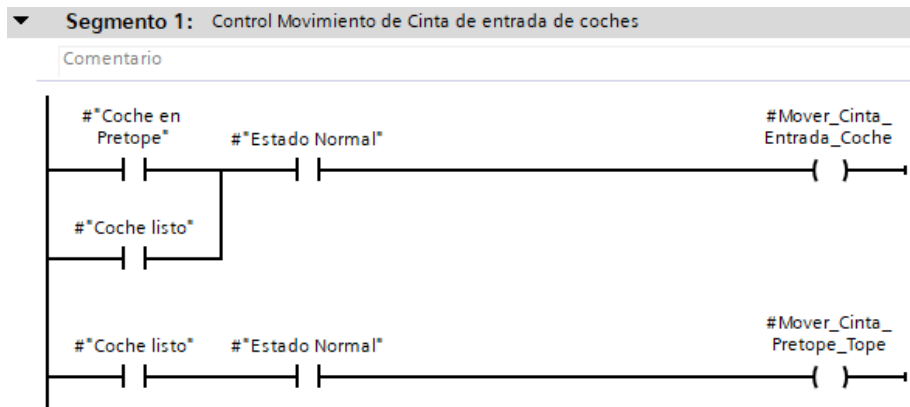


Figura 43: Segmento 1 de FB100

- **SEGMENTO 2: Control de condiciones para inicial el Ciclo de Trabajo (Figura 44)**

Este segmento contiene las condiciones que se deben dar para que se pueda iniciar el Ciclo de Trabajo en correctas condiciones. Para ello, es necesario que no estemos ni en parada ni en Emergencia/Error, que haya tanto un coche como una cubierta en sus topes correspondientes y, por último, es necesario que el anterior coche, ya procesado, haya salido de la estación.

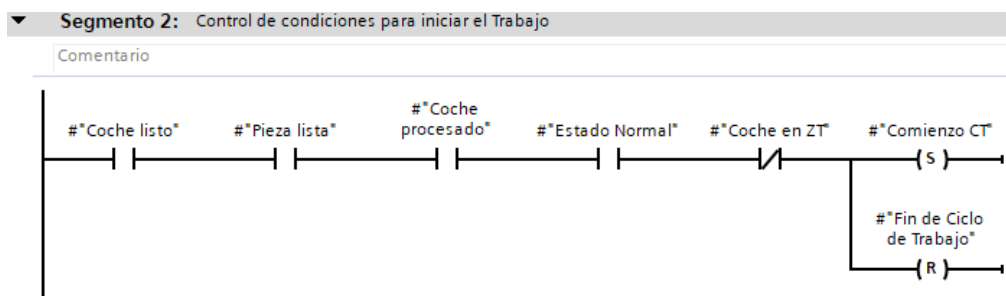


Figura 44: Segmento 2 de FB100



- **SEGMENTO 3: Movimiento del coche desde el tope hasta la ZT. (Figura 45)**

El segundo segmento se encarga únicamente de controlar la cinta transportadora para mover el coche desde el tope donde estaba esperando, hasta la zona de trabajo para poder instalar la cubierta.

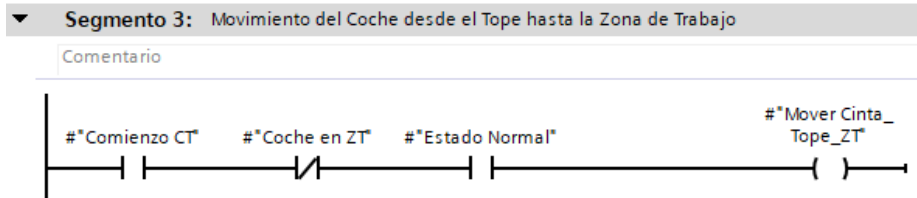


Figura 45: Segmento 3 de FB100

- **SEGMENTO 4: Comienzo del ciclo de trabajo: inicio de la rutina del robot. (Figura 46)**

Este bloque sirve para notificar por una parte al sistema de que el coche ya está en la zona de trabajo y va a empezar el proceso de instalación de cubierta, como para notificar al robot que ya puede empezar su rutina de pick & place de la cubierta.

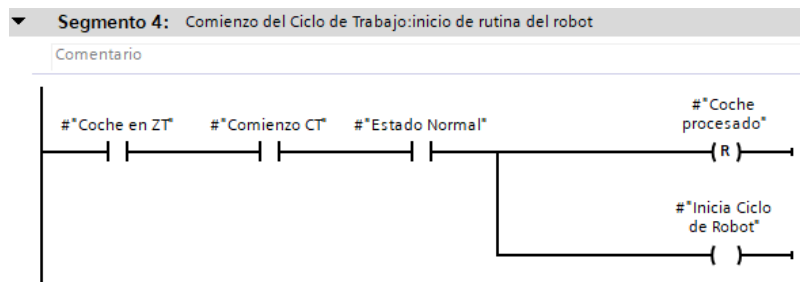


Figura 46: Segmento 4 de FB100

- **SEGMENTO 5: Control de Fin de Ciclo de Trabajo. (Figura 47)**

Este segmento, en parte semejante al anterior, se encarga de notificar al sistema que el robot ya ha terminado el ciclo de trabajo y que ya se ha instalado la cubierta (mediante el pulsador de fin de ciclo de trabajo). También administra el LED que parpadea indicando que ya podemos pulsar el botón de fin de trabajo.

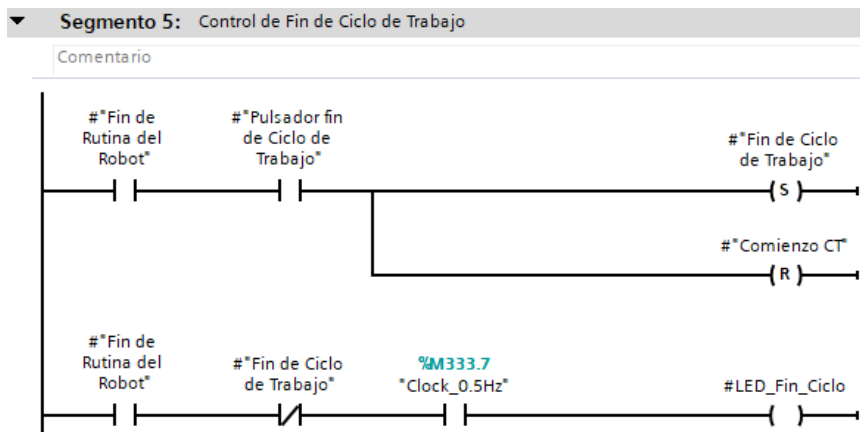


Figura 47: Segmento 5 de FB100



- **SEGMENTO 6: Movimiento del coche desde la zona de trabajo hasta la salida de la estación. (Figura 48)**

Al igual que el segmento 2, este se encarga del movimiento de la cinta transportadora para llevar al coche, ya procesado, desde la zona de trabajo, hasta la salida de la estación.

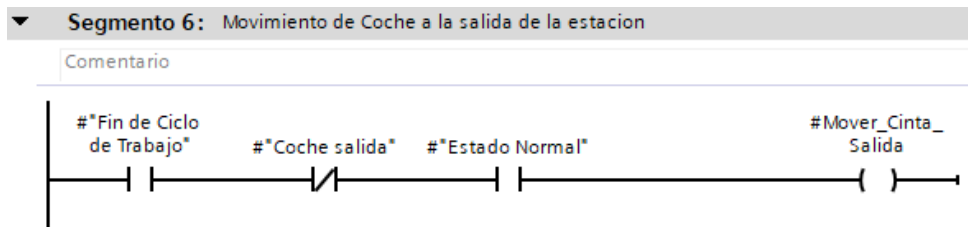


Figura 48: Segmento 6 de FB100

- **SEGMENTO 7: Control de rutina cuando el coche cruza el tope de salida de la estación. (Figura 49)**

Este es el último segmento de la FB100, es el encargado de notificar al sistema que el coche ya ha salido de la estación y se puede realizar el siguiente ciclo de trabajo.

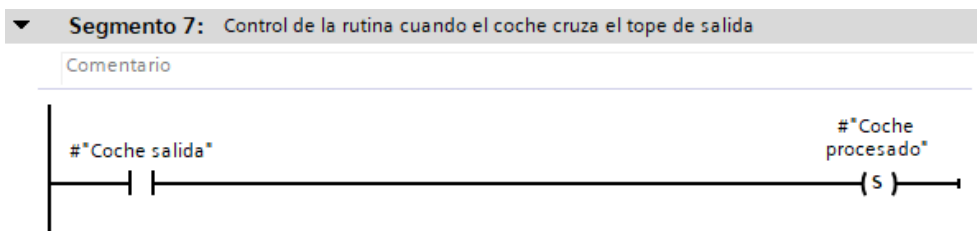


Figura 49: Segmento 7 de FB100

FB150: CONTROL DE EMERGENCIA

Este es el último Bloque de Función del que vamos a hablar, con una numeración específica, este bloque se encarga de notificar al sistema en caso de que alguno de los elementos de seguridad (alguna de las 3 setas de emergencia, o el detector de presencia) se haya disparado.

Las variables de este bloque son las siguientes. (Figura 50)

Control Emergencia				
	Nombre	Tipo de datos	Valor predet.	Remanencia
1	Input			
2	Paro Emergencia 1	Bool	false	No remane...
3	Paro Emergencia 2	Bool	false	No remane...
4	Paro Emergencia 3	Bool	false	No remane...
5	Sensor proximidad	Bool	false	No remane...
6	Output			
7	Estado de emergencia	Bool	false	No remane...

Figura 50: Tabla de variables de la FB150



Teniendo como entrada los elementos de seguridad y como salida, el estado de emergencia de la estación.

- **SEGMENTO 1: Control de seguridad de las setas de emergencia. (Figura 51)**

Este segmento lleva el control de las setas de emergencia. Utilizando lógica negativa, el estado de alarma se dispara si alguna de las 3 setas de emergencia es pulsado.

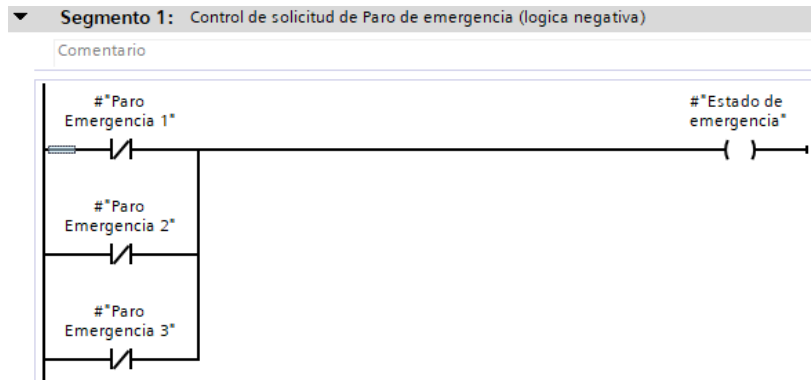


Figura 51: Segmento 1 de FB150

- **SEGMENTO 2: Control del sensor de proximidad. (Figura 52)**

Este segmento se encarga de disparar el estado de emergencia si el sensor de proximidad, situado en una zona no accesible por motivos de seguridad mientras la estación esté en funcionamiento, detecta la presencia de una persona.

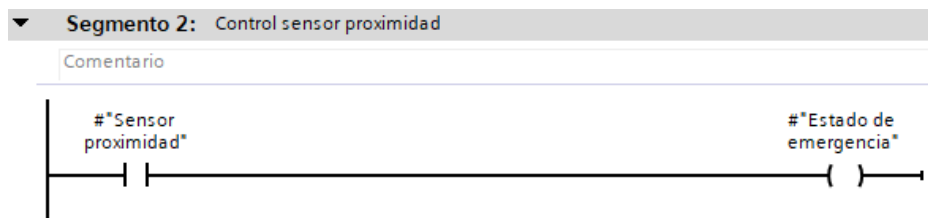


Figura 52: Segmento 2 de FB150

OB1: MAIN

Una vez descritos todos los Bloques de Función, es hora de hablar del Bloque de Organización. El Bloque de Organización (OB) de este programa tiene como tarea principal inicializar todas las FBs descritas con anterioridad y asignar las variables globales a las variables locales correspondientes de cada uno de esos bloques. La organización por segmentos está diseñada para que cada segmento tenga una FB y estos están ordenados numéricamente de menor a mayor.

- **SEGMENTO 1: FB2(CONTROL DE SEMÁFORO). (Figura 53, siguiente página)**

El primer segmento inicializa la FB2 que corresponde al control de la columna de señalización. Como entradas tiene los estados de la estación y como salida las señales para habilitar los colores de la columna de señalización.

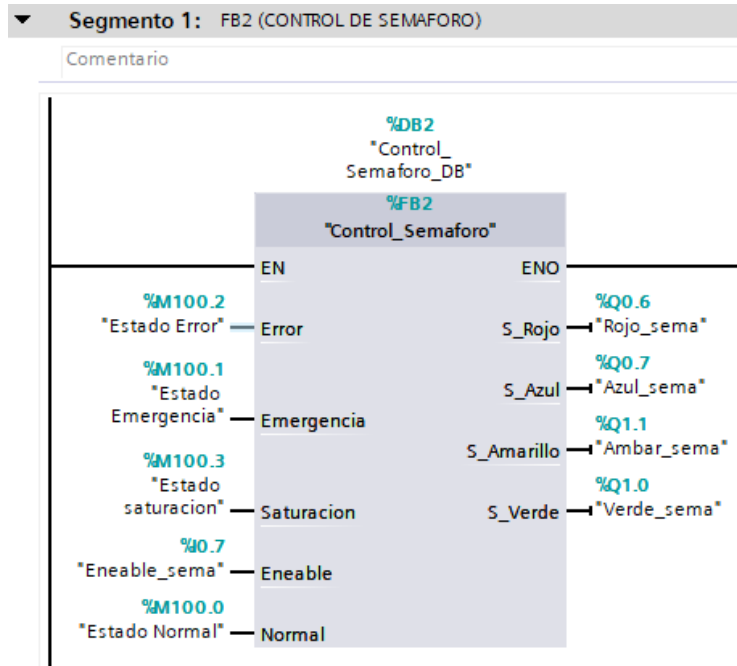


Figura 53: Segmento 1 de la OB1

- **SEGMENTO 2: FB3 (CONTROL DE ESTADO DE TRABAJO). (Figura 54)**

Este segmento contiene la inicialización y asignación de variables de la FB3 correspondiente al control de estado de trabajo en función de la señal recibida por los topes tanto en la manutención correspondiente a los coches, como en la manutención correspondiente a las cubiertas.

Como entradas tiene cada uno de los topes de la estación y como salidas, los estados de la estación y del ciclo de trabajo comentados en la descripción de esta FB.

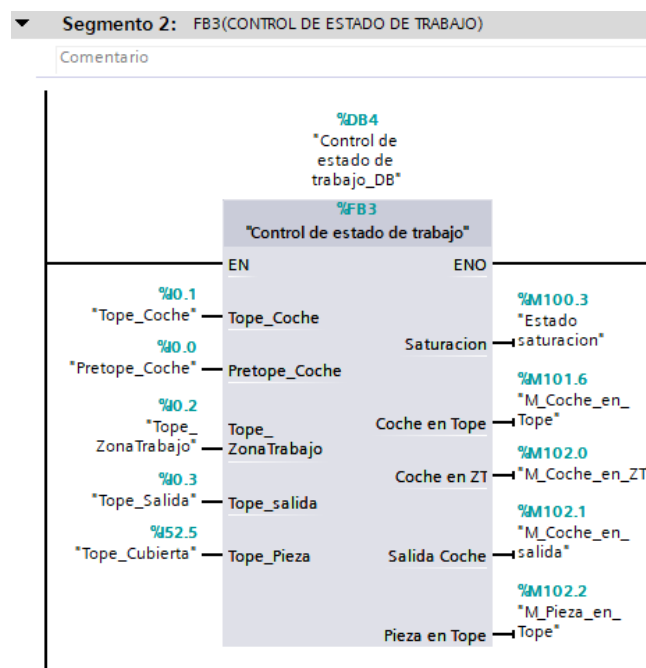


Figura 54: Segmento 2 de la OB1



- **SEGMENTO 3: FB4 (CONTROL NEUMÁTICO). (Figura 55)**

Este segmento contiene la inicialización y asignación de variables de la FB4 correspondiente al control de las electroválvulas para el correcto funcionamiento del sistema neumático.

Como entrada tiene las solicitudes de actuación del sistema neumático y los posibles estados de la pinza. Como salida, tiene las 4 señales correspondientes a las 4 electroválvulas que controlan el sistema neumático.

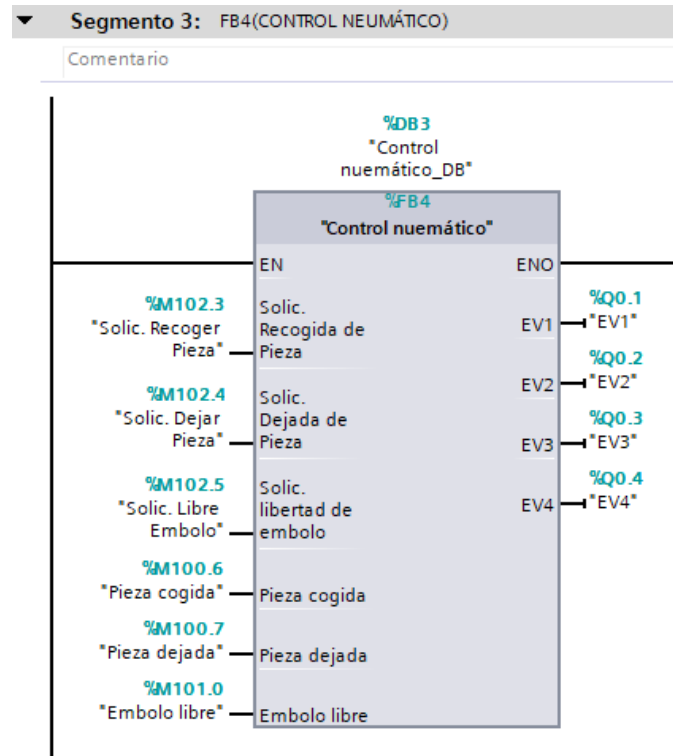


Figura 55: Segmento 3 de la OB1

- **SEGMENTO 4: FB5 (CONTROL DE PAUSA Y REARME). (Figura 56, siguiente página)**

Este segmento se encarga de la inicialización y asignación de variables correspondiente a la FB5, encargada del control de pausa de seguridad y rearme en caso de emergencia/error.

En el lado de las entradas encontramos el botón de rearme y el de marcha y los diferentes estados de la estación correspondientes al control de parada y reanudación de la estación.

Al otro lado, encontramos la solicitud para reanudar la rutina de la estación y el LED que nos indica que el sistema está rearmado y podemos pulsar el botón Marcha para volver al estado normal.

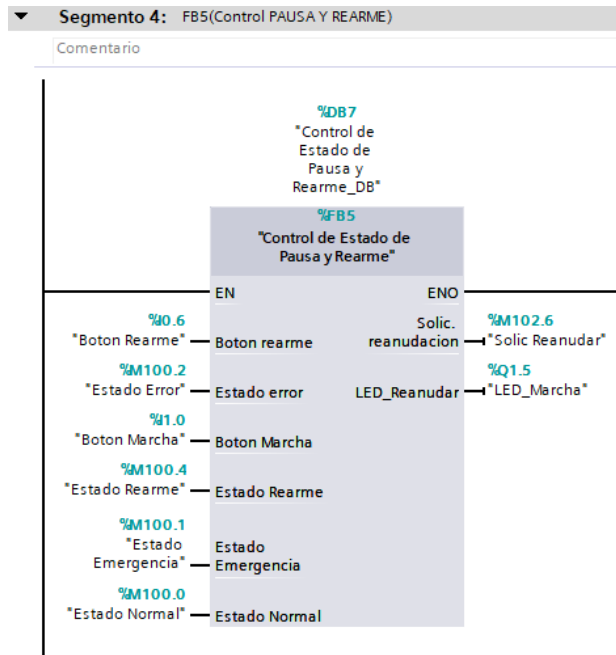


Figura 56: Segmento 4 de la OB1

- **SEGMENTO 5: FB50 (CONTROL COMUNICACIÓN CON UR). (Figura 57)**

Este segmento se encarga de la inicialización y asignación de variables para la FB50, la encargada de la comunicación con el controlador del UR. A diferencia del resto de bloques, este utiliza las variables especiales para acceder a los registros de entrada y salida del controlador.

En la siguiente tabla se puede ver la asignación de las variables a los registros, tanto de entrada como de salida del controlador.

Registro	URI	URO
0	Llegada al punto Aprox P10	Cubierta cogida
1	Robot en posición de Rebuclaje	Cubierta dejada
2	Llegada al punto P20	-
3	Llegada al punto P10	-
15	-	Parada de emergencia
20	Fin de Ciclo del programa	Pausa de seguridad
30	-	Inicio de ciclo
31	-	Solicitud reanudar rutina
ST	Error en el robot	-

Tabla 2: Registros E/S utilizados para la comunicación Robot-Autómata

En el lado izquierdo del bloque, se encuentran las variables de tipo "URI" que son escritas por el robot y leídas por el PLC. También se encuentran algunas condiciones que se deben dar para lanzar información a los registros de salida.



Al lado derecho, se encuentran las variables de tipo “URO” que son escritas por el PLC y leídas por el robot, también podemos encontrar las respuestas que da el PLC al recibir las entradas del controlador.

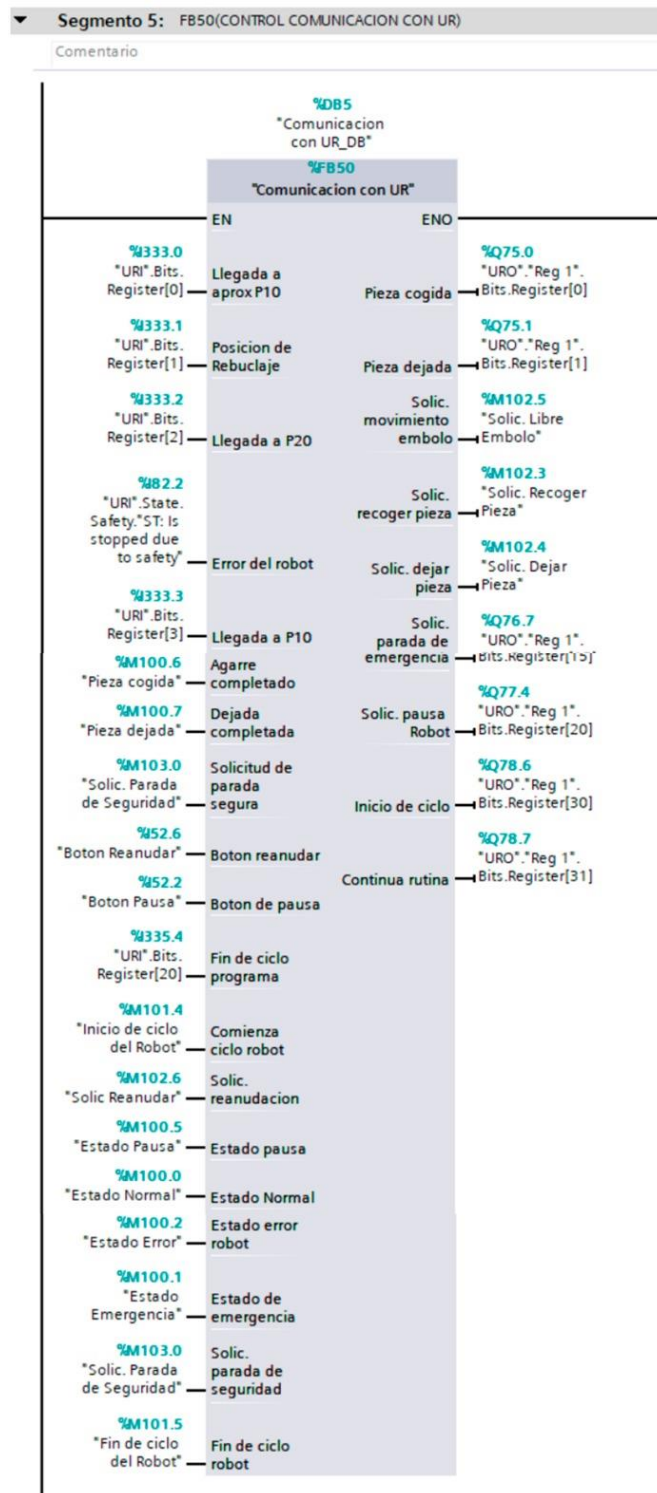


Figura 57: Segmento 5 de la OB1



- **SEGMENTO 6: FB100 (CONTROL DE RUTINA PRINCIPAL). (Figura 58)**

En este segmento se inicializa y se realiza la asignación de variables de la FB100 correspondiente al control de la rutina principal de la estación.

En cuanto a entradas, encontramos las variables definidas en la FB3 sobre el estado del ciclo de trabajo y variables de estado de la estación.

En lado de las salidas, encontramos la sentencia para inciar el ciclo del robot y el movimiento de las cintas.

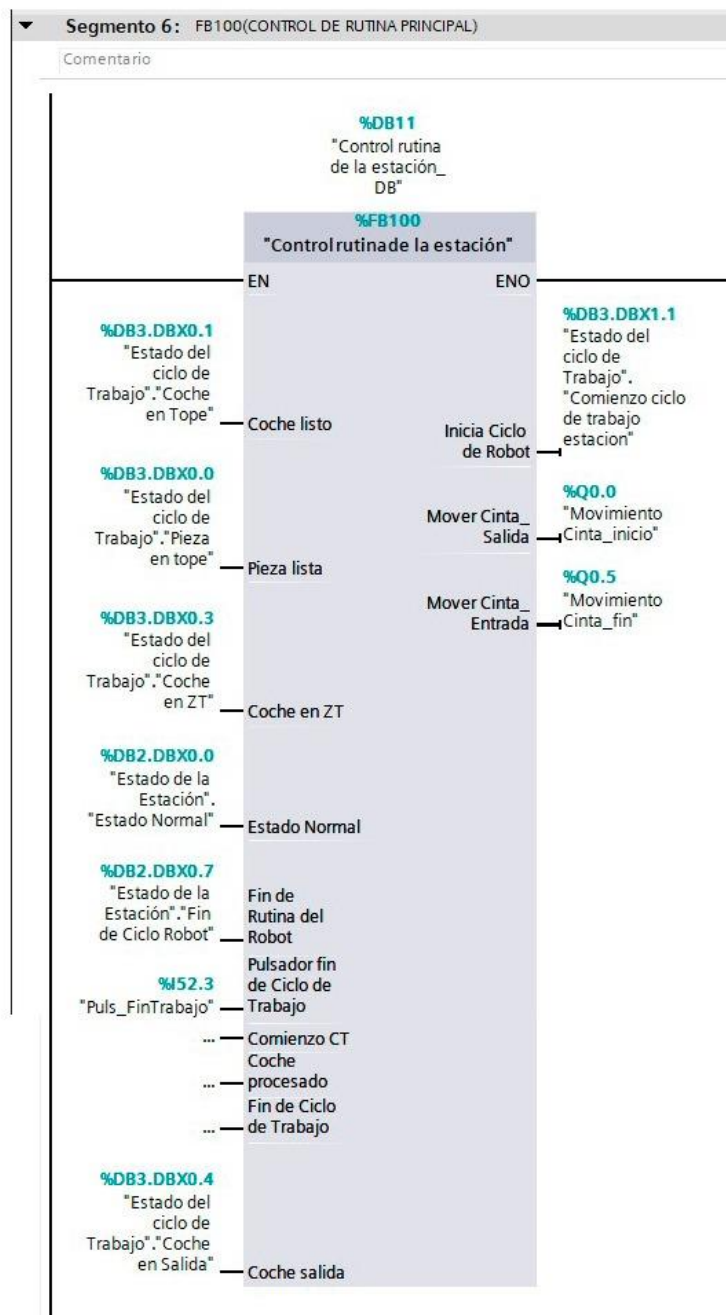


Figura 58:Segmento 6 de la OB1



- **SEGMENTO 7: FB150 (CONTROL DE ELEMENTOS DE SEGURIDAD). (Figura 59)**

Este segmento contiene la inicialización y asignación de variables de la FB150 correspondiente al control de los elementos de seguridad.

En el bloque se encuentran como entradas los diferentes elementos de seguridad y como entrada-salida los estados de la estación relacionados con el control de seguridad.

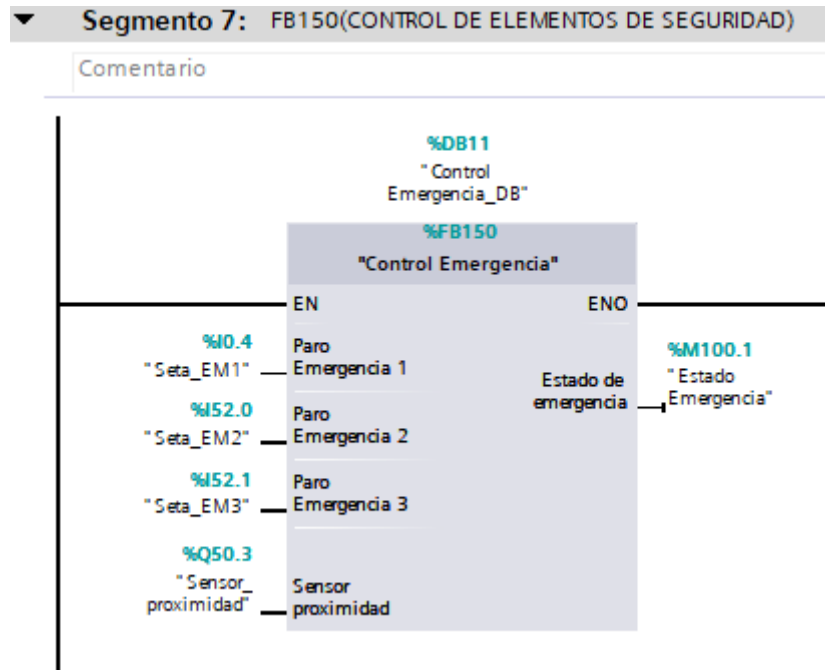


Figura 59: Segmento 7 de la OB1

2.7.3.- Simulación del programa del autómatas

Una vez el programa del PLC ha sido definido, debemos realizar la simulación del programa del mismo.

En la actualidad, no existen apenas simuladores de robótica colaborativa y son más escasos aquellos que cuentan con licencia libre o uso de estudiantes. Debido a esto, en la simulación del PLC debemos obviar el ciclo de trabajo del robot y centrarnos en la administración de los topes, el correcto movimiento de las cintas transportadoras y la correcta respuesta al uso de los botones del panel de control.

Para la simulación del PLC vamos a utilizar la aplicación de Factory IO [25], un software de simulación de PLC dónde podemos construir nuestra estación mediante el uso de objetos predefinidos y de esa manera observar, a tiempo real, el comportamiento de la estación e interactuar con los elementos de esta. La aplicación es de pago, pero cuenta con una prueba gratuita de 30 días.

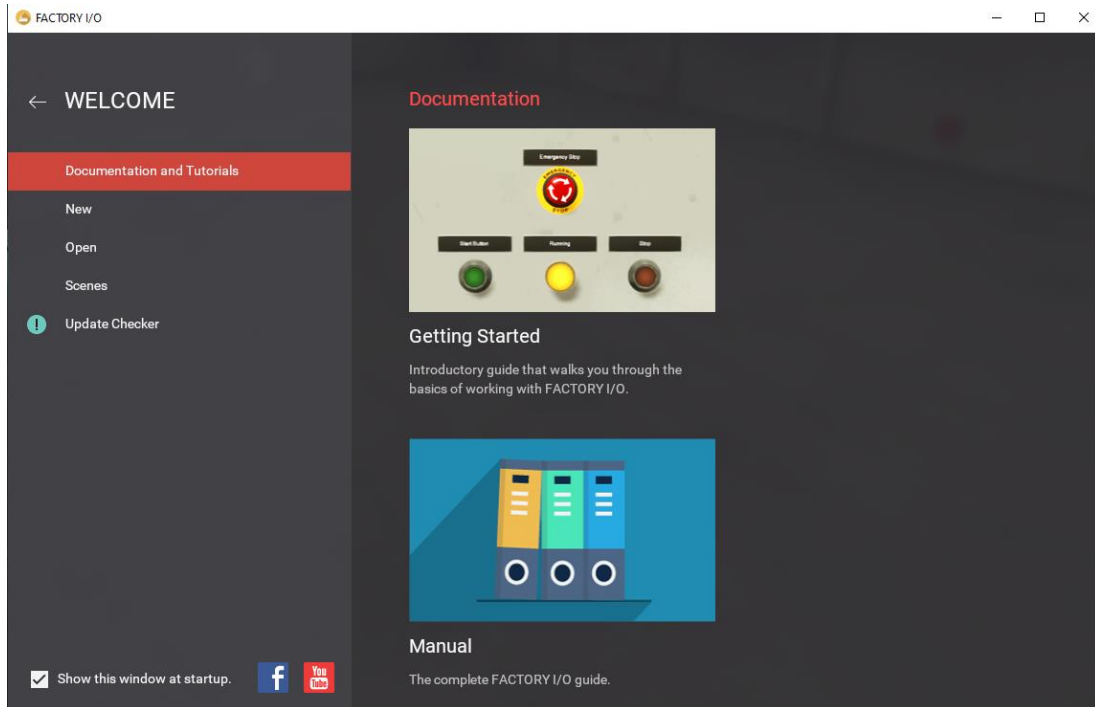


Figura 60: Menú principal de la aplicación

En este apartado explicaremos los elementos básicos de la estación a simular y los procedimientos que debemos seguir para establecer conexión entre nuestro PLC simulado y la aplicación.

En primer lugar, creamos una nueva estación desde la pestaña 'New' (Figura 60). Nos aparecerá el entorno para diseñar nuestra estación, en la figura 61 podemos observar la interfaz del programa.

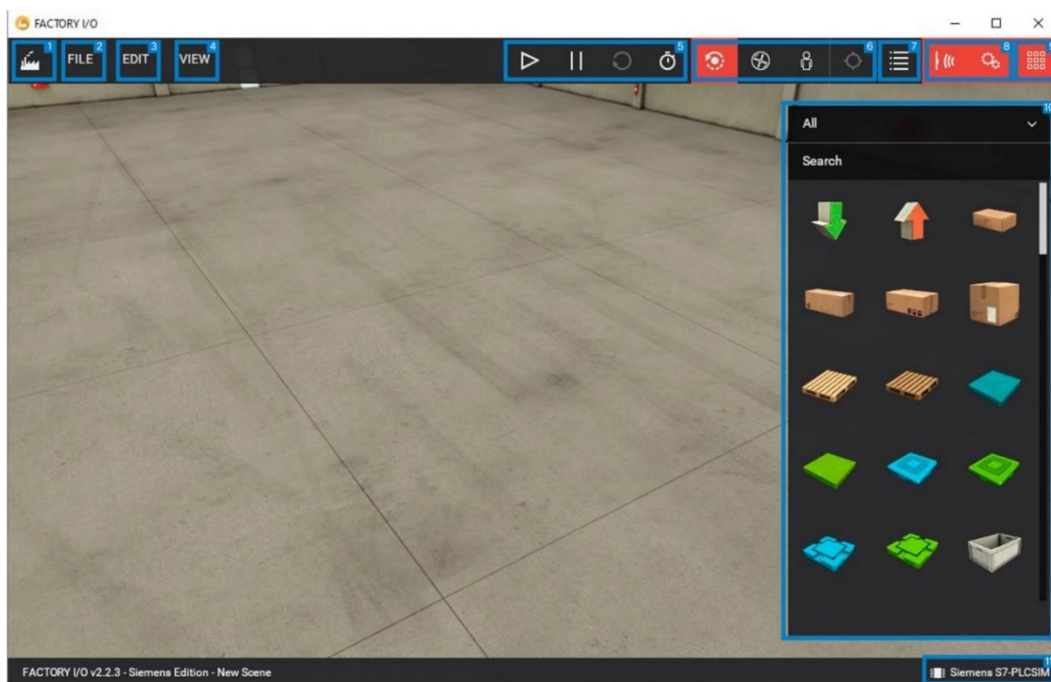


Figura 61: Entorno de programación de TIA Portal



En dicha figura 61 se han remarcado mediante recuadros azules numerados diferentes partes de la interfaz que conviene hacer una breve descripción:

- 1.- Botón para acceder al menú principal
- 2.- Opciones de fichero (abrir, guardar, ...)
- 3.- Opciones de la estación (copiar, pegar, deshacer, ...)
- 4.- Opciones de visualización de elementos de la estación
- 5.- Botones control de flujo de simulación
- 6.- Controles de cámara para la simulación
- 7.- Posiciones de cámaras
- 8.- Visualizar etiquetas de sensores/actuadores
- 9.- Botón para desplegar catálogo de bloques
- 10.- Catálogo de bloques
- 11.- Información del PLC utilizado

En la figura 62 podemos observar una vista general de la estación creada. Eliminando el ciclo y las comunicaciones relacionadas con el cobot y las sentencias del circuito neumático, el programa se queda en una administración del movimiento tanto de los coches, como de las cubiertas (coches sustituidos por palés con caja y las cubiertas por prismas debido a la falta de objetos prediseñados del programa) por las diferentes zonas de trabajo y una administración de los diferentes botones que encontramos en el armario eléctrico y la mesa de control del usuario (Anexo VI: Panel de control del usuario).



Figura 62: Vista general de la estación a simular

Observando la Figura 62 Podemos ver que la estación cuenta con dos zonas principales. La manutención de entrada de las cubiertas plásticas (con el bloque de la flecha verde que es un generador de objetos) que cuenta con un sensor óptico NC (Normalmente cerrado) al final de la



cinta que sirve como tope para detectar si hay una cubierta preparada para iniciar el ciclo de trabajo.

En segundo lugar, tenemos la línea general de la estación, es la manutención (cinta transportadora) que se encarga del movimiento de los coches a lo largo de toda la estación. Cuenta con 4 sensores ópticos NC (Tope, Pretope, Zona de Trabajo y Salida) que dividen toda la línea en 4 sectores básicos (Entrada, Zona de espera, Zona de trabajo y Salida) cada uno de ellos administrado por su correspondiente motor. Al inicio de la cinta encontramos un generador de piezas, y a la salida un destructor de piezas (bloque con la flecha roja).

Con los elementos que nos ofrece este programa no se puede realizar una representación fidedigna de la estación que queremos diseñar y los únicos sensores que encontramos son sensores ópticos de reflexión (Que no son los que se instalarán en la estación final). No obstante, con estos medios podemos realizar una simulación totalmente válida para comprobar el correcto funcionamiento del programa del autómat. En esta simulación no se busca un control de los tiempos de ciclo, se busca el correcto funcionamiento de los componentes

“Una simulación debe ser tan sencilla como sea posible y tan compleja como sea necesario”

Hemos introducido un panel de control, que incluye una serie de botones que son necesarios para llevar a cabo la simulación. En la figura 63 podemos observar el armario de control con mayor detalle.



Figura 63: Panel de control



Partiendo desde la zona superior, sobre el armario encontramos una columna de señalización con 3 colores (Rojo: Emergencia, Amarillo: Saturación y Verde: Estado Normal) y una luz extra de color rojo que corresponde al estado de error del robot. Con estos elementos podemos ver el estado en el que se encuentra la estación en todo momento.

En la parte superior izquierda encontramos una seta de emergencia para simular un error del robot y un botón verde que se encarga de definir el final de trabajo (este botón se situará en el panel de control del trabajador (anexo VI) que, por razones de comodidad, se ha situado junto al resto en el armario de control). La segunda seta que encontramos sirve para solicitar una parada de emergencia.

Por último, en la parte inferior encontramos el botón de rearme (Amarillo) y el botón de marcha (Verde). Estos botones los utilizamos para iniciar el trabajo de la estación y para rearmar la misma en caso de que haya un error del robot o se pulse una seta de emergencia.

Comunicación con el PLC

Con la estación ya definida, podemos empezar a realizar la simulación. En este apartado se explicará paso por paso la manera de establecer conexión entre Factory IO y un PLC simulado del TIA Portal. Este programa, permite también conexión física con un autómata real, no obstante, en este proyecto no nos es de interés.

En primer lugar, debemos descargar la plantilla de comunicación entre el PLC y la aplicación correspondiente. En el menú principal, encontramos en el apartado de tutoriales, diferentes marcas de autómatas. En nuestro caso, seleccionamos Siemens. (Figura 64)

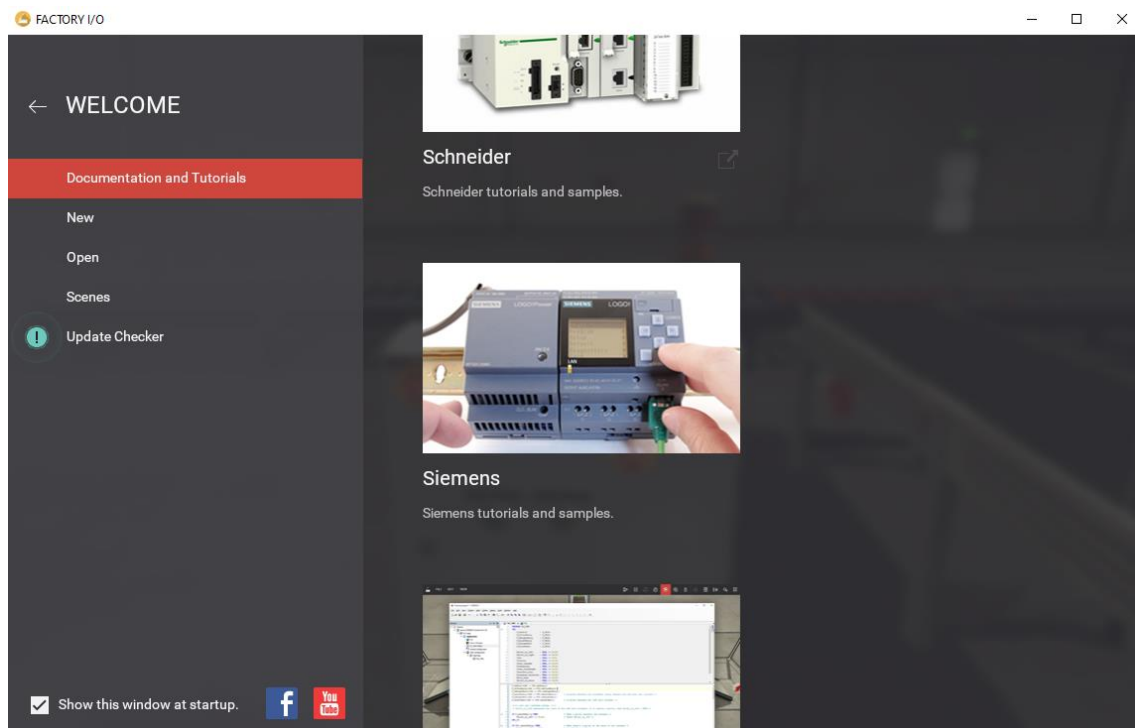


Figura 64: Menu principal de TIA Portal. Selección Siemens.



Al pinchar sobre él, se abrirá el navegador web que nos llevará a la página de Factory IO dónde encontraremos las diferentes opciones que podemos utilizar. De nuevo, entre todas las opciones, nosotros seleccionaremos S7-PLCSIM porque usaremos un autómatas simulado y como estamos utilizando la versión 15 del TIA Portal, seleccionamos la opción “V13/V14/V15 Setting up in Factory I/O”. (Figura 65)

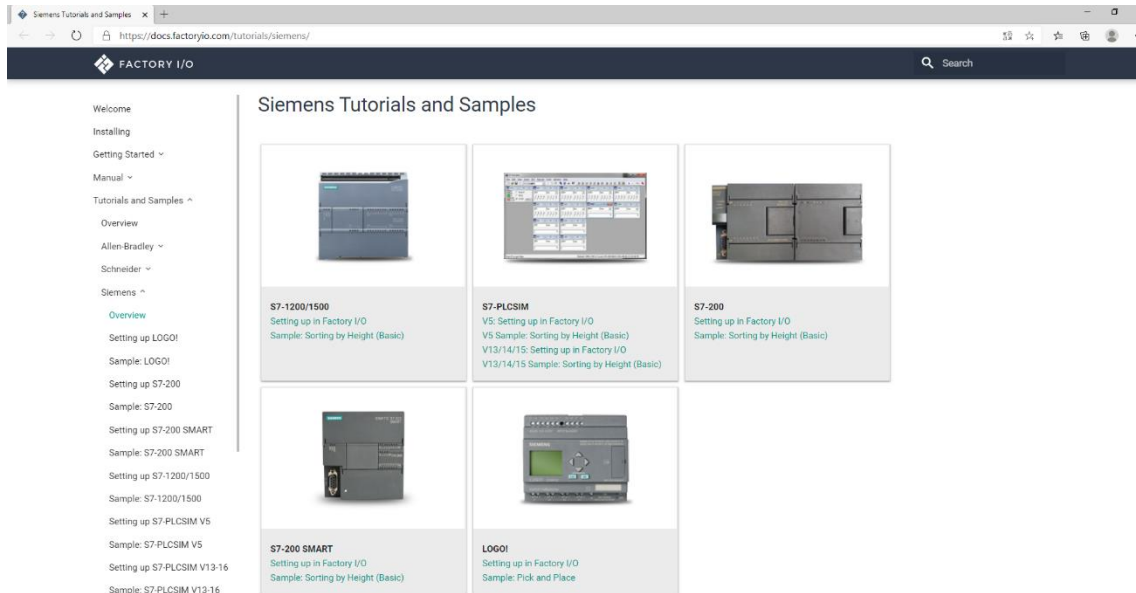


Figura 65: Selección de autómatas en la página web de Factory IO

Nos llevará a una nueva página [26] con un tutorial sobre cómo establecer la conexión y nos aparecerán las diferentes opciones de plantilla que podemos utilizar según nuestra versión de TIA Portal y nuestro modelo de autómatas. Seleccionamos la plantilla para un S7-1500 en TIA Portal V15. (Figura 66)

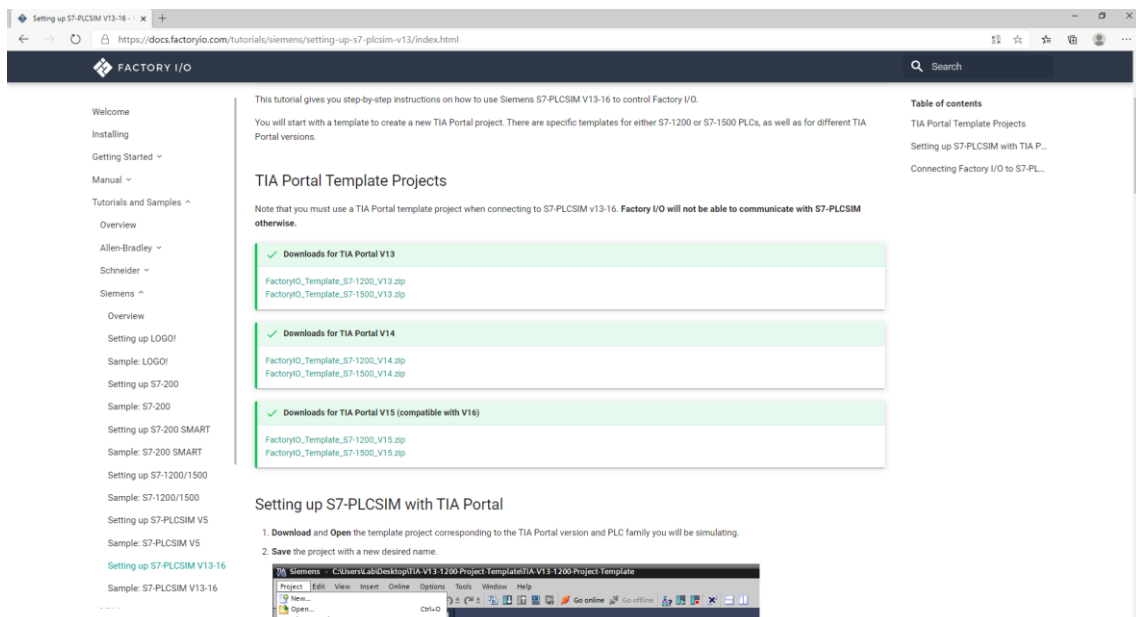


Figura 66: Página de selección de plantilla [26]



Con el archivo descargado y descomprimido, abrimos el proyecto en el TIA Portal y encontramos en el apartado de 'bloque de programa' la FC (Función) correspondiente a la comunicación con Factory IO, en lenguaje SCL [27] (Figura 67). Como ya tenemos el programa diseñado en otro proyecto, copiaremos esta FC y lo llevamos a nuestro proyecto.

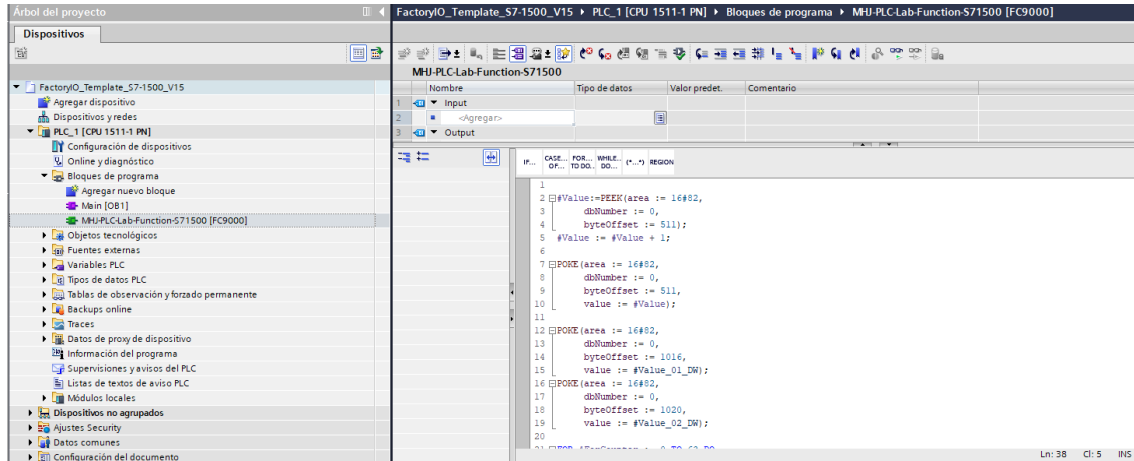


Figura 67: Función de comunicación con el Factory IO

Antes de seguir con la explicación, hay que resaltar que el programa del PLC utilizado no es el mismo al programa final. Se han debido realizar ligeras modificaciones para adaptar el programa al entorno de simulación, no obstante, la estructura básica de funcionamiento permanece intacta.

En nuestro proyecto, ya con el bloque de función importado e inicializado en el Bloque de Organización (OB1). Iniciamos los procedimientos para establecer la conexión. Para ello, seleccionamos la opción de iniciar simulación y nos conectamos con el PLC. (Figura 67)

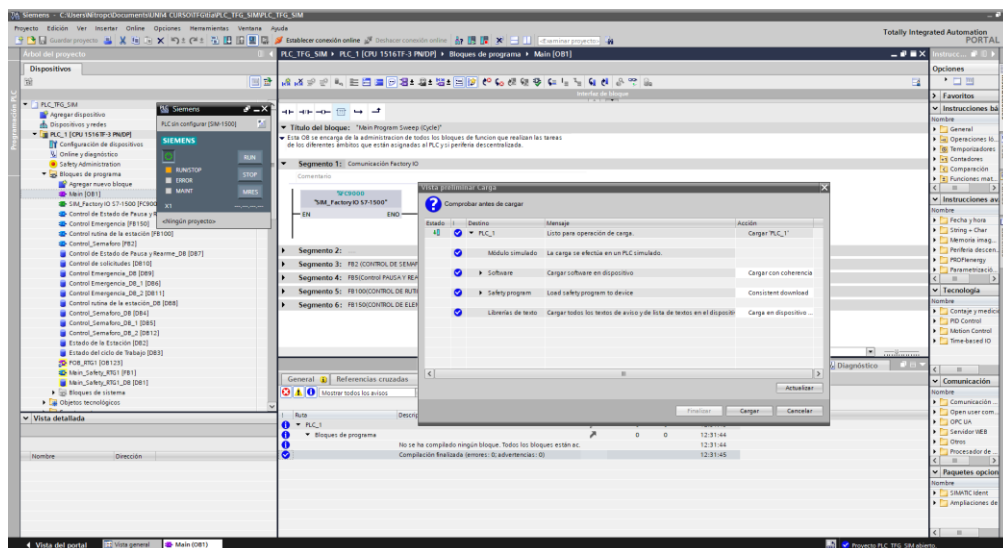


Figura 68: Menú de inicio de simulación

Una vez hemos completado los procedimientos e iniciado la simulación, es muy importante inicial el PLC pinchando en el botón "RUN". De lo contrario, no podremos establecer la conexión.



Después de iniciar la simulación, pinchamos en “establecer conexión online”, de esta manera, el PLCSIM está totalmente activado y listo para establecer la conexión. Ahora solo queda realizar los procedimientos en el Factory IO.

Por su parte, desde Factory IO entramos en el menú de drivers (pulsando F4) y nos aparecerá el menú para establecer la conexión (Driver). En la figura 69 podemos observar la interfaz de este menú. En la parte superior, debemos seleccionar el PLC que vamos a utilizar (en nuestro caso Siemens S7-PLCSIM).

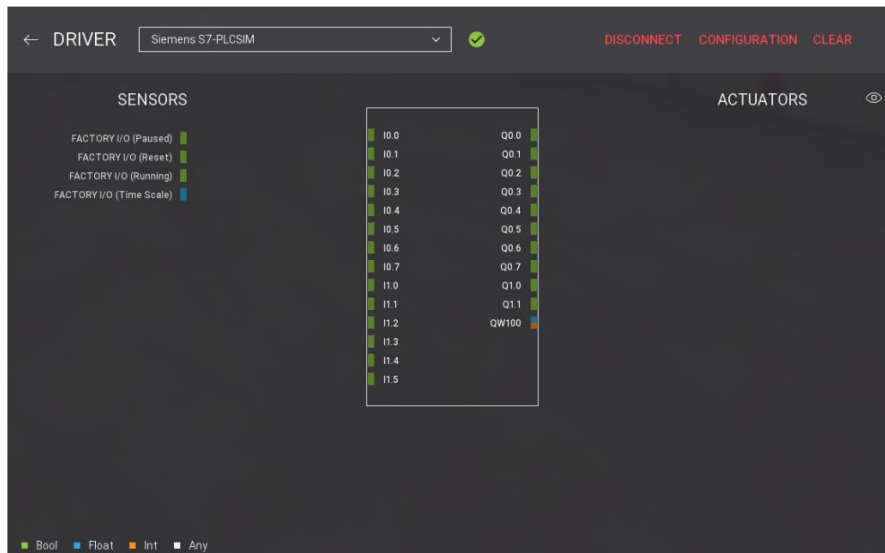


Figura 69: Menú Driver de Factory IO

Para configurar el autómatas que vamos a utilizar, debemos entrar en la ventana “configuration” que encontramos en la parte superior derecha de la pantalla. (Figura 70)

Dentro de esa ventana, seleccionamos la opción de “Siemens S7-PLCSIM” de la lista de opciones de la izquierda y el modelo seleccionamos el correspondiente, en nuestro caso, el S7-1500.

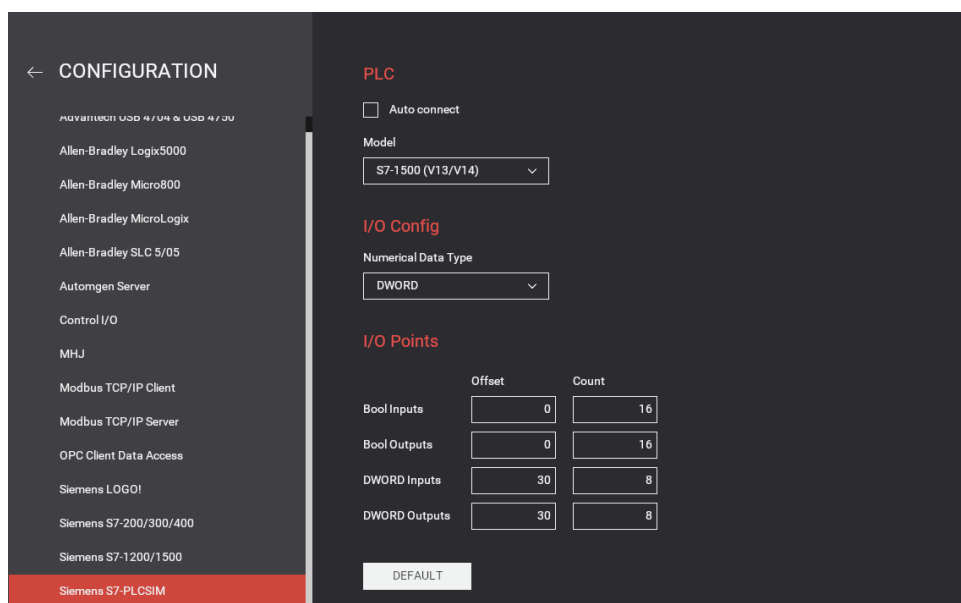


Figura 70: Menú Configuration de Factory IO



Realizado este procedimiento, ya está todo configurado para establecer la conexión, por lo que solo queda volver al menú anterior y seleccionar la opción de conectar (“Connect”). Si la conexión se ha realizado correctamente, en la parte superior aparecerá un icono verde confirmando la conexión, de lo contrario, aparecerá un icono rojo.

Por último, antes de iniciar la simulación, debemos asignar a cada uno de los sensores y actuadores su puerto correspondiente en el autómatas. El resultado, lo podemos observar en la figura 71.

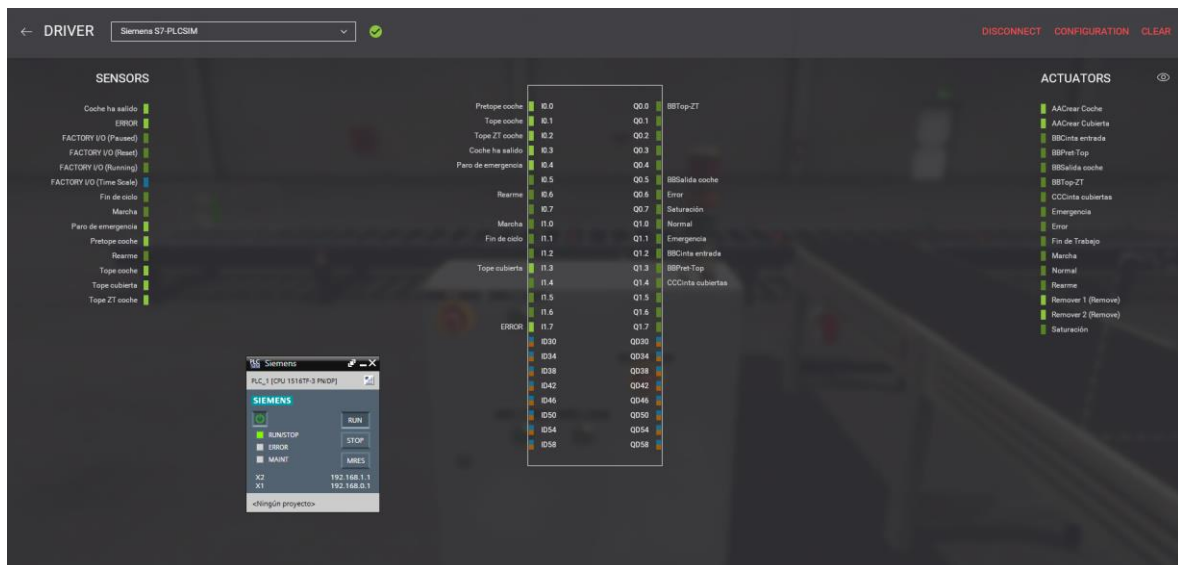


Figura 71: Menú driver con las variables asignadas y el simulador corriendo + ventana de PLCSIM

Con el PLC ya conectado, está todo configurado para iniciar la simulación. Volviendo al entorno de simulación pulsamos el botón de inicio de la simulación y esta comenzará. Para iniciar nuestra estación en el estado inicial es necesario rearmar la estación, por lo que debemos pulsar el botón Rearme y después el botón Marcha.



Figura 72: Simulación de la estación en funcionamiento



Tras realizar unos ligeros ajustes, conseguimos que la estación funcione correctamente (Figura 72), tal y como se había planeado que funcionara. Gracias a esta simulación hemos podido comprobar que el programa del PLC referente al movimiento de las cintas, control del estado de la estación en función de los topes y respuesta de la estación ante las señales de los botones del armario de control es satisfactoria y podemos llevar este programa a la siguiente etapa.

Con las simulaciones realizadas, hemos confirmado el correcto funcionamiento del circuito neumático y del comportamiento de la estación. Los únicos elementos que no se han podido simular son el sensor de presencia y el ciclo de trabajo del robot colaborativo. Estos elementos sin simular se pueden comprobar de manera física en un entorno de pruebas.

2.8.- Programación y simulación de la rutina del robot

Introducción

El programa de la rutina del robot se puede realizar de dos maneras similares. La primera es a través de la TeachPendant. PolyScope Robot User Interface (generalmente llamado solamente PolyScope) es el sistema operativo de Universal Robots que permite trabajar con el controlador. Su interfaz es intuitiva y sencilla cuyo entorno tiene una curva de aprendizaje muy suave. En cuanto a la programación, se basa en el uso de bloques predeterminados (MoveJ, MoveL, if...else, ScriptCode, etc.) con el que podemos configurar nuestro programa.

La segunda posibilidad, y la utilizada en este caso, es a través de una máquina virtual (VM) que contiene el Sistema Operativo del controlador (PolyScope) dentro de una distribución de Ubuntu, este es el mismo que vemos en la TeachPendant. A través de esta VM podremos realizar las mismas tareas que con la pantalla táctil de la TeachPendant, incluido la programación del robot. Una vez terminado el programa, debemos guardar el programa en una unidad USB para posteriormente cargarlo en el controlador introduciendo el USB en el puerto correspondiente que hay en la parte trasera de la TeachPendant (o dentro del armario de controlador para versiones antiguas) y cargar la salvaguarda del programa.

El Manual de Usuario del UR10e incluye un capítulo sobre la programación y el entorno de PolyScope [15].

Entorno de programación PolyScope: Configuración para conexión PROFINET

En este apartado se explicará los pasos a seguir para habilitar la conexión PROFINET del controlador, a crear un nuevo programa y por último se explicarán los pasos para inicializar el robot. Antes de empezar, se debe comentar que PolyScope cuenta con una traducción al español, pero no es del todo fiable, por lo que se utilizará con el idioma original, el inglés.



- **MENU PRINCIPAL**

Tanto al encender el controlador del UR, como al iniciar el programa en la máquina virtual, tras una pantalla de carga, nos aparecerá el menú que aparece en la figura 73 (el menú puede haber cambiado estéticamente en versiones posteriores).

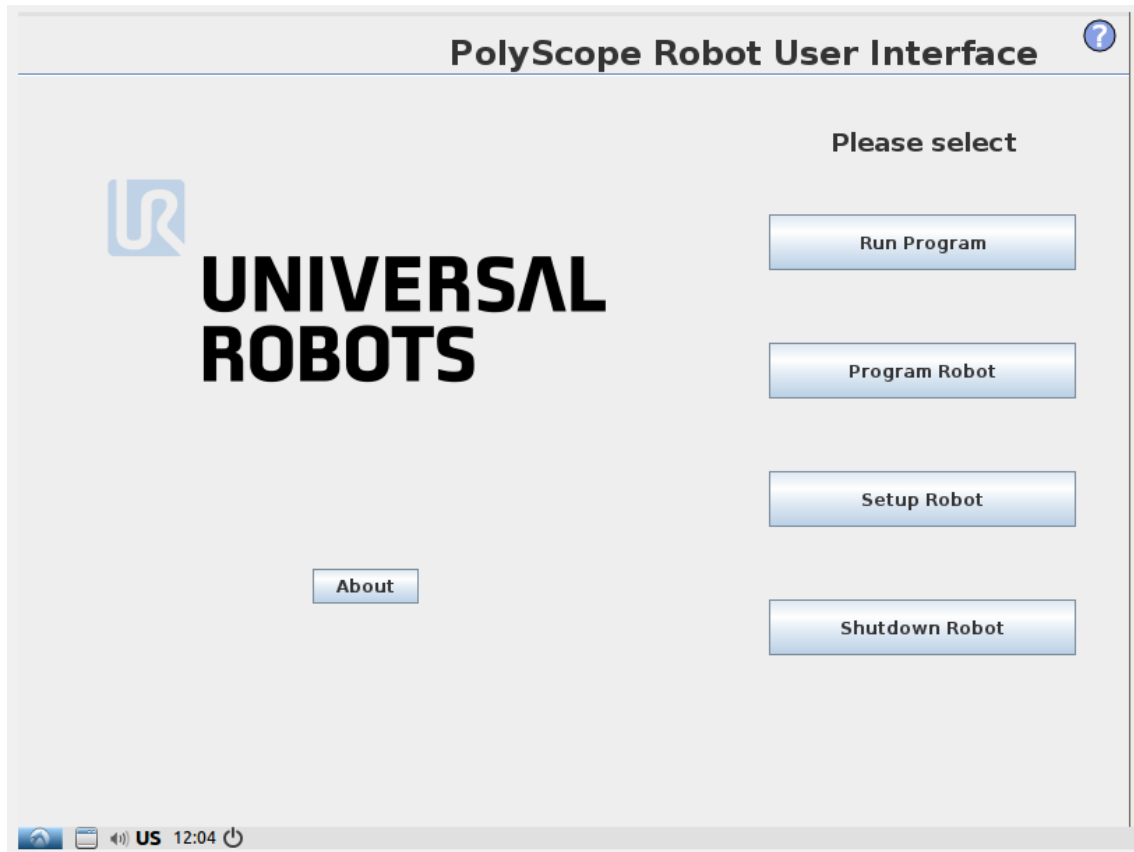


Figura 73: Menú principal del PolyScope

En este menú principal, a parte de la pestaña “About”, que contiene información correspondiente a la versión y configuración del PolyScope; encontramos 4 principales ventanas:

- **Run Program:** Nos muestra información correspondiente con el programa que está actualmente en ejecución; como el tiempo que lleva el programa en ejecución, el estado, el valor de las entradas y salidas digitales/analógicas, información de temperatura y valores de tensión y corriente de alimentación.
- **Program Robot:** Con esta pestaña entramos al editor del programa. Al seleccionar esta ventana nos aparece un nuevo menú donde podemos elegir entre cargar un programa guardado en memoria, utilizar una plantilla para una rutina Pick & Place y abrir un programa vacío.
- **Setup Robot:** Esta tercera pestaña corresponde a la configuración del PolyScope. A partir de ella podemos inicializar el robot, cambiar el idioma del programa, calibrar la pantalla, entre otros.



- **Shutdown Robot:** Esta es la última pestaña de este menú principal. En el caso de la máquina virtual sirve para cerrar el programa, mientras que en la TeachPendant sirve para apagar el controlador del cobot.

- **HABILITAR PROFINET EN EL UR**

En este apartado vamos a describir los pasos necesarios para habilitar la conexión PROFINET del controlador del UR.

Para habilitar la conexión PROFINET del UR, debemos seleccionar la ventana “Program Robot” en el menú principal, y en la parte superior seleccionar la pestaña “Installation”, una vez dentro de esta pestaña, debemos seleccionar la opción de PROFINET, dónde nos aparecerá un menú como el siguiente (Figura 74):



Figura 74: Manú habilitar PROFINET

Para habilitar la conexión PROFINET debemos pulsar en el botón “Eneable”. Una vez habilitada, el LED que aparece a la izquierda de la palabra ‘PROFINET’ se encenderá en amarillo si el PROFINET se está ejecutando en el robot, pero no hay ningún controlador PLC / IO conectado al robot. Saldrá el LED de color rojo si hay algún error que no permita habilitar la conexión PROFINET en el robot. Y aparecerá de color verde si hay una conexión entre el robot y el PLC/ IO. En la figura 75 (siguiente página) observamos una correcta conexión PROFINET entre robot y PLC:



Figura 75: PROFINET habilitado y activo

Una vez establecida la conexión es muy importante guardar la configuración de la estación pulsando sobre la última opción de la columna. Una vez guardada esta nueva configuración, el código de números, letras y colores que aparecen en la parte superior derecha de la pantalla cambiará de manera aleatoria. Este código se trata de un 'Safety Checksum' utilizado para llevar el control ante posibles cambios no autorizados en la configuración de la estación.

En el Anexo II podemos ver el funcionamiento de la conexión PROFINET del robot.



- **INICIALIZAR ROBOT**

En este apartado se explicarán los pasos a seguir para inicializar el robot. Inicializar el robot significa habilitar la alimentación a los diferentes componentes del brazo robótico y liberar frenos de seguridad para que el robot se pueda mover al iniciar la rutina.

Para acceder al menú de inicialización partiendo del menú principal debemos pulsar en la ventana de "Setup Robot". Una vez dentro, seleccionamos la primera pestaña llamada "Initialize Robot". El menú de inicialización del robot que nos aparece es el de la figura 76:

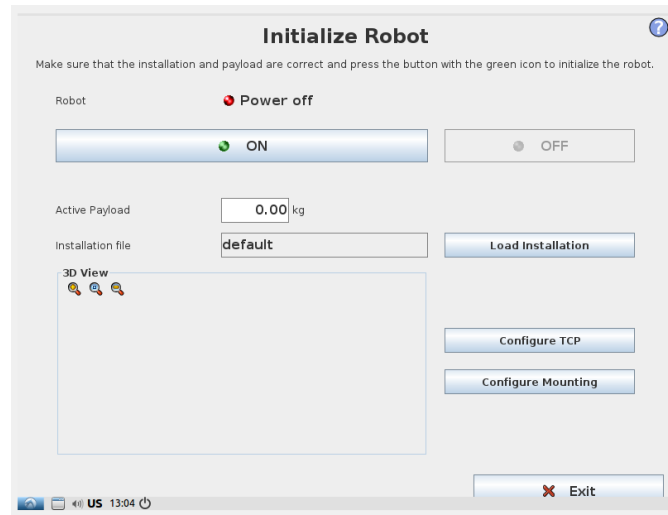


Figura 76: Menú de inicialización del robot

Como podemos ver, al iniciar el programa, el robot nos aparece con la alimentación apagada (Power off). Si pulsamos sobre el botón de encendido (ON) habilitaremos la entrada de alimentación al robot. En este punto, en lugar de 'power off' nos mostrará el estado del robot.

Como nosotros partimos de un estado inicial, al pulsar por primera vez el botón ON el LED aparecerá de color amarillo y nos aparecerá la palabra 'idle' (ocioso) (Figura 77).

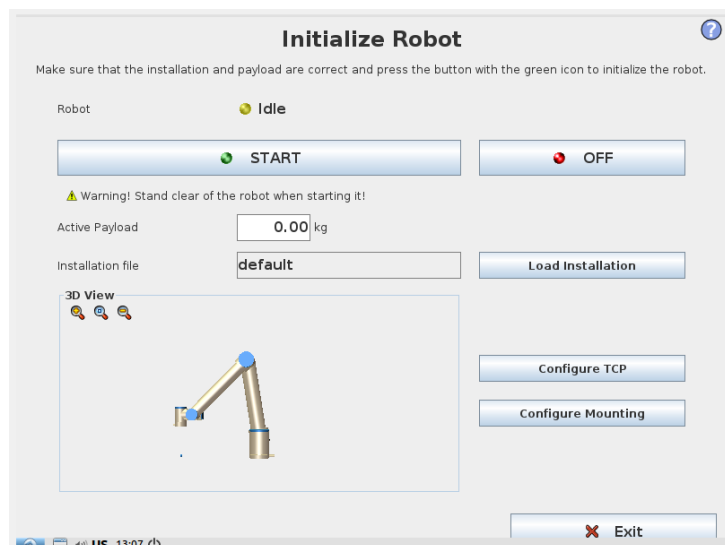


Figura 77: Robot en estado ocioso



Como podemos ver en la imagen, ahora en lugar del botón ON nos aparece la ventana 'START' y a la derecha nos aparece el botón para deshabilitar la alimentación del robot. Una vez pulsado el botón "START" el robot pasará de 'Idle' a 'Normal' y el LED será de color verde.

En este menú también podemos cargar una configuración de instalación específica pulsando sobre el botón "Load Installation".

- **CREAR UN PROGRAMA NUEVO**

En este apartado se expondrán los pasos a seguir para crear un programa nuevo y alguno de los elementos principales a tener en cuenta a la hora de programar.

Partiendo del menú principal, seleccionamos el botón "Program Robot". Una vez dentro nos aparecerá el siguiente menú (Figura 78), dónde seleccionaremos la opción "Empty Program" dentro del bloque "Use template".

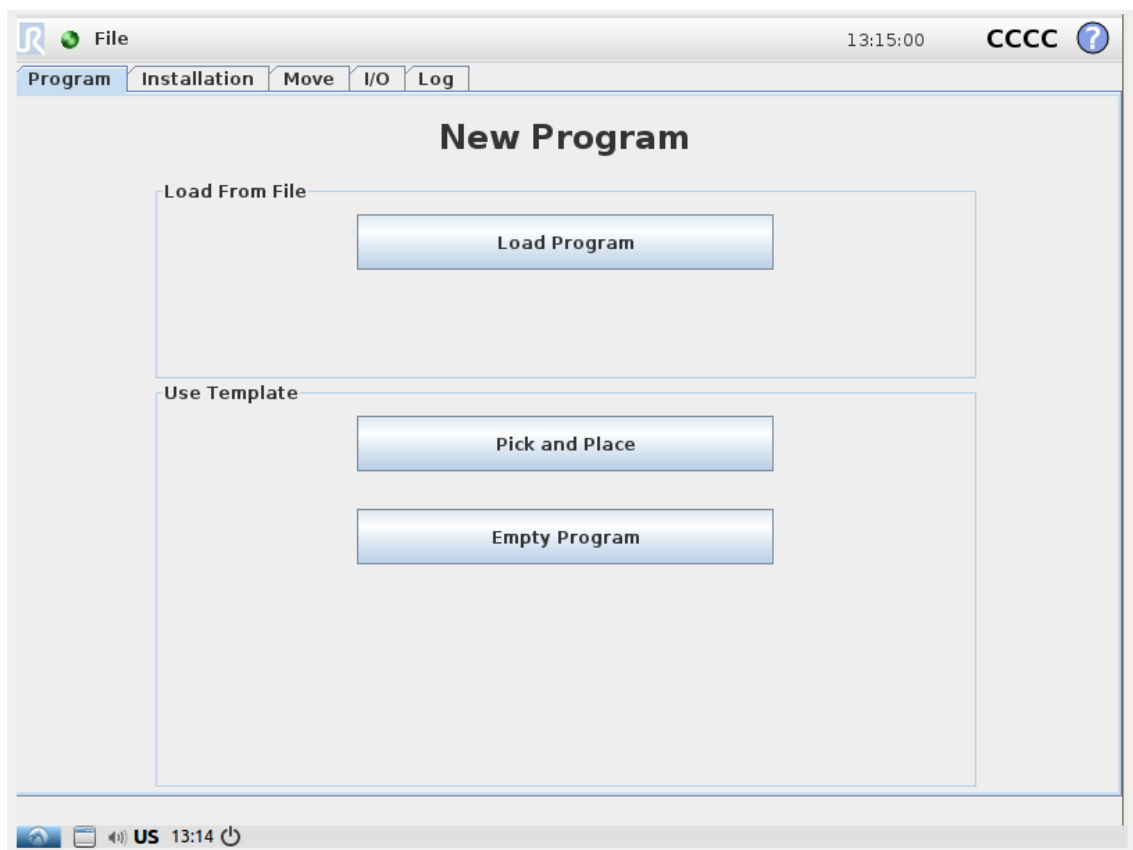


Figura 78: Menú "New Program"

Una vez seleccionada esa opción, nos aparecerá el entorno de programación (Figura 79). Hablando brevemente de él, podemos ver que en la parte izquierda aparece el código del programa definido a través de sentencias predefinidas.

En la parte inferior encontramos el botón de inicio y parada, el botón para saltar a la siguiente sentencia y el botón de retroceder a la primera sentencia del programa. Al lado derecho de estos botones encontramos una barra para seleccionar los valores de velocidad nominal de robot. Esta



velocidad, es una proporción de velocidad nominal definida en los parámetros de seguridad del robot, dentro del propio programa se puede establecer una velocidad propia para una trayectoria específica.

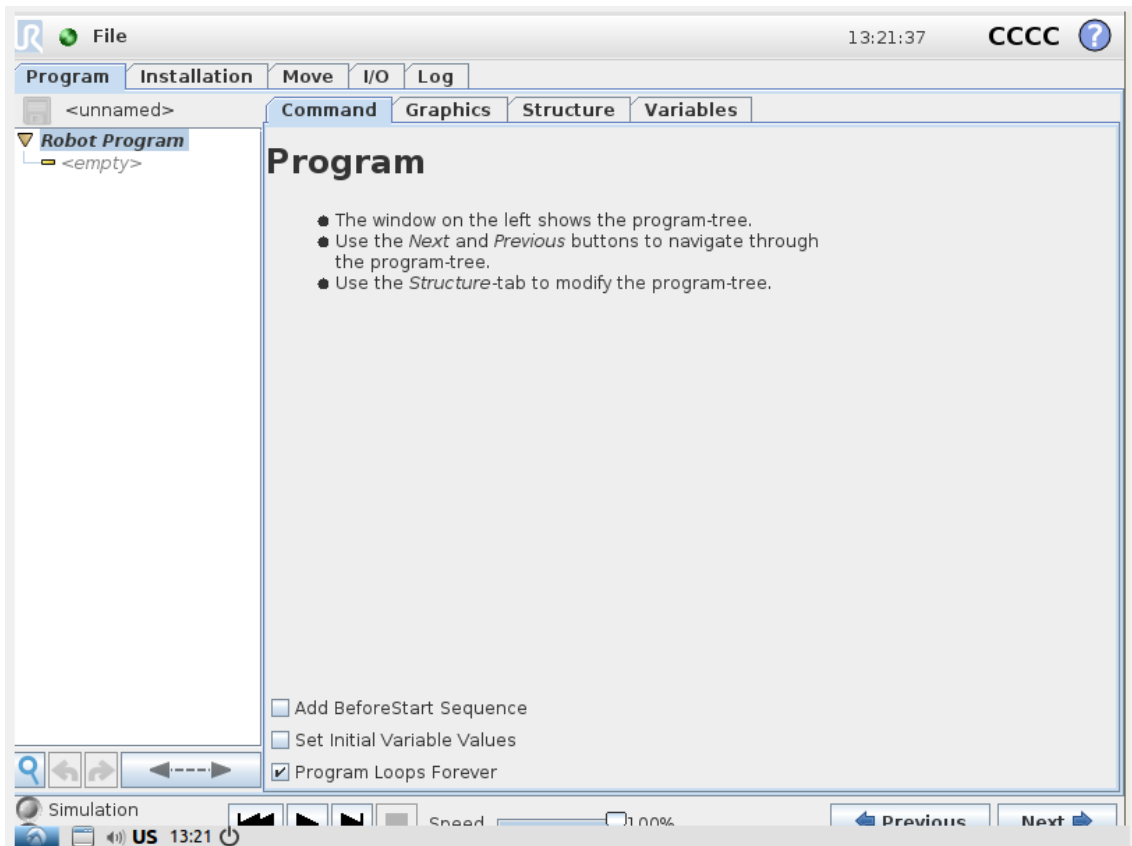


Figura 79: Entorno de programación de PolyScope

En la parte derecha de la figura 79, podemos ver, organizado en pestañas, la zona de programación. Desde el apartado de “Structure” podemos seleccionar los bloques predefinidos para la programación del cobot, algunos de ellos son el movimiento libre (MoveJ), movimiento lineal (MoveL), flujo condicional (if... else), esperar a cierto evento o un cierto tiempo (Wait) y una gran variedad de bloques de programación que podemos encontrar explicados en el Manual de Usuario [15]

En cuanto a las pestañas de “Graphics” y “Variables”, podemos ver la una representación gráfica del brazo robótico y el estado de las variables respectivamente a lo largo de la ejecución del programa.

Por último, en la ventana “Command” podemos configurar los bloques predefinidos de los que hemos hablado anteriormente.



- **PROGRAMA PRINCIPAL**

Por último, vamos a hablar del programa encargado de realizar la rutina del robot, incluyendo la comunicación con el PLC mediante los registros de entrada y salida del robot.

La ejecución de estos programas se realiza de forma cíclica, es decir, una vez termina todas las sentencias del programa, vuelve al principio del programa, esto se puede restringir utilizando el comando "Halt"; no obstante, en este proyecto no se va a utilizar.

El programa consiste en una rutina de Pick & Place dónde nos comunicamos con el PLC para enviarle información del estado de trabajo del robot y recibiremos información del PLC correspondiente al estado de la estación.

Descripción del programa

En este apartado dividiremos el programa principal en bloques para poder hacer una descripción del programa más exhaustiva. Para poder ver el código completo y conocer la configuración de los puntos, debemos ir al Anexo V: "Programa de Polyscope". Para observar las variables asignadas a los registros E/S del robot consultar la tabla 2.

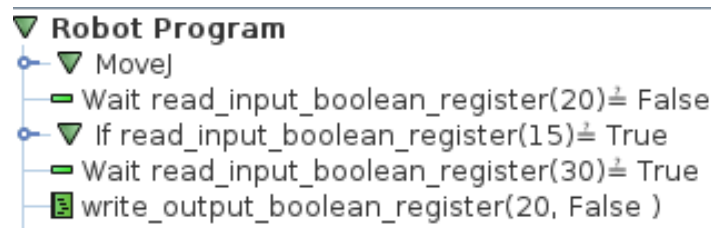


Figura 80: Segmento 1 del programa

(Figura 80): Al comienzo del programa, debemos asegurarnos de que el robot se encuentra en la posición de rebucleaje, una posición segura de reposo desde la cual el robot puede acceder a los diferentes puntos de su trayectoria. Una vez en dicha posición, mediante el registro de salida 1, notificamos que el robot se encuentra en rebucleaje. Ahora, mediante una función de demora (Wait), comprobamos, mediante la lectura del registro de entrada 20, que no haya una parada de seguridad. Esto está enfocado principalmente para futuros ciclo de iteración.

Una vez se han realizado las comprobaciones, se reinicia el registro de salida 20 que corresponde con la señal de fin de ciclo de robot, de la cual hablaremos más adelante.

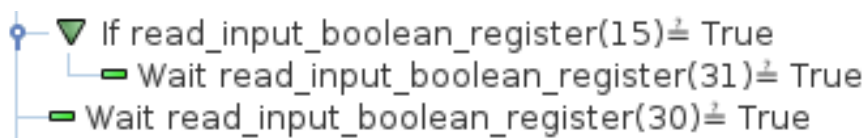


Figura 81: Segmento 2 del programa

(Figura 81) Una vez hemos comprobado que no hay parada de seguridad, comprobamos si se ha solicitado una parada de emergencia (registro de entrada 15) y en caso de que haya una emergencia, se espera a que desde el armario de control se haya solucionado dicha emergencia



y se haya solicitado la reanudación de la rutina (registro de entrada 31). Esta sentencia condicional junto con la demora para reanudar la rutina va a aparecer varias veces a lo largo del programa, pero solo hablaremos en detalle de dicha sentencia en este bloque.

No habiendo ninguna emergencia, el robot se queda a la espera en su posición de rebuclaje hasta que desde el PLC se le solicita, mediante el registro de entrada 30, que inicie la rutina de Pick & Place.

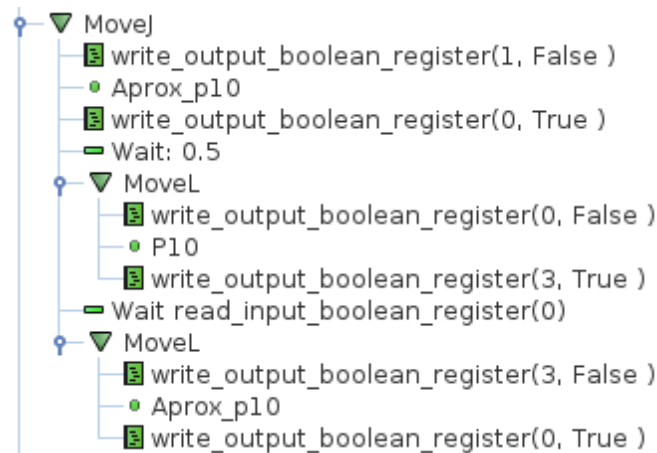


Figura 82: Segmento 3 del programa

(Figura 82): Una vez se ha solicitado el inicio de la rutina Pick & Place comienza el movimiento del robot a la posición p10, en este punto se encuentra la cubierta a recoger. La trayectoria del robot hasta el punto p10 pasa por un punto de aproximación con el fin de atacar a la cubierta con una posición y orientación de la pinza determinada.

En el programa podemos ver que al iniciar el movimiento de robot hasta el punto de aproximación “Aprox_10” mediante una sentencia MoveJ (movimiento libre de los ejes) se elimina del registro 1, el valor que indicaba que el robot se encontraba en rebuclaje, y al llegar al punto de aproximación, se indica en el registro 0 que se ha llegado al punto “Aprox_p10”. Con esa señal, el PLC comienza el proceso para liberar el émbolo.

Dejamos que el robot repose 0.5s para asegurar, por una parte, que el sistema neumático responda a la señal (por lo general el sistema neumático actúa en un tiempo prácticamente instantáneo); y por otro lado para asegurar que el robot se queda en dicha posición para atacar correctamente al punto 10 y eliminando las inercias que se pueden provocar al hacer un cambio de trayectoria perpendicular.

Con el robot en posición y el émbolo libre, empezamos el ataque al punto p10 con el fin de coger la cubierta. A diferencia del resto de movimientos, este se trata de un movimiento lineal (MoveL, un movimiento más restrictivo que el MoveJ con el fin de lograr que la trayectoria describa una línea recta), al igual que con el anterior movimiento, al iniciar la trayectoria se borra el registro 0 que indicaba que el robot se encontraba en el punto “Aprox_p10” y una vez llega al punto “p10” se indica dicho evento mediante el registro 3.

Con la señal del registro 3, se indica al PLC que el robot está en posición para agarrar la pieza y se inicia el control neumático para dicho proceso. Al terminar el proceso en el PLC, se envía una



señal de verificación (Acknowledgement) a través del registro 0 al robot que se mantenía a la espera de dicha verificación.

Recibida esta verificación el robot vuelve mediante un movimiento lineal a la posición “Aprox_p10” con la cubierta recogida e indicando mediante los registros nombrados anteriormente la posición en la que se encuentra.

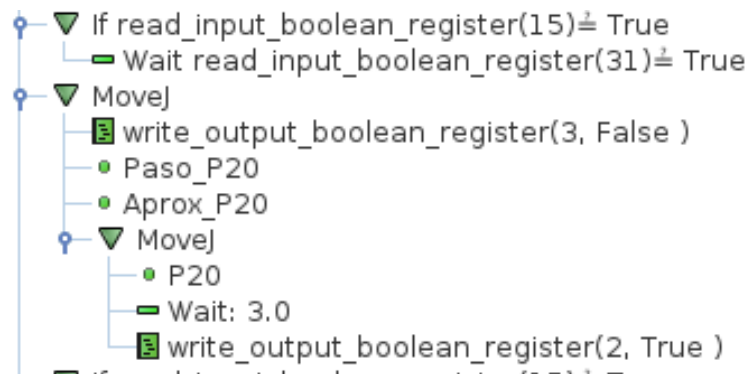


Figura 83: Segmento 4 del programa

(Figura 83): Antes de pasar al segundo movimiento, se comprueba que no haya saltado el sistema de emergencia. En caso de que la estación siga en estado normal, el robot pasará del punto “Aprox_p10” al punto “p20” pasando por los puntos “Paso_p20” y “Aprox_P20” para establecer una trayectoria específica evitando las singularidades que provoquen un fallo y detengan la trayectoria del robot. Tras una espera de 3 segundos, se notifica al PLC que el robot se encuentra en el punto de dejada de la cubierta mediante el registro de salida 2. Una vez el PLC reciba dicha señal, realiza la sentencia de dejada de pieza sobre el sistema neumático.

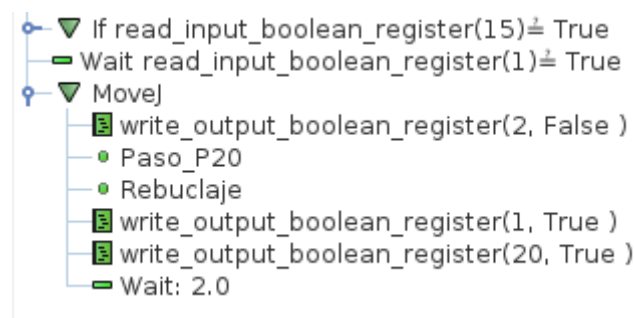


Figura 84: Segmento 5 del programa

(Figura 84): Con la pieza ya dejada comprueba una vez si hay alguna emergencia y en caso de que no haya, volverá al punto de rebuclaje pasando por el punto de paso llamado “Paso_P20”, notificando mediante los registros de salida la posición en la que se encuentra. Al llegar a dicha posición, activa el registro de salida 20, que corresponde con la señal que indica el fin de ciclo del robot. Esta señal nos sirve para detectar el momento en el que el robot ha terminado su ciclo de trabajo para quedar a la espera de que el usuario pulse el botón “Fin de ciclo de Trabajo” dando así por terminado el ciclo de trabajo global de la estación y desalojando el coche ya procesado hacia la siguiente estación.



La última sentencia del programa es una espera de 2 segundos dónde el robot se mantiene en rebuclaje y sirve como transición entre ciclos de trabajo. Esta sentencia no es obligatoria debido a que, en líneas generales, los procedimientos de fin de ciclo tardan más de 2 segundos en completarse; no obstante, en caso de que haya algún problema con el funcionamiento de la estación, supone un elemento de seguridad que puede evitar un solapamiento en los ciclos si se produce una desincronización con los elementos de la estación.

2.9.- Estudio económico

En este apartado se va a realizar el estudio correspondiente a la carga económica que supondría la construcción de un prototipo de la estación. Este prototipo, servirá para confirmar las hipótesis realizadas en la etapa de estudio de viabilidad y en las simulaciones y nos permitirá comprobar la correcta comunicación entre el PLC, el robot y el sistema neumático.

A lo largo de este apartado aparecerán los elementos y procedimientos que suponen un cargo económico para el desarrollo del prototipo, estos estarán agrupados en diferentes tablas, teniendo en la última fila un subtotal. Al final de este apartado, encontraremos una tabla dónde aparecerá el precio final.

Nombre	Referencia	Marca/Proveedor	Unidades	Precio/U	Total
CPU 1516TF-3 PN/DP	6ES7 516-3UN00-0AB0	Siemens	1	4.300,00 €	4.300,00 €
CP1542-1	6GK7 542-1AX00-0XE0	Siemens	1	755,00 €	755,00 €
DI 16x24VDC HF	6ES7 521-1BH00-0AB0	Siemens	1	170,00 €	170,00 €
DQ 16x24VDC/0,5A HF	6ES7 522-1BH01-0AB0	Siemens	1	175,00 €	175,00 €
Tarjeta Memory Card SIMATIC 12Mb	6ES7954-8LC03-0AA0	Siemens	1	120,00 €	120,00 €
				Subtotal:	5.520,00 €

Tabla 3: Precio del sistema principal de seguridad

Nombre	Referencia	Marca/Proveedor	Unidades	Precio/U	Total
ET 200SP (IM 155-6 PN HF)	6ES7 155-6AU00-0CNO	Siemens	1	303,40 €	304,40 €
CM 4xIO-Link	6ES 137-6BD00-0BA0	Siemens	1	239,76 €	240,76 €
DI 8x24VDC ST	6ES7 131-6BF00-0BA0	Siemens	1	113,90 €	114,90 €
DQ 4x24VDC/2A HF	6ES7 132-6BD20-0CA0	Siemens	1	62,16 €	63,16 €
Módulo servidor	6ES7 193-6PA00-0AA0	Siemens	1	30,34 €	31,34 €
				Subtotal:	754,56 €

Tabla 4 Precio del sistema de periferia descentralizada

Nombre	Referencia	Marca/Proveedor	Unidades	Precio/U	Total
UR10 CB3		Universal Robot	1	37.400,00 €	37.400,00 €
Pedestal robot		SOMEC	1	2.139,00 €	2.139,00 €
Pupitre de usuario		SOMEC	1	5.844,77 €	5.844,77 €
				Subtotal:	45.383,77 €

Tabla 5: Precio de Robot +pedestal + pupitre



Universidad de Valladolid

Nombre	Referencia	Marca/Proveedor	Unidades	Precio/U	Total
Simatic Field PG M6		Siemens	1	6.916,00 €	6.916,00 €
Licencia Siemens SIMATIC STEP 7 Professional V16	6ES7822-1AA06-0YA5	Siemens	1	2.467,32 €	2.467,32 €
Subtotal:				9.383,32 €	

Tabla 6: Precios de las herramientas de programación

Nombre	Referencia	Marca/Proveedor	Unidades	Precio/U	Total
Seta de emergencia	XALK178E	Schneider	3	44,65 €	133,95 €
Detector de presencia laser	S30A-4011BA	Sick	1	4.200,00 €	4.200,00 €
Subtotal:				4.333,95 €	

Tabla 7: Precio de elementos de seguridad

Nombre	Referencia	Marca/Proveedor	Unidades	Precio/U	Total
Pinza Adaptativa	DHEF 20-A	Festo	1	527,00 €	527,00 €
Electroválvula 5/2	VUVS-LT20-M52-MD-G18-F7	Festo	1	74,62 €	74,62 €
Válvula de estrangulación y antirretorno	152881	Festo	2	63,81 €	127,62 €
Electroválvula 3/2	VUVS-LT20-M32C-MD-G18-F7	Festo	2	64,88 €	129,76 €
Electroválvula 2/2	VQC4200-51	SMC	1	124,94 €	124,94 €
Racord G1/8	QS-G1/8-8	Festo	16	46,37 €	741,92 €
Estación acondicionamiento de aire	MSB6-1/2:C3:J1:D14-WP	Festo	1	475,34 €	475,34 €
Subtotal:				2.201,20 €	

Tabla 8: Precio de los elementos neumáticos

Nombre	Referencia	Marca/Proveedor	Unidades	Precio/U	Total
Transformador 220VAC/24VDC	6EP1333-2BA20	Siemens	1	89,42 €	89,42 €
Magnetotérmico 220VAC	5SL6116-7	Siemens	1	24,40 €	24,40 €
Siemens Sentron	7KM2112-0BA00-2AA0	Siemens	1	606,48 €	606,48 €
Subtotal:				720,30 €	

Tabla 9: Precio de los elementos eléctricos

Nombre	Horas	Precio/Hora	Total
Instalación de Vías neumáticas (30m) G1/8			270,40 €
Instalación UR y pinza			288,00 €
Instalación de elementos en panoplia			265,00 €
Cableado estación	12	70,00€	1.010,00 €
Programación	120	7,20€	864,00 €
Subtotal:			2.697,40 €

Tabla 10: Precio de los procedimientos de implementación del prototipo



Concepto	Precio
Sistema principal de control	5.520,00 €
Periferia descentralizada	754,56 €
Robot colaborativo + base + pupitre	45.383,77 €
Herramientas de programación	9.383,32 €
Elementos de seguridad	4.333,95 €
Elementos neumáticos	2.201,20 €
Elementos eléctricos	720,30 €
Procedimientos de instalación	2.697,40 €
TOTAL:	70.994,50 €

Tabla 11: Sumatorio de los subtotales para obtener el precio final

Una vez definidos todos los costes que supone la construcción del prototipo, el precio total del mismo alcanzaría la cifra de **setenta mil novecientos noventa y cuatro euros con cincuenta céntimos**.

Cabe destacar que este estudio económico es teórico y mientras que el precio de algunos de los elementos está tomado de facturas reales de la empresa, el precio de los otros elementos ha sido seleccionados de diferentes catálogos online.

NOTA: El sueldo de la mano de obra para el cableado de la estación se ha sacado de presupuestos de otras estaciones y el sueldo del programador sería el sueldo medio por hora de un ingeniero Junior.





CAPÍTULO 3: CONCLUSIONES



3.1.- Conclusiones etapa de estudio

En la etapa de estudio se han investigado los siguientes puntos:

- Estudio de valores mecánicos del robot colaborativo.
- Estudio de la zona de trabajo eficaz del robot colaborativo.
- Estudio de valores de agarre de la pinza colaborativa.
- Diseño de esquemas neumáticos para accionamiento de la pinza.
- Diseño de esquemas eléctricos de la estación.
- Diseño de esquemas de conexión PROFINET/PROFISAFE de todos los elementos de la estación.
- Estudio de puntos de agarre sobre cubiertas plásticas.
- Estudio de elementos de seguridad necesarios para asegurar la categoría 4.

En esta etapa hemos podido comprobar, en primer lugar, que tanto los valores mecánicos del robot colaborativo como de la pinza son adecuados para cumplir el propósito de la estación de trabajo.

En segundo lugar, se han estudiado y definido los parámetros generales de la estación: dimensiones generales de la estación y naturaleza del agarre entre la pinza adaptativa y la cubierta plástica.

Como tercer punto, se han desarrollado los esquemas neumáticos, eléctricos y de comunicaciones de los principales elementos de la estación con el fin de comprender la estructura interna de la estación.

Por último, en este apartado, se han definido las pautas necesarias para que la estación sea considerada de categoría 4 y de esta manera asegurar la cooperación entre operario y robot de forma segura y eficiente.

3.2.-Conclusiones etapa de desarrollo

En la etapa de desarrollo se han realizado los siguientes puntos:

- Simulación del circuito neumático y análisis de los resultados
- Montaje del circuito de comunicación PROFINET de todos los elementos de control en el entorno de TIA Portal.
- Diseño del programa de la rutina del robot.
- Diseño del programa de control del PLC con TIA Portal V15.
- Simulación del programa de PLC mediante Factory IO

Se ha realizado la simulación de la red neumática dónde se ha comprobado el funcionamiento del circuito neumático definido en la etapa de estudio. El objetivo de la simulación era la verificación del funcionamiento del cilindro de doble efecto que es uno de los mecanismos principales de la pinza neumática dejando el apartado de la cámara de aire y comprobación del correcto agarre con diferentes objetos para una prueba real.



Las verificaciones que no se han realizado por medio de simulaciones informáticas se debe a la falta de software para esta pinza al tratarse de una herramienta que supone una novedad en el ámbito industrial.

Se ha realizado una explicación de los fundamentos para crear la red PROFINET entre los elementos de control de la estación y se ha descrito en detalle la programación del autómatas que se encargará del control de la estación y la comunicación bidireccional entre el PLC y el robot. La descripción del programa del PLC se ha realizado con el fin de comprender el funcionamiento interno del PLC y justificar el uso de cada uno de los elementos de la estación.

Se ha llevado a cabo una simulación de la parte del programa del PLC correspondiente al control de flujo de trabajo de la estación. Este control de flujo de trabajo hace alusión a la administración del movimiento de los coches y las cubiertas por sus respectivas cintas transportadoras, definiendo el movimiento de cada una de ellas según las necesidades de la estación. También se ha comprobado el correcto funcionamiento de los estados internos de la estación, comprobando la respuesta de la estación ante posibles paradas de emergencia (error en el robot y activación de seta de emergencia) y su posterior rearme para volver a reanudar el ciclo de trabajo.

Los únicos aspectos que no se han podido simular en el programa del autómatas son la comunicación con el robot, debido a que, al igual que con la simulación neumática, en la actualidad no hay un software específico de simulación que incluya el apartado de comunicación entre PLC y robot; por lo que la comprobación del correcto funcionamiento de la comunicación entre ambos elementos se deberá realizar mediante una prueba física.

Tampoco ha sido posible la comprobación del uso de las señales de salida del PLC para la activación de las electroválvulas debido a que el software de simulación utilizado (Factory IO) no incluye esa característica (elementos neumáticos). No obstante, durante la simulación se ha observado que las señales que activan las electroválvulas funcionan de manera correcta (en el TIA Portal podemos ver durante la simulación el comportamiento a tiempo real de los segmentos que componen el programa). Conociendo que las señales del PLC correspondientes a las electroválvulas se activan de forma correcta y conociendo, gracias a la simulación del circuito neumático, que el circuito neumático es correcto, podemos conjeturar el correcto funcionamiento de este apartado.

En esta etapa de desarrollo se ha definido y explicado el programa del robot con el fin de conocer el funcionamiento interno de este. Todas las sentencias del programa se pueden dividir en 2 grandes grupos: sentencias relacionadas con la trayectoria del robot y sentencias relacionadas con la comunicación con el PLC mediante los registros de Entrada/Salida.

De estos dos grupos únicamente se ha podido comprobar el primero. El software del controlador del robot (PolyScope) incluye una aplicación donde podemos ver una representación gráfica del programa del robot en funcionamiento. Mediante esta aplicación podemos verificar que la trayectoria que realiza el robot es correcta. Por otra parte, al igual que con el PLC, no es posible simular la comunicación entre robot y autómatas. Para comprobar este aspecto es necesario realizar la prueba de estos de manera física.

En cuanto a los elementos de seguridad, se ha podido comprobar el perfecto funcionamiento de todos ellos y la respuesta de la estación a todos ellos. El único aspecto para tener en cuenta



es que no ha sido posible comprobar la señal que debe enviar el robot al PLC en caso de fallo (por las razones descritas anteriormente sobre la simulación de la comunicación entre PLC y robot).

3.3.- Conclusiones generales

A lo largo de esta memoria se han descrito los pasos seguidos para la resolución del problema planteado. Tanto en cada uno de los capítulos dónde se hablaba del desarrollo del proyecto como en los apartados anteriores de este capítulo dónde se hace una referencia a las conclusiones obtenidas a lo largo de la memoria podemos afirmar, con un alto nivel de certeza, que a falta de la comprobación de comunicación entre PLC y el robot colaborativo (se ha realizado un amplio estudio de la naturaleza de la comunicación y podemos conjeturar que la estructura definida tanto en el programa del autómatas y el robot es suficiente para que la comunicación se realice de forma satisfactoria), la estación cumple todos los aspectos requeridos y por consiguiente, se puede construir el prototipo de la misma.

Sin embargo, en la descripción del proyecto, hablamos de que esta estación tiene como objetivo final su implementación en la línea de montaje del Grupo Renault. Esto significa que debe cumplir los estándares y seguir las normas establecidas por la empresa. Algunas de estas normas hacen referencia a la estructura de la programación, y son de obligado cumplimiento. En el desarrollo del proyecto se ha tenido en cuenta, con la mayor consideración posible, estos estándares y se han intentado cumplir con ellos. No obstante, puede que sea necesario hacer ciertas modificaciones con el fin de cumplir todos estos estándares.



CAPÍTULO 4: LÍNEAS FUTURAS



4.1.- Líneas futuras a corto plazo

Como se ha nombrado varias veces a lo largo de la memoria, debido a la situación de la pandemia causada por la COVID-19 las prácticas de empresa administradas por la Universidad de Valladolid fueron canceladas de carácter general, esto ha tenido como consecuencia que para la fecha de entrega de esta memoria no fuera posible cumplir el propósito general de este proyecto, la construcción de un prototipo el análisis de viabilidad. Sin embargo, este objetivo fue sustituido por un estudio de viabilidad y funcionamiento mediante el uso de simulaciones.

Como objetivos a corto plazo, se encuentra la construcción del prototipo tal y como se tenía planeado, no obstante, este proceso se desarrollará con mayor velocidad al tener los programas de control y el programa del robot desarrollados y habiendo realizado una descripción general de todos los elementos que componen la estación, así como los procedimientos a seguir para el correcto funcionamiento de la misma.

Con este prototipo se podrán hacer en primer lugar pruebas sobre el correcto funcionamiento de los procedimientos descritos en esta memoria, centrandose los esfuerzos en el análisis de la función de agarre de la pinza, una vez se hayan confirmado todas las hipótesis planteadas, estudiar la viabilidad y eficiencia del modelo propuesto.

4.2.- Líneas futuras a medio y largo plazo

En cuanto al impacto de este estudio a medio y largo plazo, con el prototipo desarrollado, la empresa habrá observado el potencial de la robótica colaborativa y la mejora de productividad que ofrece una estación colaborativa respecto a un puesto totalmente manual.

En esta memoria se ha descrito las bases de la estructura de datos para la comunicación PLC-robot, se ha definido un procedimiento para establecer conexión entre ambos elementos y se ha desarrollado una guía para la implementación de la pinza colaborativa (esta pinza es la primera pinza adaptativa colaborativa de la factoría de montaje motor).

En resumidas cuentas, los resultados de este proyecto pueden derivar en una inversión por parte de la empresa para introducir la robótica colaborativa de manera generalizada y dar un paso adelante hacia la industria 4.0.



BIBLIOGRAFÍA/WEBGRAFÍA

- [1] Wikipedia, Industria 4.0: https://es.wikipedia.org/wiki/Industria_4.0
- [2] Deloitte, Industria 4.0: <https://www2.deloitte.com/es/es/pages/manufacturing/articles/que-es-la-industria-4.0.html>
- [3] Universal robots, la robótica colaborativa: <https://www.universal-robots.com/es/acerca-de-universal-robots/noticias/historia-de-los-cobots/>
- [4] YouTube, Industria 4.0: una revolución para las persona | Beatriz González | TEDxUDEusto: <https://www.youtube.com/watch?v=a0Ycxn-bZak>
- [5] Control Inteligente, fuente de figura 2: <https://www.controlinteligente.com.mx/robot-colaborativo.php>
- [6] UNE, Norma UNE-EN ISO 13849-1:2016: “Seguridad de las máquinas. Partes de los sistemas de mando relativas a la seguridad. Parte 1: Principios generales para el diseño: <https://www.une.org/encuentra-tu-norma/busca-tu-norma/norma?c=N0057029>
- [7] RS, Seguridad de personas y máquinas (pdf): <https://es.rs-online.com/es/pdf/Schneider.pdf>
- [8] UVA, TABUENCA ALCUSÓN, Diego, et al. Implantación de robots colaborativos en línea de producción. 2017.: <http://uvadoc.uva.es/handle/10324/23076>
- [9] Springer link, METZNER, Maximilian, et al. “Human-In-The-Loop”-Virtual Commissioning of Human-Robot Collaboration Systems. En Tagungsband des 3. Kongresses Montage Handhabung Industrieroboter. Springer Vieweg, Berlin, Heidelberg, 2018. p. 131-138: https://link.springer.com/chapter/10.1007/978-3-662-56714-2_15
- [10] Theseus, DUC BINH, Vo. Robotic Tiling System. 2019: <https://www.theseus.fi/handle/10024/267691>
- [11] Scholarly Commons, Miami University, DYER, Catherine, et al. Universal Robot Gripper. 2020.: <https://sc.lib.muohio.edu/handle/2374.MIA/6622>
- [12] UNE, Norma UNE-EN ISO 11200:2014 “acústica. Ruido emitido por máquinas y equipos. Directrices para la utilización de las normas básicas para la determinación de los niveles de presión acústica de emisión en el puesto de trabajo y en otras posiciones especificadas: <https://www.une.org/encuentra-tu-norma/busca-tu-norma/norma/?c=N0054139>
- [13] Universal Robots, UR10e y fuente de la figura 109: <https://www.universal-robots.com/es/productos/robot-ur10/>
- [14] Wikipedia, nomenclatura IP, grados de protección contra partículas y líquidos: https://es.wikipedia.org/wiki/Grado_de_protecci%C3%B3n_IP
- [15] Universal Robots, manual de usuario del robot colaborativo UR10e y fuente de la figuras 4 y 111 (pdf): https://s3-eu-west-1.amazonaws.com/ur-support-site/69487/99225_UR10_User_Manual_es_Global.pdf



- [16] FESTO, pinza adaptativa DHEF, fuente de figura 102,104: https://www.festo.com/cms/es_es/69457.htm
- [17] Wikipedia, conexión PROFINET: <https://en.wikipedia.org/wiki/PROFINET>
- [18] FESTO, Software de simulación neumática FluidSIM 5: <https://www.festo-didactic.com/es-es/productos/software-e-learning/fluidsim/fluidsim-5.htm?fbid=ZXMuZXMuNTQ3LjE0LjE4LjU5MS43OTc1>.
- [19] Universal Robots, guía de conexión PROFINET con un autómatas de Siemens: <https://www.universal-robots.com/articles/ur-articles/profinet-guide/>
- [20] Centro de formación técnica para la industria, conexión PROFINET: [https://www.cursosaula21.com/profinet-que-es-y-como-funciona/#:~:text=PROFINET%20\(Process%20Field%20Network\)%20es,datos%20entre%20controladores%20y%20dispositivos](https://www.cursosaula21.com/profinet-que-es-y-como-funciona/#:~:text=PROFINET%20(Process%20Field%20Network)%20es,datos%20entre%20controladores%20y%20dispositivos).
- [21] Wikipedia, PROFIBUS: <https://es.wikipedia.org/wiki/Profibus>
- [22] Siemens, importar estructuras de datos a TIA Portal: <https://support.industry.siemens.com/tf/WW/en/posts/plc-data-types-from-s7-pct-to-tia-portal/207841?page=0&pageSize=10>
- [23] S7-1200 controlando robot UR3 por profinet – video 1: https://www.youtube.com/watch?v=hjQwUg_zNVk&t=885s
- [24] Wikipedia, lenguaje Ladder: https://es.wikipedia.org/wiki/Lenguaje_ladder
- [25] Factory I/O, página principal: https://factoryio.com/?gclid=CjwKCAjwnK36BRBVEiwAsMT8WF-CI426dHuVCjt45nD5LSJ6cDPk4GR96c1IMGz1ftUBn219fRUFKxoColgQAvD_BwE
- [26] Factory I/O, Setting up S7-PLSIM V13-16: <https://docs.factoryio.com/tutorials/siemens/setting-up-s7-plcsim-v13/>
- [27] Tecno PLC, SCL lenguaje estructurado de Siemens: <http://www.tecnopl.com/scl-lenguaje-estructurado/>
- [28] Universal Robots, estructura de datos conexión PROFINET entre robot y autómatas (pdf): <https://s3-eu-west-1.amazonaws.com/ur-support-site/20596/pn-iomessage.pdf>
- [29] Universal Robots, e-Series de Universal Robots, fuente de la figura 98: <https://www.universal-robots.com/media/1802610/es-e-series-brochure.pdf>
- [30] PLC City, fuente de la figura 99 : <https://www.plc-city.com/shop/es/siemens-simatic-s7-1500-fail-safe-cpus/6es7516-3fn01-0ab0.html>
- [31] Siemens, documentación técnica del autómatas S7-1516F-3 PN/DP: <https://mall.industry.siemens.com/mall/es/es/Catalog/Product/6ES7516-3FN01-0AB0>



- [32] Schneider Electric, fuente de la figura 100 y documentación técnica de la baliza de señalización: https://www.se.com/es/es/product/XVMB2RAGBSWSB/indicator-bank-with-buzzer-%2B-4-illuminated-units---%C3%B8-45-mm---24-v-ac-dc-%28%28*%29%29/?range=1057-harmony-xvm#characteristics
- [33] SICK, fuente de la figura 101 y documentación técnica del detector de presencia láser: <https://www.sick.com/es/es/dispositivos-de-proteccion-optoelectronicos/escaneres-laser-de-seguridad/s3000-standard/s30a-4011ba/p/p44614>
- [34] TechCXO, fuente de la figura 103, ejemplo de soft gripper: <https://www.techcxo.com/robotic-automation-leader-and-techcxo-client-soft-robotics-raises-20m/>
- [35] Schneider Electric, fuente de figura 105 y documentación técnica de la seta de emergencia: <https://www.se.com/es/es/product/XALK178G/control-station%2C-plastic%2C-yellow-lid%2C-1-red-mushroom-push-button-%C3%B840%2C-turn-to-release%2C-1-no-%2B-2-nc/>
- [36] WI automation, fuente de la figura 106: https://es.wiautomation.com/siemens/modulos/es-distribuidas/6ES71556AA000BN0?SubmitCurrency=1&id_currency=1&gclid=CjwKCAjwnK36BRBVEiwAsMT8WIlx7tl6ocGMSgAyPjdL6HUqAQdeTJSuTU5FmGYpOAYowRSJMxokJRoCMukQAvD_BwE
- [37] Siemens, documentación técnica de la periferia descentralizada ET-200SP (pdf): <https://c4b.gss.siemens.com/resources/images/articles/e20001-a160-p240-x-7800.pdf>
- [38] Eurobots, fuente de la figura 110: <https://www.eurobots.es/robot-ur10e-usado-universal-robots-es.html>





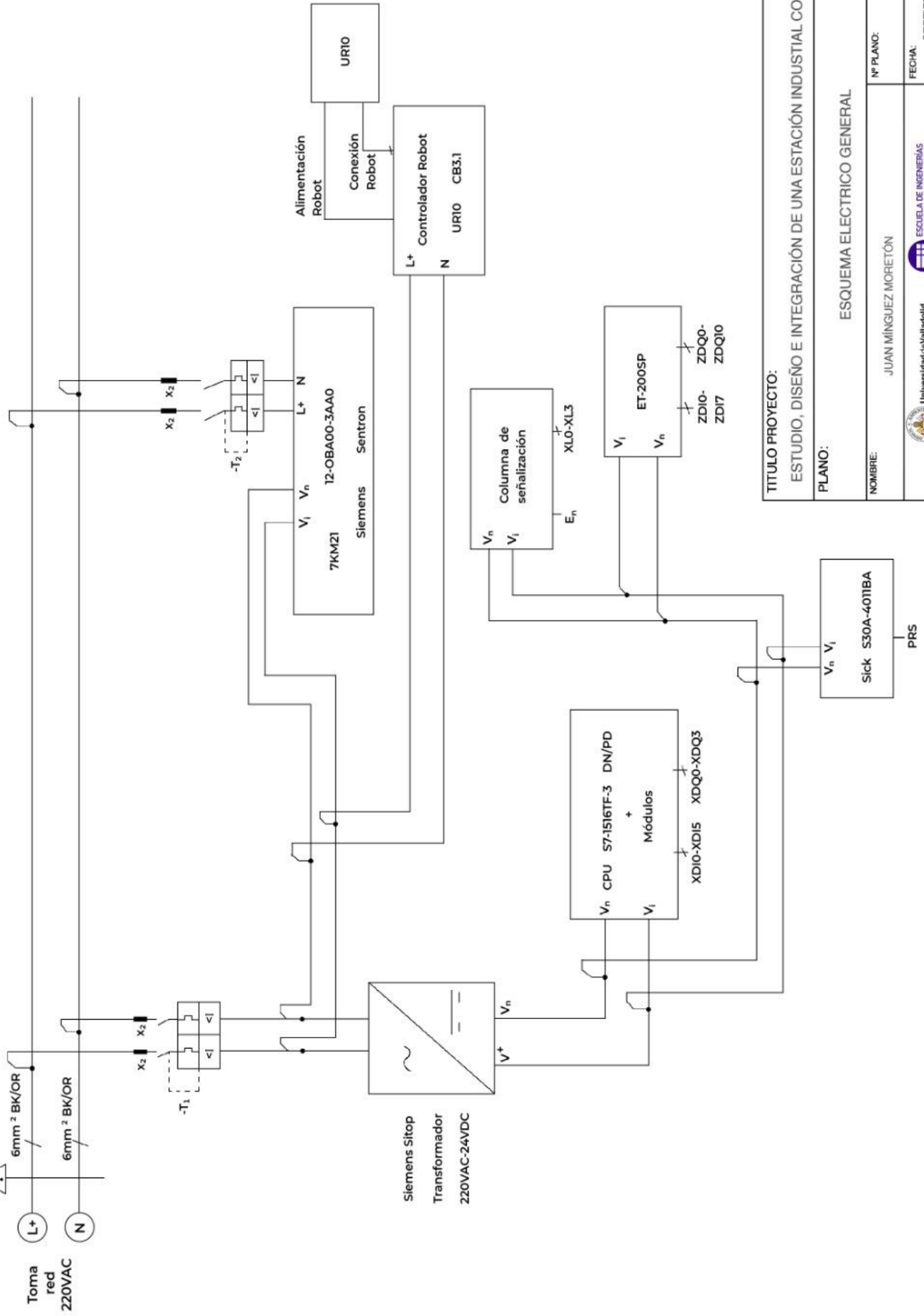
ANEXO I: PLANOS

En este anexo se adjuntan los planos que se han desarrollado para la implementación del prototipo de la estación. En las siguientes páginas encontraremos los planos referentes al cableado de la estación y un plano descriptivo del panel de control del usuario.



Todas las páginas en blanco que se han dejado en este anexo han sido a propósito para mantener una estructura de un plano por hoja, estando este en las hojas impares.

Peligro
alta tensión



TITULO PROYECTO:

ESTUDIO, DISEÑO E INTEGRACIÓN DE UNA ESTACIÓN INDUSTRIAL COLABORATIVA

PLANO:

ESQUEMA ELECTRICO GENERAL

NOMBRE:

JUAN MINGUEZ MORETÓN

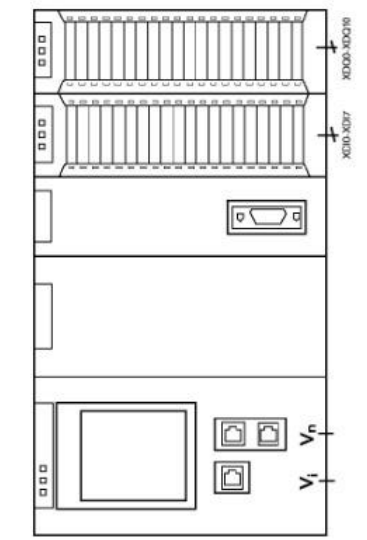
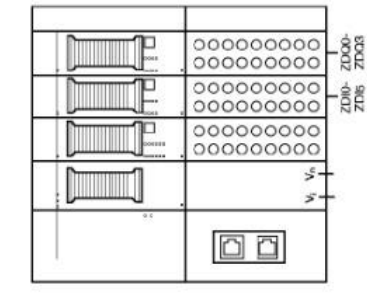
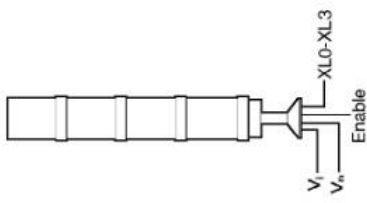
Nº PLANO:

1



ESUELA DE INGENIERIAS
INDUSTRIALES

FECHA:
SEPTIEMBRE-2020



XL

- XL0 _____ %Q0.6
- XL1 _____ %Q1.1
- XL2 _____ %Q1.0
- XL3 _____ %Q1.7

ZDI

- %I52.0 _____ Seta2
- %I52.1 _____ Seta3
- %I52.2 _____ Botón Pausa
- %I52.3 _____ Botón Fin ciclo
- %I52.5 _____ Tope-Cubierta
- %I52.6 _____ Botón reanudar

ZDQ

- %Q50.0 _____ Led Fin
- %Q50.2 _____ Sensor proximidad
- %Q50.4 _____ Conveyor3
- %Q50.5 _____ Conveyor4

XDI

- %I0.0 _____ Pretopo
- %I0.1 _____ Tope
- %I0.2 _____ Z1
- %I0.3 _____ Salida
- %I0.4 _____ Seta1
- %I0.6 _____ Rearme
- %I0.7 _____ Enable
- %I1.0 _____ Marcha

XDQ

- %Q0.0 _____ Conveyor
- %Q0.1 _____ EV1
- %Q0.2 _____ EV2
- %Q0.3 _____ EV3
- %Q0.4 _____ EV4
- %Q0.5 _____ Conveyor2
- %Q0.6 _____ Led Rojo
- %Q0.7 _____ Led Azul
- %Q1.0 _____ Led Verde
- %Q1.1 _____ Led Amarillo
- %Q1.5 _____ Led-Marcha

TITULO PROYECTO:

ESTUDIO, DISEÑO E INTEGRACIÓN DE UNA ESTACIÓN INDUSTRIAL COLABORATIVA

PLANO:

TOPOLOGÍA DE CABLEADO

NOMBRE:

JUAN MINGUEZ MORETÓN

Nº PLANO:

2



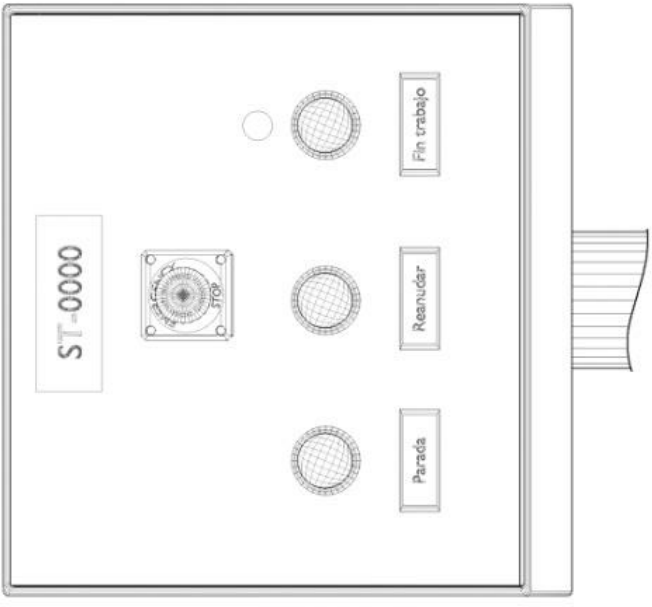
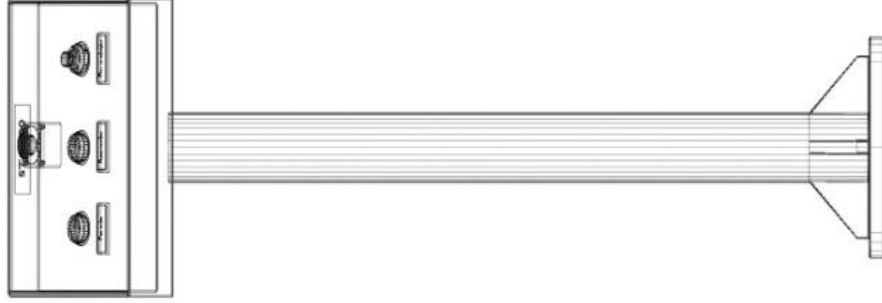
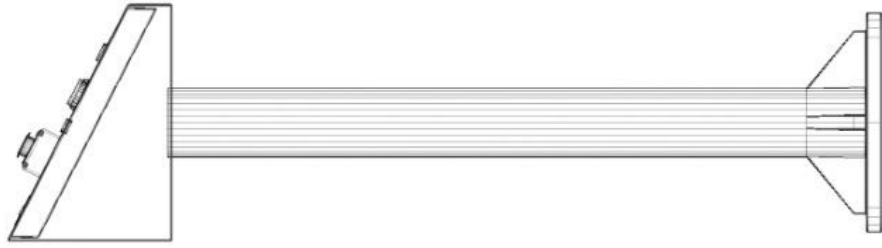
Universidad de Valladolid



ESCUOLA DE INGENIERÍAS INDUSTRIALES

FECHA:

SEPTIEMBRE-2020



TITULO PROYECTO:

ESTUDIO, DISEÑO E INTEGRACIÓN DE UNA ESTACIÓN INDUSTRIAL COLABORATIVA

PLANO:

PANEL DE CONTROL DEL USUARIO

NOMBRE:

JUAN MINGUEZ MORETÓN

Nº PLANO:

3



Universidad de Valladolid



ESCUOLA DE INGENIERÍAS INDUSTRIALES

FECHA:

SEPTIEMBRE-2020



ANEXO II: COMUNICACIÓN PROFINET DEL CONTROLADOR: ESTRUCTURA DE DATOS

En este anexo hablaremos de la estructura de datos que utiliza el controlador del robot en la conexión PROFINET con el PLC. [28]

Es muy importante conocer esta estructura de datos debido a que la conexión entre PLC y nuestro robot es uno de los elementos más importantes del desarrollo de esta estación junto con la trayectoria del brazo robótico y el diseño del programa de control del PLC.

Antes de explicar la tabla de todas las variables que aparecen en la transmisión de datos, vamos a definir una serie de variables que vienen definidas por sus siglas (Figura 86):

AOM	Analog output mask
AOT	Analog output types
ES	Is emergency stopped
FT	Is fault
NO	Is normal mode
PB	Is power button pressed
PR	Is program running
PS	Is protective stopped
PW	Is power on
RC	Is recovery mode
RD	Is reduced mode
RES	Is robot emergency stopped
SES	Is system emergency stopped
SS	Is safeguard stopped
SSM	Speed slider fraction mask
ST	Is stopped due to safety
TAIT	Tool analog input types
TB	Is teach button pressed
TDI	Tool digital inputs
TDO	Tool digital outputs
TDOM	Tool digital output mask
VL	Is violation

Figura 85: Tabla de variables definidas por sus siglas [28]

Podemos obtener una gran cantidad de información del estado del robot; desde comprobar en qué modo de funcionamiento se encuentra el robot, hasta comprobar si hay alguno de los botones de la TeachPendant pulsado.

Esta estructura de datos nos interesa especialmente en la programación del TIA Portal debido a que es el propio PLC el encargado de las tareas de control y necesita cierta información del controlador del cobot. Por su parte, la información que recibe el robot procedente del autómata son una serie de señales binarias con las que el robot actuará de manera determinada según la señal que se active. Más adelante veremos las variables que utilizaremos en esta estación.



TIA Portal tiene una serie de clases de variables predefinidas, donde podemos enumerar variables de tipo binario (Bool), enteros (UInt), datos de tiempo (Time), etcétera. No obstante, para el uso de los datos que nos ofrece el robot debemos utilizar un tipo de variable que no aparece en el catálogo de TIA Portal, por lo que deberemos instalar esa librería. En el apartado 2.7.1.- “Descripción de red Hardware” podemos encontrar el tutorial de cómo importar ese archivo de estructura de datos.

Para entender la tabla de estructura de datos, debemos indicar que el robot se denomina con la letra ‘T’ y el autómatas con la letra ‘O’. Otro aspecto que hay que mencionar es que este grupo de variables se define en dos grupos principales: UR_T2O (Señales del robot al autómatas) y UR_O2T (Señales del autómatas al robot). Dentro de estos dos tipos de variables podemos encontrar toda la estructuración de los datos (Figura 87).

29	URI	*UR_T2O*	%I53.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
30	State	UR_1_T2O_State	%I53.0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
31	Robot	UR_T2O_Robot	%I53.0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
32	Safety	UR_T2O_Safety	%I77.0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
33	IO	UR_2_T2O_IO	%I85.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
34	IOs	UR_T2O_IOs	%I85.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
35	Tool	UR_T2O_Tool	%I129.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
36	Joints	UR_3_T2O_Joints	%I153.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
37	Joint position [rad]	Array[0..5] of R...	%I153.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
38	Joint velocity [rad/s]	Array[0..5] of R...	%I177.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
39	Joint current [A]	Array[0..5] of R...	%I201.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
40	Joint temperature [°C]	Array[0..5] of R...	%I225.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
41	Joint mode	Array[0..5] of U...	%I249.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
42	Reserved_9	UInt	%I255	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
43	TCP	UR_4_T2O_TCP	%I257.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
44	TCP Position (X,Y,Z) [m]	Array[0..2] of R...	%I257.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
45	TCP Position (RX,RY,RZ) [...]	Array[0..2] of R...	%I269.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
46	TCP velocity (X,Y,Z) [m/s]	Array[0..2] of R...	%I281.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
47	TCP velocity (RX,RY,RZ) [...]	Array[0..2] of R...	%I293.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
48	TCP force (X,Y,Z) [N]	Array[0..2] of R...	%I305.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
49	TCP torque (X,Y,Z) [Nm]	Array[0..2] of R...	%I317.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
50	TCP force scalar [N]	Real	%ID329	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
51	Bits	UR_5_T2O_BitR...	%I333.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
52	Register	Array[0..63] of ...	%I333.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
53	Register[0]	Bool	%I333.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
54	Register[1]	Bool	%I333.1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
55	Register[2]	Bool	%I333.2	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
56	Register[3]	Bool	%I333.3	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
57	Register[4]	Bool	%I333.4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
58	Register[5]	Bool	%I333.5	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

Figura 86: Estructura de la variable tipo “UR_T2O”

También debemos tener en cuenta que los valores de los bits de esta tabla hacen referencia a las direcciones absolutas dónde se encuentran dichas variables. En otras palabras, observando la tabla anterior podemos observar que la variable “URI” que es de tipo “UR_T2O” está asignada en el bit 53.0, por lo que el dato en la dirección de memoria de la tabla correspondiente al bit 0 se encuentra en el bit I53.0 de nuestro autómatas y así con toda la estructura de los datos T2O. Lo mismo pasaría con los datos O2T que partirían del bit que se haya asignado a la variable de tipo “UR_O2T”.

En las figuras 88-97 encontramos la estructura de datos. Es un mapa gráfico dónde cada fila corresponde a 32 bits y el principio de cada fila vemos el bit por el que empezamos dicha fila. A la derecha vemos el tipo de dato al que corresponde cada grupo de variables.



UR 1 T20 State : T (Robot) -> O (PLC) - Robot state and safety mode (32 bytes)

Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	Group
0	Controller major version (uint)							Controller minor version (uint)							Reserved								Robot UR_T20_Robot										
32	Robot mode (uint)							Real time machine seconds (uint)							Real time machine milliseconds (uint)																		
64	Real time machine minutes (uint)							Real time machine hours (uint)							Real time machine days (uint)																		
96	Robot current (float) [A]																Reserved																
128	PW	PR	TB	PB	Reserved																											Safety UR_T20_Safety	
160	Speed slider fraction (float)																Reserved																
192	Safety mode (uint)																Reserved																
224	NO	RD	PS	RC	SS	SES	RES	ES	VL	FT	ST	Reserved																					

Figura 87: Estructura de datos T20 bits 0 a 255 [28]

UR 2 T20 IO : T (Robot) -> O (PLC) - I/Os of control box and tool (68 bytes)

Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	Group
256	Standard digital inputs							Configurable digital inputs							Standard digital outputs							Configurable digital outputs							I/Os UR_T20_Ios				
288	Analog I/O Types																																
320	Reserved																																
352	Standard analog input 0 (float) [A or V]																																
384	Standard analog input 1 (float) [A or V]																																
416	Standard analog output 0 (float) [A or V]																																
448	Standard analog output 1 (float) [A or V]																																
480	I/O current (float) [A]																																
512	Euromap67 input bits																																
544	Euromap67 output bits																																
576	Euromap67 24V voltage (float) [V]																																
608	Euromap67 24V current (float) [A]																																
640	Tool mode (uint)							Reserved							TDO							Reserved								Tool UR_T20_Tool			
672	TDI	Reserved						Reserved							TAIT							Reserved											
704	Tool analog input 0 (float) [A or V]																																
736	Tool analog input 1 (float) [A or V]																																
768	Tool output voltage (float) [V]																																
	Tool current (float) [A]																																

Figura 88: Estructura de datos T20 bits 256-799 [28]

UR 3 T20 Joints: T (Robot) -> O (PLC) - Joint states (104 bytes)

Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	Group
800	Joint 0 position (float) [rad]																																
832	Joint 1 position (float) [rad]																																
864	Joint 2 position (float) [rad]																																
896	Joint 3 position (float) [rad]																																
928	Joint 4 position (float) [rad]																																
960	Joint 5 position (float) [rad]																																
992	Joint 0 velocity (float) [rad/s]																																
1024	Joint 1 velocity (float) [rad/s]																																
1056	Joint 2 velocity (float) [rad/s]																																
1088	Joint 3 velocity (float) [rad/s]																																
1120	Joint 4 velocity (float) [rad/s]																																
1152	Joint 5 velocity (float) [rad/s]																																
1184	Joint 0 current (float) [A]																																
1216	Joint 1 current (float) [A]																																
1248	Joint 2 current (float) [A]																																
1280	Joint 3 current (float) [A]																																
1312	Joint 4 current (float) [A]																																
1344	Joint 5 current (float) [A]																																
1376	Joint 0 temperature (float) [°C]																																
1408	Joint 1 temperature (float) [°C]																																
1440	Joint 2 temperature (float) [°C]																																
1472	Joint 3 temperature (float) [°C]																																
1504	Joint 4 temperature (float) [°C]																																
1536	Joint 5 temperature (float) [°C]																																
1568	Joint 0 mode (uint)							Joint 1 mode (uint)							Joint 2 mode (uint)							Joint 3 mode (uint)							Joints				
1600	Joint 4 mode (uint)							Joint 5 mode (uint)							Reserved																		

Figura 89: Estructura de datos T20 bits 800-1631 [28]

UR 4 T20 TCP: T (Robot) -> O (PLC) - TCP (76)

Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	Group
1632	TCP position X (float) [m]																																
1664	TCP position Y (float) [m]																																
1696	TCP position Z (float) [m]																																
1728	TCP position RX (float)																																
1760	TCP position RY (float)																																
1792	TCP position RZ (float)																																
1824	TCP velocity X (float) [m/s]																																
1856	TCP velocity Y (float) [m/s]																																
1888	TCP velocity Z (float) [m/s]																																
1920	TCP velocity RX (float)																																
1952	TCP velocity RY (float)																																
1984	TCP velocity RZ (float)																																
2016	TCP force X (float) [N]																																
2048	TCP force Y (float) [N]																																
2080	TCP force Z (float) [N]																																
2112	TCP torque X (float) [N.m]																																
2144	TCP torque Y (float) [N.m]																																
2176	TCP torque Z (float) [N.m]																																
2208	TCP force scalar (float) [N]																																

Figura 90: Estructura de datos T20 bit 1632-2239 [28]



UR 5 T2O BitRegisters : T (Robot) -> O (PLC) - General purpose bit registers (8 bytes)

Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	Group
2240	Bit output registers 0 - 31																														Bit registers		
2272	Bit output registers 32 - 63																																

Figura 91: Estructura de datos T2O bits 2240-2303 [28]

UR 6 T2O IntRegisters : T (Robot) -> O (PLC) - General purpose int registers (96 bytes)

Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	Group
2304	Int output register 0 (int)																														Int registers		
2336	Int output register 1 (int)																																
2368	Int output register 2 (int)																																
2400	Int output register 3 (int)																																
2432	Int output register 4 (int)																																
2464	Int output register 5 (int)																																
2496	Int output register 6 (int)																																
2528	Int output register 7 (int)																																
2560	Int output register 8 (int)																																
2592	Int output register 9 (int)																																
2624	Int output register 10 (int)																																
2656	Int output register 11 (int)																																
2688	Int output register 12 (int)																																
2720	Int output register 13 (int)																																
2752	Int output register 14 (int)																																
2784	Int output register 15 (int)																																
2816	Int output register 16 (int)																																
2848	Int output register 17 (int)																																
2880	Int output register 18 (int)																																
2912	Int output register 19 (int)																																
2944	Int output register 20 (int)																																
2976	Int output register 21 (int)																																
3008	Int output register 22 (int)																																
3040	Int output register 23 (int)																																

Figura 92: Estructura de datos T2O bits 2304-3071 [28]

UR 7 T2O FloatRegisters : T (Robot) -> O (PLC) - General purpose float registers (96 bytes)

Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	Group
3072	Float output register 0 (float)																														Float registers		
3104	Float output register 1 (float)																																
3136	Float output register 2 (float)																																
3168	Float output register 3 (float)																																
3200	Float output register 4 (float)																																
3232	Float output register 5 (float)																																
3264	Float output register 6 (float)																																
3296	Float output register 7 (float)																																
3328	Float output register 8 (float)																																
3360	Float output register 9 (float)																																
3392	Float output register 10 (float)																																
3424	Float output register 11 (float)																																
3456	Float output register 12 (float)																																
3488	Float output register 13 (float)																																
3520	Float output register 14 (float)																																
3552	Float output register 15 (float)																																
3584	Float output register 16 (float)																																
3616	Float output register 17 (float)																																
3648	Float output register 18 (float)																																
3680	Float output register 19 (float)																																
3712	Float output register 20 (float)																																
3744	Float output register 21 (float)																																
3776	Float output register 22 (float)																																
3808	Float output register 23 (float)																																

Figura 93: Estructura de datos T2O bits 3072-3839 [28]

UR 8 O2T Robot IO : O (PLC) -> T (Robot) - Speed slider and electrical outputs (24 bytes)

Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	Group									
0	SSM																														Robot											
32	Reserved																															UR_O2T_Robot										
64	Standard digital output mask										Configurable digital output mask										Standard digital outputs					Configurable digital outputs					Outputs											
96	AOM					Reserved					AOT					Reserved					TDOM					Reserved						TDO					Reserved					UR_O2T_IOs
128	Standard analog output 0 (float) [0..1]																																									
160	Standard analog output 1 (float) [0..1]																																									

Figura 94: Estructura de datos O2T bits 0-191 [28]



UR 9 O2T Req1 : O (PLC) -> T (Robot) - General Purpose registers (100 bytes)

Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	Group	Data type (.udt file)
192	Bit input registers 0 - 31																															Bit registers		
224	Int input register 0 (int)																															Int registers UR_O2T_ints		
256	Int input register 1 (int)																																	
288	Int input register 2 (int)																																	
320	Int input register 3 (int)																																	
352	Int input register 4 (int)																																	
384	Int input register 5 (int)																																	
416	Int input register 6 (int)																																	
448	Int input register 7 (int)																																	
480	Int input register 8 (int)																																	
512	Int input register 9 (int)																																	
544	Int input register 10 (int)																																	
576	Int input register 11 (int)																																	
608	Float input register 0 (float)																															Float registers UR_O2T_floats		
640	Float input register 1 (float)																																	
672	Float input register 2 (float)																																	
704	Float input register 3 (float)																																	
736	Float input register 4 (float)																																	
768	Float input register 5 (float)																																	
800	Float input register 6 (float)																																	
832	Float input register 7 (float)																																	
864	Float input register 8 (float)																																	
896	Float input register 9 (float)																																	
928	Float input register 10 (float)																																	
960	Float input register 11 (float)																																	

Figura 95: Estructura de datos O2T bits 192-991 [28]

UR 10 O2T Req2 : O (PLC) -> T (Robot) - General Purpose registers (100 bytes)

Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	Group	Data type (.udt file)
992	Bit input registers 32 - 63																															Bit registers		
1024	Int input register 12 (int)																															Int registers UR_O2T_ints		
1056	Int input register 13 (int)																																	
1088	Int input register 14 (int)																																	
1120	Int input register 15 (int)																																	
1152	Int input register 16 (int)																																	
1184	Int input register 17 (int)																																	
1216	Int input register 18 (int)																																	
1248	Int input register 19 (int)																																	
1280	Int input register 20 (int)																																	
1312	Int input register 21 (int)																																	
1344	Int input register 22 (int)																																	
1376	Int input register 23 (int)																																	
1408	Float input register 12 (float)																															Float registers UR_O2T_floats		
1440	Float input register 13 (float)																																	
1472	Float input register 14 (float)																																	
1504	Float input register 15 (float)																																	
1536	Float input register 16 (float)																																	
1568	Float input register 17 (float)																																	
1600	Float input register 18 (float)																																	
1632	Float input register 19 (float)																																	
1664	Float input register 20 (float)																																	
1696	Float input register 21 (float)																																	
1728	Float input register 22 (float)																																	
1760	Float input register 23 (float)																																	

Figura 96: Estructura de datos O2T bits 992-1791 [28]

Para nuestro sistema vamos a utilizar el bit 234 correspondiente a ST (Is Stopped due to safety) que nos indica si se ha producido algún error en el robot a lo largo de la rutina establecida. Al corregirse dicha falla, esta variable vuelve a su valor 0. Para el resto de las comunicaciones, como se trata de una estación que no tiene un ciclo demasiado complejo ni se necesita gran cantidad de información del robot, se utilizarán los registros de entrada y salida que el robot tiene instalados.

Utilizando los bits del 2240 hasta el 2271 para la comunicación de mensajes del robot al PLC, y los bits del 192 al 223 para la comunicación de señales del PLC al robot. En la tabla 2 podemos ver la asignación de estos bits a las variables que deseamos que formen parte de la comunicación.





ANEXO III: COMPONENTES DE LA ESTACIÓN

En este anexo se hará un desarrollo de los elementos principales que componen la estación y que aparecen a lo largo de esta memoria.

Robot colaborativo de Universal Robot: UR10 (Figura 98)

“En una época de continuo avance tecnológico y aumento de la competencia, los fabricantes quieren impulsar la productividad, mejorar la calidad del producto y proporcionar una mayor satisfacción de los equipos de trabajo. Las empresas buscan constantemente formas de innovar para crecer y ser más competitivas en el futuro” [29].

El robot colaborativo (también conocidos como ‘cobots’) UR10 es uno de los modelos más versátiles del mercado. Este cobot de la empresa danesa Universal Robot es capaz de integrarse sin mayor dificultad en todos los entornos de producción. Cuando hablamos de colaboración, hacemos referencia a la estrecha proximidad con la que estos cobots trabajan junto con los operarios, siendo capaces de asegurar un entorno seguro para ambos.

Otra gran ventaja es la facilidad de los entornos de programación de estos cobots, según la propia empresa “solo necesitamos 87 minutos para aprender a utilizar el entorno de programación y este nos ofrece infinitas posibilidades al alcance de todos”.

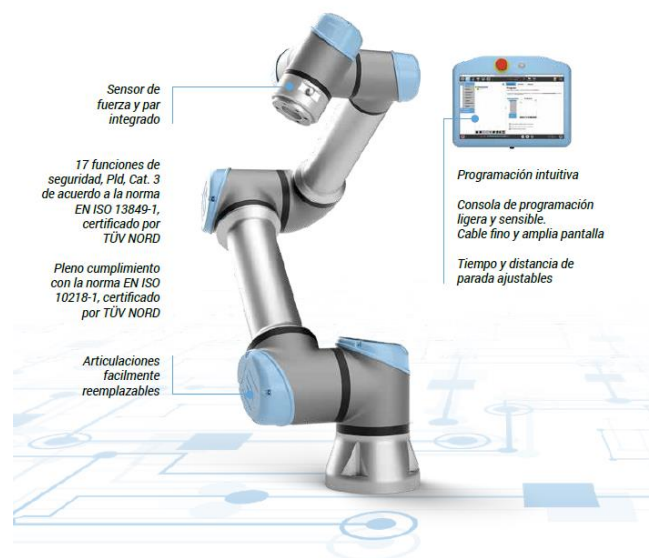


Figura 97: Robot colaborativo UR10 [29]

El robot colaborativo UR10 es el modelo más grande actualmente. Como su nombre indica, su carga máxima es de 10kg y tiene un alcance efectivo de 1300mm. Es un brazo robótico de 6 articulaciones (base, hombro, codo, muñeca 1, muñeca 2 y muñeca 3) que cuenta con sensores de par y fuerza en cada uno de sus eslabones con los que es capaz de detectar cualquier tipo de colisión o movimiento forzado. Para el funcionamiento cuenta con un controlador con capacidad para introducir entradas y salidas digitales, analógicas y de seguridad; así como conexiones PROFINET para conexión con autómatas programables.



Cada robot tiene un número de serie y una tarjeta (CB-Series) o USB (E-Series) única que contiene los parámetros de calibración de ese brazo específico. A diferencia de robots de otras compañías, el controlador es universal; es decir, que se puede utilizar un mismo controlador con diferentes robots, pero necesita tener introducida la tarjeta (o USB) específico del robot a controlar.

Conectada al controlador, encontramos también la TeachPendant. Una pantalla táctil desde la cual podemos administrar el funcionamiento del robot, desde la configuración de los parámetros de seguridad, hasta la creación de un programa para el robot; pasando por el control de movimiento de robot eje a eje (jogging) y salvaguarda/carga de programa gracias a un puerto USB situado en un lateral de la Teach Pendant. El entorno desarrollado por Universal Robots es muy intuitivo y simple, lo que permite una curva de aprendizaje suave. En la parte trasera de la TeachPendant podemos encontrar la “Palanca del hombre muerto”, un botón que nos permite el movimiento manual del robot cuando estamos en el modo de movimiento eje a eje.

Este cobot es la mejor opción para llevar a cabo nuestro proyecto de una estación colaborativa debido al alcance de trabajo y las cargas tanto de la herramienta como de los componentes que va a sostener.

Para más información, podemos acceder a la página de Universal Robots, donde encontramos gran cantidad de documentación y fichas técnicas. La política de la empresa permite acceder a toda la documentación sobre robots colaborativos, así como herramientas de software. [13]

Autómata programable de seguridad SIEMENS S7-1516F (Figura 99)

El autómata programable (generalmente conocidos como “PLC”) S7-1516F es un dispositivo de la empresa Siemens destinado al control de procesos (generalmente industriales). La serie S7-1500 es la última que ha salido al mercado (siendo las más conocidas las series S7-300 y S7-1200) y cuenta con grandes innovaciones destacando en primer lugar, la introducción de una pantalla LCD con la que podemos ver y acceder a diferentes funcionalidades.



Figura 98: PLC Siemens S7-1516F [30]



En otros apartados hablaremos del carácter modular de este tipo de PLC, así como los diferentes módulos que utilizaremos para el proyecto; no obstante, ahora hablaremos únicamente de este módulo.

El S7-1516F es un autómata programable de seguridad (como indica la F al final). Suele ir con varios módulos acoplados en la misma fila e incluso varias filas controladas por una única CPU, por lo que es el 'cerebro' de esta estructura de control. En él se encuentra almacenado el programa creado y es el encargado de su ejecución, análisis de las señales procedentes del módulo de entrada y generación las salidas que irán al módulo de salida. Tiene 4 puertos PROFINET/PROFISAFE divididos en 2 grupos, cada grupo tiene una IP que puede ser editada y en cada grupo 2 puertos para conexión de un cable Ethernet (RJ-45) y un puerto para conexión PROFIBUS [31].

Abriendo la tapa de la pantalla LCD (que es extraíble) podemos encontrar unos LEDs de estado de las conexiones PROFISAFE/PROFINET, un interruptor para el cambio manual entre los estados RUN, STOP y M-RES (reseteo manual).

Tanto la configuración de la CPU, como el diseño del programa se realiza con el software de Siemens conocido como TIA Portal y necesitando la extensión STEP SAFETY 7 para acceder a las características de seguridad.

Esta CPU cuenta con una memoria de trabajo de 1.5Mb para almacenamiento del programa y 5Mb para almacenamiento de datos, aunque cuenta con un puerto para añadir una tarjeta de memoria de hasta 32Gb. Necesita una alimentación 24V que obtendrá con un módulo de potencia, del cual hablaremos en otros apartados.

Como hemos nombrado con anterioridad, esta serie cuenta en su tapa con una pantalla LCD de 6.1 cm de diagonal y 6 teclas para poder navegar por los menús.

Al ser una CPU de seguridad, podemos configurar una contraseña para bloquear la lectura/escritura del autómata y podremos utilizar PROFISAFE para la comunicación con los diferentes elementos de seguridad de la estación.

Para más información podemos acceder a la página de oficial Siemens, donde encontraremos la ficha técnica junto con una serie de manuales de funcionamiento y seguridad. [31]

Columna de señalización de 4 colores: XVMB2RAGBSWSB (Figura 100).

En un ciclo industrial es muy importante detectar rápidamente el estado de una estación. Si hay algún fallo y la estación deja de funcionar, dicho fallo no permite que el producto continúe por la línea de montaje perdiendo capacidad de producción en toda la línea. Mediante las balizas (o columnas) de señalización, situadas en una zona visible, podemos observar de manera rápida y sencilla el estado en el que se encuentra una estación.

La baliza que vamos a utilizar en esta estación es una columna de señalización de 45mm de diámetro con 4 colores y alimentación de 24V AC/DC. Para ser más específicos, utilizaremos el modelo Harmony XVM de la serie de balizas de señalización de Schneider Electric.

Este modelo cuenta con protección IP54 contra polvo y agua, acorde con la IEC 60529 y su nivel de ruido se encuentra en un rango de 0...80dB en 1 m. Este modelo cuenta con las etiquetas Green Premium, REACh y RoHS UE sobre sostenibilidad ambiental y es libre de mercurio.



Cuenta con 4 cables de entrada para enviar la señal a las diferentes luces, un cable de alimentación de la baliza y un cable de salida que indica si todo funciona correctamente



Figura 99 : Columna de señalización [32]

Otro aspecto muy importante a la hora de incluir una baliza de señalización es el código de colores de esta. Como esta estación no está aislada, sino que se va a encontrar dentro de una línea de montaje, el código de colores debe ser universal para todas las estaciones:

- **Rojo:** indica que se ha producido un fallo en la estación, generalmente está producido por el robot
- **Amarillo:** Indica que ha habido una parada de emergencia, ya sea por la pulsación de una seta de emergencia, o una invasión a una zona no permitida detectada por el sensor de presencia.
- **Verde:** Indica que la estación está funcionando con normalidad.
- **Azul:** Indica que hay saturación en esta estación, es decir, que hay una cantidad de coches a la espera que provoca una reducción en la capacidad de producción de la línea. Esta señal es compatible con las tres anteriores.

Para más información de este componente, en la página de Schneider Electric referente a este producto, se puede acceder a la hoja de datos de la baliza y consultar precio entre otras cosas [32].

Detector de presencia por láser: Sick S30A-4011BA (Figura 101).

Junto a las setas de emergencia, debemos añadir un sensor capaz de detectar la presencia de algún intruso en una zona limitada para el libre movimiento del robot. Esa zona limitada se trata de aquella en la que el robot realiza la trayectoria de recoger la pieza. Esa acción no es potencialmente peligrosa, no obstante, es una medida de prevención.

El detector escogido, funciona con tecnología láser, abarca un ángulo de escaneo de 190º y a con un margen de medida de hasta 49m. De esta manera, si a lo largo de la ejecución normal de la estación se produce una intrusión en esta zona, el sensor se dispara (señal digital de 50mA y



24V DC) e indica la presencia de un intruso en la zona limitada, activando el estado de emergencia y parando el ciclo de la estación. La alimentación de este sensor esta entre 16.8V DC y 28.8V DC, siendo su valor nominal 24.0V DC [33].



Figura 100: sensor de presencia [33]

Podemos encontrar más información del producto en la página de SICK. [33]

Pinza neumática adaptativa de FESTO: DHEF-20-A (Figura 104)

Una de las restricciones más importantes de la robótica colaborativa es que el robot no tenga partes afiladas o que puedan provocar cortes, y que la herramienta con la que trabaja este, no pueda provocar daños graves en el operario. En el ámbito de las pinzas, lo más común son pinzas de 2 o 3 dedos accionadas neumática o eléctricamente. Estas pinzas no cuentan con sensores de fuerza, por lo que pueden provocar atrapamientos que pueden causar daños graves en el operario.

En cuanto al ámbito colaborativo y a la manipulación de elementos no rígidos o delicados, ha surgido en los últimos años lo que se conoce con el término inglés 'soft grippers'. Estos soft grippers son generalmente de accionamiento neumático y las superficies de agarre suelen estar hechas de cámaras de silicona. Con estas cámaras de silicona tratan de reproducir el comportamiento de algunas partes de animales, como pueden ser las ventosas y los tentáculos de un pulpo, o la lengua retráctil de los anfibios y camaleones.



Figura 102: Ejemplo de soft gripper [34]

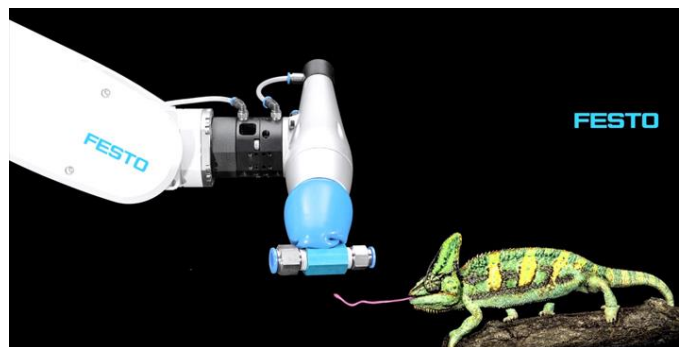


Figura 101: Similitud de la pinza con lengua de un camaleón [16]



En este proyecto, el robot se encargará de recoger cubiertas plásticas, por lo que necesitaremos una pinza de ventosa. Hemos decidido el uso de la pinza adaptativa con accionamiento neumático DHEF-20-A de FESTO, debido a la versatilidad y eficiencia a la hora de agarrar superficies lisas.

La pinza DHEF-20-A consta de un actuador neumático y una tapa de sujeción de forma cuyo extremo está conectado con el vástago interno. El movimiento del cilindro contrae el extremo de la tapa hacia dentro, lo cual permite agarrar objetos durante el movimiento de inversión. En el lateral del cilindro cuenta con unas ranuras en T en las que podemos introducir unos sensores de posición para un mayor control del movimiento del brazo [16].

La pinza necesita 3 entradas neumáticas las cuales podemos conectar mediante las válvulas que se encuentran en el lateral del cilindro, o bien podemos conectarlas directamente al brazo robótico, mediante 3 orificios en la zona de anclado del cilindro. Sobre la base del cilindro cuenta con 7 taladros de M6 que facilita la instalación sobre el brazo robótico.



Figura 103: Pinza adaptativa [16]

El funcionamiento neumático se basa en un vástago de 2 estados, cuyo extremo se encuentra anclado a la cámara de aire. Dos de las tomas neumáticas se encargan del movimiento del vástago y la tercera se encarga de mantener y liberar la presión de la cámara de aire.

Para recoger un objeto, avanza el vástago y se llena la cámara de aire, bloqueamos la presión dentro de la cámara y cuando nos acercamos al objeto, al retroceder el vástago y mantener la presión de la cámara de aire, se produce una depresión que provoca un efecto ventosa y nos permite agarrar el objeto. La diferencia con respecto a las ventosas convencionales es que esta cámara de aire se adapta al objeto que se quiere recoger (tornillos, tuercas, frutas, y un amplio etcétera) pudiendo recoger objetos de diferentes tamaños y formas sin necesidad de cambiar de herramienta.

Para más información sobre esta pinza, podemos acceder a la página oficial de FESTO, donde podemos ver información de la herramienta, fichas técnica y videos sobre el funcionamiento y la versatilidad de la pinza.[16]



Seta de emergencia con enclavamiento y giro para restablecimiento XALK178G (figura 105)

El otro elemento tipo esencial de seguridad para la estación son las setas de emergencia. Las setas de emergencia son unos pulsadores que se sitúan en la estación para que en caso de emergencia o necesidad de para urgente y se pulsan para que la estación entre en estado de emergencia y se pare el ciclo de trabajo. Según la norma EN/IEC 60204-1 estas setas de emergencia deben contar con un enclavamiento mecánico que se restaura mediante el giro en sentido de las agujas del reloj de la seta [35].

Aludiendo a la misma norma, se establece que estas setas de emergencia tengan unos colores que sean fácilmente distinguibles. Este modelo cuenta con un pulsador tipo seta de color rojo sin marcar y un panel base amarillo brillante RAL 1016 y gris claro RAL 7035.

Pese a que no es una norma de obligado cumplimiento, en las aplicaciones industriales se suele utilizar lógica negativa con este tipo de dispositivo. En caso de que la seta quede en desconexión, debido a una falta de alimentación, por ejemplo, el sistema detectará como que la seta esta pulsada.

La seta de emergencia que vamos a utilizar se trata del modelo XALK178G de Schneider que cuenta con protección IP66, IP67, IP69, IP69K y NEMA4.



Figura 104: Seta de emergencia [35]

Para más información podemos acceder a la página web de SE [35].

Módulo periferia descentralizada ET200SP (Figura 106)

En la mayoría de las aplicaciones industriales, los esquemas de comunicación son muy extensos debido a la gran cantidad de elementos que necesitan ser controlados (sensores, elementos de seguridad, semáforos de estado de operación, herramientas del brazo robótico, etc...). Debido a esto, y a que el PLC suele estar dentro del armario eléctrico es muy común el uso de elementos descentralizados, como pueden ser los conmutadores.



Estos elementos pueden ser muy útiles debido a que pueden ser situados dentro de la propia estación (por ejemplo, pueden estar anclados en el propio brazo robótico). Esta periferia descentralizada se conecta mediante PROFINET al PLC y supone una vía bidireccional tanto el envío de información de las entradas al PLC, como la recepción de señales de actuación de este.



Figura 105: Módulo ET200SP [36]

Para este proyecto utilizaremos el módulo ET-200SP. En apartados posteriores hablaremos de los módulos que van a ser utilizados junto a este ET-200SP, no obstante, aquí se hará una descripción de este dispositivo.

El ET-200SP es un sistema de periferia escalable que se caracteriza por su configuración variable y su diseño muy compacto, con capacidad de cableado independiente con conexión a uno o varios hilos. Su tamaño reducido, su variabilidad gracias a la escalabilidad en el número de módulos acoplados y su protección IP67, nos permite que sea utilizado en infinidad de aplicaciones industriales [37].

El dispositivo cuenta con dos puertos Ethernet (RJ-45) que permite una conexión al PLC o a otros dispositivos que formen parte de la red de control. Este dispositivo cuenta con una única IP que puede ser configurada mediante software (por ejemplo, mediante el TIA Portal).

Para más información, podemos acceder a la ficha técnica que aparece en la página oficial de Siemens, dónde podemos encontrar a mayores, las fichas técnicas de otros modelos de periferia descentralizada [37].



ANEXO IV: HARDWARE DEL SISTEMA DE CONTROL DE LA ESTACIÓN

En este quinto anexo hablaremos del sistema de control a nivel Hardware. Haremos una descripción detallada de cada uno de los componentes conectados entre sí que configuran nuestro sistema de control de la estación.

Sistema de control principal: PLC S7-1500F + Módulos de entrada/Salida (Figura 107)

En este apartado haremos una descripción de la topología y funcionamiento de la fila que irá en el cuadro eléctrico formada por la CPU primaria y sus módulos.

El elemento principal de este conjunto de módulos es la CPU S1516-F, la cual hemos hablado en apartados anteriores en el anexo anterior. Es el encargado de ejecutar el programa de control, almacenamiento de datos y es el maestro de la comunicación con el resto de los módulos y la periferia descentralizada.

Tanto la CPU como los diferentes módulos de Siemens cuentan con conectores en la parte trasera para comunicación PROFINET isócrona entre los módulos. Estos conectores permiten una instalación y recambio de las cartas de forma rápida y sencilla.



Figura 106: Sistema de control principal

De izquierda a derecha, podemos distinguir:

- **Módulo de potencia PM 190W 120/230VAC:** Encargado de administrar la alimentación correcta a la CPU y al resto de los módulos.
- **CPU S7-1516F:** CPU encargada de ejecutar el programa, almacenamiento de datos y comunicación con el resto de los elementos.
- **Módulo de entradas digitales DI 32x24VDC HF:** módulo que recibe señales digitales de diferentes elementos de la estación.
- **Módulo de salidas digitales DQ 32xDC24V/0,5A HF:** modulo encargado de enviar las señales digitales procedentes de la CPU, a los diferentes actuadores de la estación.



Sistema de control descentralizado: ET200-SP + Módulos de Entrada/Salida (Figura 108).

En el primer apartado de este anexo hemos hablado del módulo de control de la periferia descentralizada, no obstante, este módulo va acompañado de una serie de cartas de entradas y salidas digitales.

Esta periferia descentralizada se encargará del control del tope de las cubiertas para ver si hay una cubierta disponible para iniciar el ciclo de trabajo y se encargará del panel de botones de la mesa del trabajador.

El elemento principal de este conjunto de módulos es el ET-200SP el cual se encarga de establecer la conexión con el PLC 1500 y controla las señales de entrada y salida de las cartas.

Al igual que en el apartado anterior, dónde mencionábamos que la conexión entre los módulos se realizaba mediante los conectores que tienen en la parte trasera, los módulos utilizados aquí también cuentan con dicha tecnología; por lo que automáticamente se establece una conexión PROFINET isócrona entre ellos.

Una diferencia respecto a los módulos del S7-1500 es que cada uno de los módulos cuenta con dos partes. Las cartas, que es cómo se denomina al elemento principal del módulo que es el que se encarga de administrar las señales que llegan de los borneros y es fácilmente sustituible por otra carta del mismo tipo en caso de avería; y el soporte, el cual es el elemento que contiene los borneros dónde se encuentran el puerto de entradas o salidas del módulo, es este elemento el que tiene los puertos para conexión entre módulos y es dónde se anclan las cartas gracias a un puerto específico y unos topes para asegurar la sujeción de la carta.

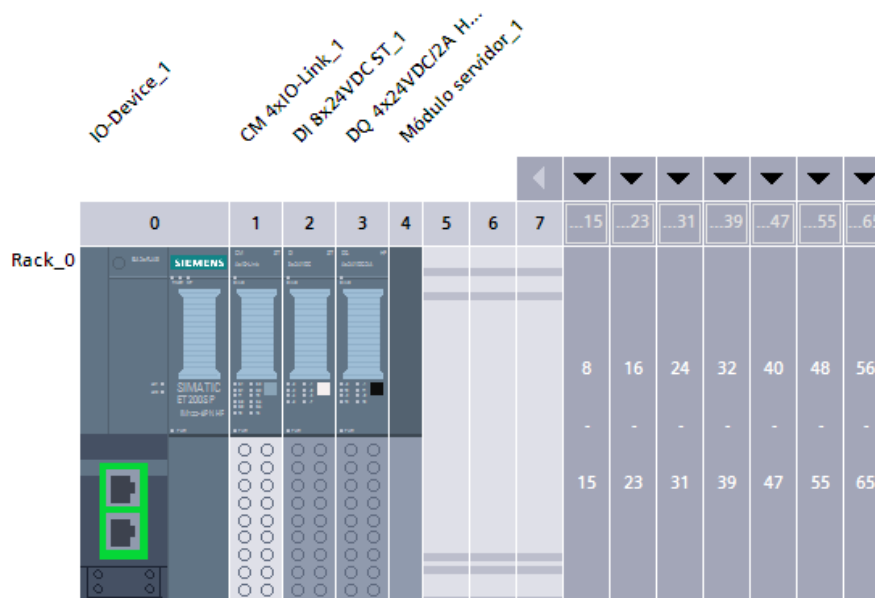


Figura 107: Periferia descentralizada



De izquierda a derecha podemos distinguir:

- **ET-200SP:** Módulo principal, equivalente al S7-1500 en el otro conjunto. Se encarga de establecer conexión con la CPU mediante PROFINET y administrar las señales del resto de módulos
- **Módulo de conexión CM 4XIO:** Control de la conexión punto a punto entre el maestro y otros dispositivos. En esta estación no es obligatorio, no obstante, esta planeado su uso en el futuro.
- **Módulo de entradas digitales DI 8x24VDC ST:** Módulo de entrada de hasta 8 señales de entrada digitales, puede trabajar con conexión de uno, dos o hasta tres hilos.
- **Módulo de salidas digitales DQ 4x24 VD/2A HT:** Módulo de salida de hasta 4 señales de salida digitales, puede trabajar con conexión a uno, dos y tres hilos.

Controlador del robot colaborativo. (Figura 110)

La amplia mayoría de los robots industriales en la actualidad necesitan un controlador. Este controlador es el encargado de gestionar el movimiento del robot, así como la administración de los puertos E/S y del almacenamiento e iniciación de las rutinas del robot, dejando dentro del brazo robótico la electrónica y sensores necesarios para su movimiento y correcto funcionamiento de la herramienta a utilizar.

En general, todos los controladores cuentan con un elemento conocido como TeachPendant. (Figura 109) Esta TeachPendant es un componente que normalmente está formado por una pantalla gráfica y una serie de botones y joysticks. Con esta TeachPendant podemos acceder a diferentes funciones del controlador como puede ser controlar el robot manualmente para moverlo a un punto específico, guardar o cargar una salvaguarda, editar el programa y los parámetros de seguridad, entre otros.

En el caso de Universal Robot esta TeachPendant es una pantalla táctil semejante a una Tablet que cuenta con una interfaz (conocida como PolyScope) basada en ventanas que permite un manejo intuitivo de las posibilidades que nos ofrece. Además, cuenta con una seta de emergencia que conecta directamente con el controlador, el botón de encendido del robot, una palanca del hombre muerto en la parte trasera y dependiendo de la versión de la TeachPendant, un puerto USB en un lateral para el uso de 'magic files' o carga/descarga de salvaguardas.

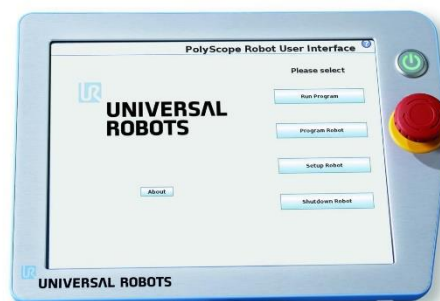


Figura 108 : TeachPendant de Universal Robors [13]

En la figura 110 podemos ver el interior del controlador. Como podemos ver, es un controlador de pequeñas dimensiones que cuenta con un asa para transportarlo y una puerta para proteger la electrónica.

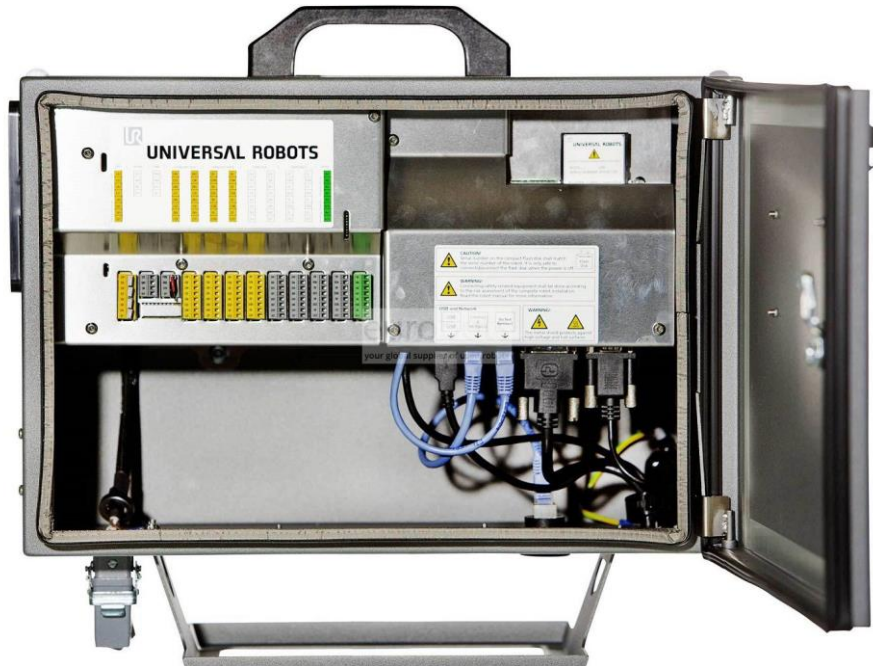


Figura 109: Armario de controladora del UR10 [38]

En la parte izquierda podemos encontrar los puertos E/S del controlador. Estos puertos se pueden utilizar para el control de una amplia variedad de equipo que incluye relés neumáticos, PLC y botones de parada de emergencia. Estos puertos cuentan con un código de colores que hace referencia a su funcionalidad:

- **Amarillos con texto rojo:** Señales de seguridad asignadas
- **Amarillo con texto negro:** Configurable para seguridad
- **Gris con texto negro:** E/S digital de uso general
- **Verde con texto negro:** E/S analógicas de uso general

En la figura 111 se muestra el diagrama de este grupo de puertos E/S del controlador.

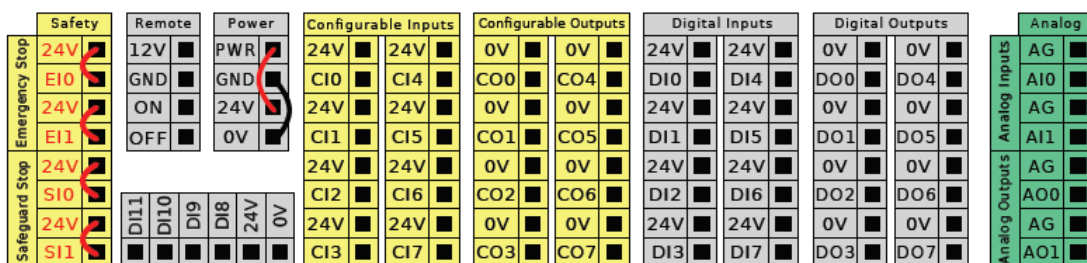


Figura 110: Diagrama de los puertos E/S del controlador [15]

En el manual de usuario de Universal Robot se puede encontrar más información sobre los puertos E/S del controlador, así como información de cableado de estos [15].



En la parte derecha encontramos los puertos de conexión para los diferentes elementos. Ahí podemos conectar la TeachPendant, los 2 conectores VGA del robot, la alimentación del controlador y, dependiendo de la versión, encontraremos o una ranura SD o un puerto USB para introducir la tarjeta de identificación del robot. Como indicamos en otros capítulos este controlador es universal para todos los robots colaborativos de Universal Robot, pero los robots tienen unos valores de calibración únicos. Por lo que cada robot tiene una tarjeta de identificación única que debemos introducir en el controlador con el que vaya a trabajar dicho robot.

Para este proyecto debemos tener un mayor enfoque en la comunicación con el PLC. Este controlador cuenta con 10 módulos opcionales que se encargan de recoger diferentes valores referentes a las condiciones de trabajo del robot, las cuales vamos a poder transmitir a otros dispositivos mediante PROFINET. Entre estas variables podemos destacar los registros de entrada y los registros de salida que pueden recibir diferentes tipos de valores como pueden ser valores binarios, enteros o coma flotante. Para más información de la estructura de datos podemos consultar el anexo II: Comunicación PROFINET del controlador.





ANEXO V: CÓDIGO DE POLYSCOPE: RUTINA DEL ROBOT

En este anexo se recoge el programa completo realizado en la máquina virtual que contiene la rutina que realizará el robot en la estación (Figura 112). También se adjuntará los parámetros de los puntos seleccionados. (Figuras 113-118)

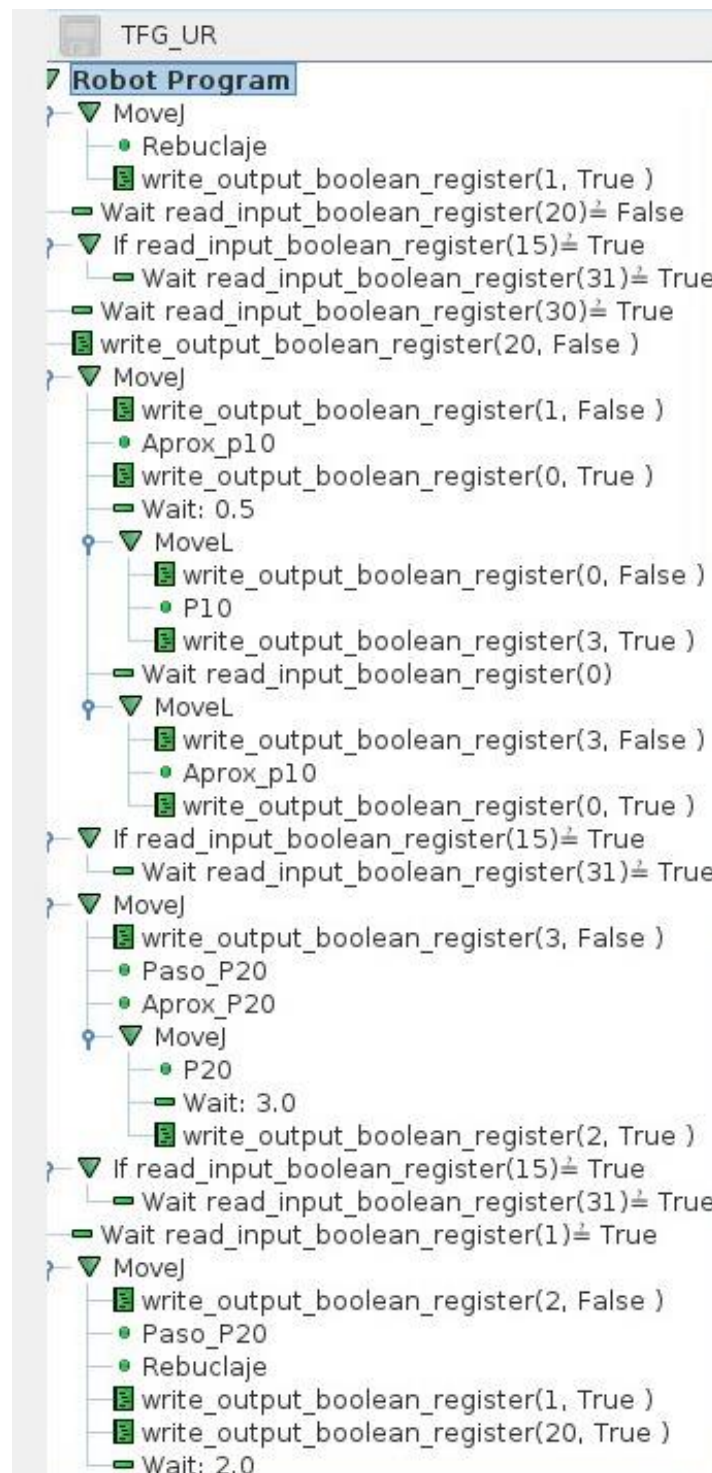


Figura 111: Programa de Polyscope



Punto de rebuclaje:

DHEF_20

X	<input type="text" value="-150.96"/>	mm	<input type="button" value="+"/>	<input type="button" value="-"/>
Y	<input type="text" value="-596.57"/>	mm	<input type="button" value="+"/>	<input type="button" value="-"/>
Z	<input type="text" value="-106.20"/>	mm	<input type="button" value="+"/>	<input type="button" value="-"/>

Rotation Vector [rad]

RX	<input type="text" value="0.070"/>	rad	<input type="button" value="+"/>	<input type="button" value="-"/>
RY	<input type="text" value="-3.082"/>	rad	<input type="button" value="+"/>	<input type="button" value="-"/>
RZ	<input type="text" value="0.695"/>	rad	<input type="button" value="+"/>	<input type="button" value="-"/>

Joint Positions

Base	<input type="text" value="-89.49"/>	°	<input type="button" value="+"/>	<input type="button" value="-"/>
Shoulder	<input type="text" value="-72.10"/>	°	<input type="button" value="+"/>	<input type="button" value="-"/>
Elbow	<input type="text" value="-126.22"/>	°	<input type="button" value="+"/>	<input type="button" value="-"/>
Wrist 1	<input type="text" value="-46.29"/>	°	<input type="button" value="+"/>	<input type="button" value="-"/>
Wrist 2	<input type="text" value="91.39"/>	°	<input type="button" value="+"/>	<input type="button" value="-"/>
Wrist 3	<input type="text" value="-1.78"/>	°	<input type="button" value="+"/>	<input type="button" value="-"/>

Figura 112: Configuración punto de rebuclaje



Punto Aprox_P10:

Tool Position				
X	<input type="text" value="-147.93"/>	mm	<input type="button" value="+"/>	<input type="button" value="-"/>
Y	<input type="text" value="-937.59"/>	mm	<input type="button" value="+"/>	<input type="button" value="-"/>
Z	<input type="text" value="-175.00"/>	mm	<input type="button" value="+"/>	<input type="button" value="-"/>
<input type="text" value="Rotation Vector [rad]"/> <input type="button" value="▼"/>				
RX	<input type="text" value="0.064"/>	rad	<input type="button" value="+"/>	<input type="button" value="-"/>
RY	<input type="text" value="-3.164"/>	rad	<input type="button" value="+"/>	<input type="button" value="-"/>
RZ	<input type="text" value="0.084"/>	rad	<input type="button" value="+"/>	<input type="button" value="-"/>
Joint Positions				
Base	<input type="text" value="-89.49"/>	°	<input type="button" value="+"/>	<input type="button" value="-"/>
Shoulder	<input type="text" value="-114.29"/>	°	<input type="button" value="+"/>	<input type="button" value="-"/>
Elbow	<input type="text" value="-80.96"/>	°	<input type="button" value="+"/>	<input type="button" value="-"/>
Wrist 1	<input type="text" value="-71.70"/>	°	<input type="button" value="+"/>	<input type="button" value="-"/>
Wrist 2	<input type="text" value="91.39"/>	°	<input type="button" value="+"/>	<input type="button" value="-"/>
Wrist 3	<input type="text" value="-1.78"/>	°	<input type="button" value="+"/>	<input type="button" value="-"/>

Figura 113: Configuración punto de aproximación a P10



Punto P10:

DHEF_20

X	<input type="text" value="-147.93"/>	mm	<input type="button" value="+"/>	<input type="button" value="-"/>
Y	<input type="text" value="-937.59"/>	mm	<input type="button" value="+"/>	<input type="button" value="-"/>
Z	<input type="text" value="-342.35"/>	mm	<input type="button" value="+"/>	<input type="button" value="-"/>

Rotation Vector [rad]

RX	<input type="text" value="0.064"/>	rad	<input type="button" value="+"/>	<input type="button" value="-"/>
RY	<input type="text" value="-3.164"/>	rad	<input type="button" value="+"/>	<input type="button" value="-"/>
RZ	<input type="text" value="0.084"/>	rad	<input type="button" value="+"/>	<input type="button" value="-"/>

Joint Positions

Base	<input type="text" value="-89.49"/>	°	<input type="button" value="+"/>	<input type="button" value="-"/>
Shoulder	<input type="text" value="-120.37"/>	°	<input type="button" value="+"/>	<input type="button" value="-"/>
Elbow	<input type="text" value="-89.86"/>	°	<input type="button" value="+"/>	<input type="button" value="-"/>
Wrist 1	<input type="text" value="-56.73"/>	°	<input type="button" value="+"/>	<input type="button" value="-"/>
Wrist 2	<input type="text" value="91.39"/>	°	<input type="button" value="+"/>	<input type="button" value="-"/>
Wrist 3	<input type="text" value="-1.78"/>	°	<input type="button" value="+"/>	<input type="button" value="-"/>

Figura 114: Configuración punto P10



Punto Paso _P20:

DHEF_20

X	<input type="text" value="-606.80"/>	mm	<input type="button" value="+"/>	<input type="button" value="-"/>
Y	<input type="text" value="-113.99"/>	mm	<input type="button" value="+"/>	<input type="button" value="-"/>
Z	<input type="text" value="151.78"/>	mm	<input type="button" value="+"/>	<input type="button" value="-"/>

Rotation Vector [rad]

RX	<input type="text" value="1.680"/>	rad	<input type="button" value="+"/>	<input type="button" value="-"/>
RY	<input type="text" value="2.753"/>	rad	<input type="button" value="+"/>	<input type="button" value="-"/>
RZ	<input type="text" value="-0.316"/>	rad	<input type="button" value="+"/>	<input type="button" value="-"/>

Joint Positions

Base	<input type="text" value="-154.70"/>	°	<input type="button" value="+"/>	<input type="button" value="-"/>
Shoulder	<input type="text" value="-76.32"/>	°	<input type="button" value="+"/>	<input type="button" value="-"/>
Elbow	<input type="text" value="-92.23"/>	°	<input type="button" value="+"/>	<input type="button" value="-"/>
Wrist 1	<input type="text" value="-89.00"/>	°	<input type="button" value="+"/>	<input type="button" value="-"/>
Wrist 2	<input type="text" value="91.39"/>	°	<input type="button" value="+"/>	<input type="button" value="-"/>
Wrist 3	<input type="text" value="-1.78"/>	°	<input type="button" value="+"/>	<input type="button" value="-"/>

Figura 115: Configuración punto de paso P10-P20



Punto Aprox_P20:

DHEF_20

X	<input type="text" value="-59.10"/>	mm	<input type="button" value="+"/>	<input type="button" value="-"/>
Y	<input type="text" value="931.77"/>	mm	<input type="button" value="+"/>	<input type="button" value="-"/>
Z	<input type="text" value="21.32"/>	mm	<input type="button" value="+"/>	<input type="button" value="-"/>

Rotation Vector [rad]

RX	<input type="text" value="3.418"/>	rad	<input type="button" value="+"/>	<input type="button" value="-"/>
RY	<input type="text" value="0.458"/>	rad	<input type="button" value="+"/>	<input type="button" value="-"/>
RZ	<input type="text" value="-0.096"/>	rad	<input type="button" value="+"/>	<input type="button" value="-"/>

Joint Positions

Base	<input type="text" value="-256.74"/>	°	<input type="button" value="+"/>	<input type="button" value="-"/>
Shoulder	<input type="text" value="-103.42"/>	°	<input type="button" value="+"/>	<input type="button" value="-"/>
Elbow	<input type="text" value="-80.19"/>	°	<input type="button" value="+"/>	<input type="button" value="-"/>
Wrist 1	<input type="text" value="-68.49"/>	°	<input type="button" value="+"/>	<input type="button" value="-"/>
Wrist 2	<input type="text" value="91.39"/>	°	<input type="button" value="+"/>	<input type="button" value="-"/>
Wrist 3	<input type="text" value="-1.78"/>	°	<input type="button" value="+"/>	<input type="button" value="-"/>

Figura 116: Configuración punto de aproximación a P20



Punto P20:

DHEF_20

X	<input type="text" value="182.60"/>	mm	<input type="button" value="+"/>	<input type="button" value="-"/>
Y	<input type="text" value="1073.93"/>	mm	<input type="button" value="+"/>	<input type="button" value="-"/>
Z	<input type="text" value="-28.10"/>	mm	<input type="button" value="+"/>	<input type="button" value="-"/>

Rotation Vector [rad]

RX	<input type="text" value="3.467"/>	rad	<input type="button" value="+"/>	<input type="button" value="-"/>
RY	<input type="text" value="0.018"/>	rad	<input type="button" value="+"/>	<input type="button" value="-"/>
RZ	<input type="text" value="-0.026"/>	rad	<input type="button" value="+"/>	<input type="button" value="-"/>

Joint Positions

Base	<input type="text" value="-271.40"/>	°	<input type="button" value="+"/>	<input type="button" value="-"/>
Shoulder	<input type="text" value="-118.91"/>	°	<input type="button" value="+"/>	<input type="button" value="-"/>
Elbow	<input type="text" value="-63.96"/>	°	<input type="button" value="+"/>	<input type="button" value="-"/>
Wrist 1	<input type="text" value="-68.49"/>	°	<input type="button" value="+"/>	<input type="button" value="-"/>
Wrist 2	<input type="text" value="91.39"/>	°	<input type="button" value="+"/>	<input type="button" value="-"/>
Wrist 3	<input type="text" value="-1.78"/>	°	<input type="button" value="+"/>	<input type="button" value="-"/>

Figura 117: Configuración punto P20





ANEXO VI: PANEL DE CONTROL DEL USUARIO

En este anexo vamos a explicar el panel de control del usuario que se sitúa dentro de la estación y que usará el usuario para guiar el ciclo de trabajo de la estación.

Con este panel de control, el trabajador podrá administrar las funciones básicas de la estación desde el área de trabajo del operario. Los elementos principales del panel es una seta de emergencia y 3 botones con los que podremos administrar el ciclo de trabajo. Más adelante se explicará con detalle el panel de control.

Las dimensiones de este panel están pensadas para que una persona de altura media pueda acceder a todos los botones de forma sencilla sin tener que salir de la zona de trabajo.

En las figuras 119 y 120 se pueden ver las dimensiones del panel de control.



Figura 118: Pupitre de control, vista 1



Figura 119: Pupitre de control, vista 2

En cuanto a la interfaz del panel de control, se puede ver en la parte superior una placa de identificación con el número de estación correspondiente. Justo debajo de esta placa, encontramos la seta de emergencia diseñada para que el operario pueda pulsarla de forma inmediata si detecta algún fallo en la estación.

En la parte inferior encontramos 3 botones con colores determinados, cada uno de ellos con una etiqueta en la parte inferior indicando su función:

- **Botón Azul oscuro:** su función es la programación de una parada segura al final del ciclo del robot. Esta parada segura está destinada a poder realizar una pausa sin necesidad de realizar un rearme para volver a empezar la rutina, como puede darse en los casos de descanso, revisión o cambio de turno. La etiqueta indica “Parada”
- **Botón Verde claro:** su función es reanudar el ciclo de trabajo una vez se ha realizado una parada de seguridad. No se necesita un rearme en caso de que se haya solicitado este tipo de parada y solo hará efecto cuando la estación se encuentre parada. La etiqueta indica “Reanudación”.



- **Botón blanco:** este botón, a diferencia de los dos anteriores, cuenta con un LED que parpadea cuando es necesario que sea pulsado. Su función es indicar a la estación que la cubierta ha sido instalada por el operario y ya se puede llevar el coche procesado a la siguiente estación. Su etiqueta indica “Fin de trabajo”.

En la figura 121 podemos observar la interfaz de este panel de control del usuario.



Figura 120: Pupitre de control, vista de panel de botonera



ANEXO VII: MANUAL DEL USUARIO

Introducción

En este anexo se describirán los procedimientos necesarios para poner en funcionamiento la estación una vez esta haya sido montada. El objetivo de este apartado es que el operario sea capaz de iniciar la primera puesta en funcionamiento de la estación y el rearme-reanudación en caso de parada.

Primera puesta en marcha

Partiendo de la estación ya montada, debemos comprobar en primer lugar que los elementos de seguridad están operativos y no activados (setas de emergencia levantadas y sensor de presencia sin detectar ninguna intrusión).

Una vez se han comprobado estos aspectos, debemos cargar el programa del PLC. Para ello, con el autómatas encendido, debemos conectarnos con un ordenador al autómatas vía PROFINET, utilizando el otro puerto Ethernet que tenemos libre, el correspondiente a la red de control. Para cargar el programa debemos seleccionar el icono de “Cargar en dispositivo” y nos aparecerá un menú (Figura 85) para seleccionar el autómatas dónde cargaremos el programa.

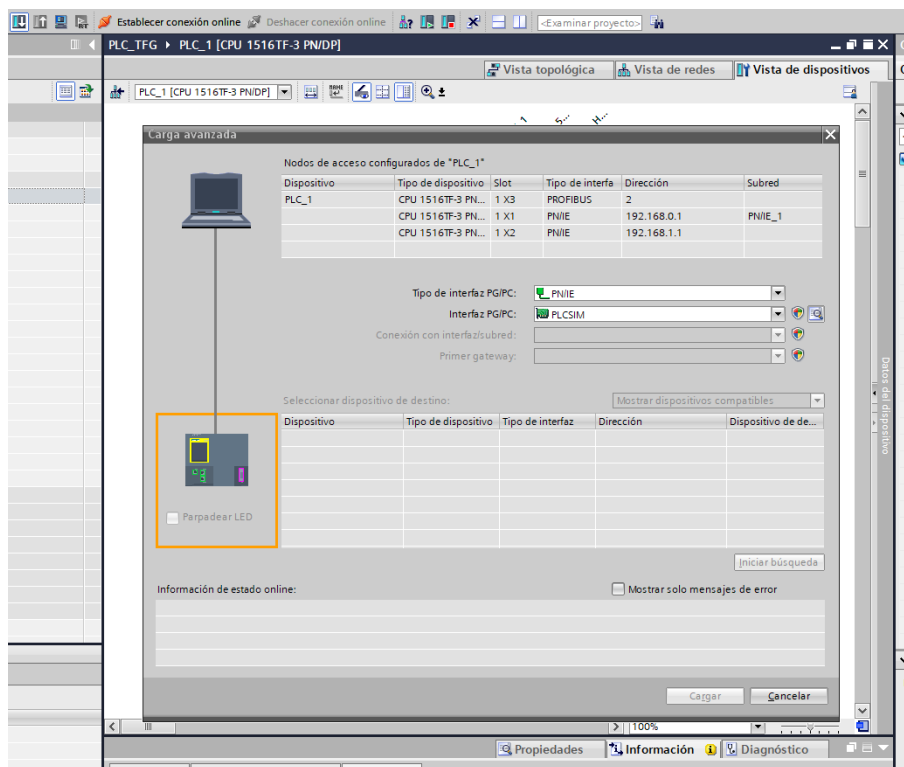


Figura 121: Cargar programa en dispositivo TIA Portal

Con el programa ya cargado en el autómatas, debemos cargar también el programa del robot (en caso de que no se haya realizado con la TeachPendant propia). Con el controlador apagado, debemos introducir el USB en la parte trasera de la TeachPendant (o en el armario de control, dependiendo de la versión del controlador) y encender el controlador.



Cuando aparezca el menú principal (Figura 77) debemos seleccionar la opción “Program Robot”. Dentro de ese menú nos aparecerá la opción de importar un programa de la unidad USB, seleccionamos esa opción.

Una vez sendos programas, el del autómatas y el del controlador están activos, debemos comprobar que esté correctamente configurada la red PROFINET en el robot, tal y como se ha descrito en el apartado 2.7.- “Programación y simulación del programa de control del PLC”.

Con estos procedimientos la estación está a punto y lista para iniciar su funcionamiento. Por defecto, al iniciarse todo por primera vez, tal y como estamos describiendo en este apartado, la estación partirá del estado de “Pausa de inicio”. Este estado de pausa es similar al que nos encontramos una vez se ha pulsado una seta de emergencia y se ha liberado el botón. En este estado, la columna de señalización no indica que se haya producido un error, no obstante, es necesario rearmar la estación.

Para poner en marcha la estación, será necesario pulsar primero el botón de rearme y después, el botón de reanudación.

Una vez pulsados sendos botones, la columna de señalización encenderá la luz verde indicando el funcionamiento de la estación en estado “Normal” y se empezarán a mover ambas mantenciones llevando los coches y las piezas hasta sus respectivos topes.

Puesta en marcha en caso de parada de error/emergencia

En caso de que se produzca una parada de emergencia o una parada de error el procedimiento de puesta en marcha sería el mismo.

Lo primero que debemos hacer ante esta parada sería resolver el conflicto que haya producido la parada. Por ejemplo, en caso de intrusión en la zona de trabajo del robot, se debe desplazar dicho elemento fuera de la zona de trabajo del robot.

Una vez se ha resuelto el conflicto, debemos confirmar al sistema de control de ello. Para enviar esa señal a la estación debemos pulsar el botón de rearme situado en el armario eléctrico. Este rearme se hace efectivo únicamente en caso de que el conflicto se haya solucionado, si se pulsa dicho botón mientras exista el conflicto no se producirá el rearme.

En el estado de rearme, la columna de señalización queda totalmente apagada (salvo el caso de que haya saturación) y la estación se queda en espera de que el operario pulse el botón de reanudar. En caso de que se produzca otro conflicto mientras la estación se encuentra en rearme, esta volverá al estado de error y será necesario repetir los procedimientos descritos con anterioridad.

Para finalizar el estado de error/emergencia y poner a la estación en marcha debemos pulsar el botón de reanudación del armario eléctrico cuando la estación se encuentra en rearme.

En el apartado 1.3.- “Funcionamiento en automático” podemos observar el diagrama de flujo “Sistema de control de seguridad” que representa de forma gráfica lo descrito en este subapartado.



Puesta en marcha tras parada programada.

La estación cuenta también con un tipo de parada diferente a los descritos en el anterior apartado. Esta parada supone una pausa programada por el operario. Al ser una parada programada no es necesario un rearme para volver a poner a la estación en marcha. Tanto el botón de pausa como el de reanudación se encuentran en la mesa de control del operario.

Esta parada programada está destinada para los casos de cambio de turno o para los turnos de descanso.

Como hemos indicado anteriormente, para solicitar esta parada de seguridad debemos pulsar el botón de pausa (botón de color azul) del panel de control. Al finalizar el ciclo de trabajo del robot se hará efectiva esta parada y todo el sistema entra en pausa.

Para reanudar el trabajo, no es necesario realizar ningún tipo de rearme, sino que con pulsar el botón de reanudación (botón de color verde) situado en el panel del operario es suficiente y la estación volverá a trabajar como lo hace de manera normal.