



Universidad de Valladolid



ESCUELA DE INGENIERÍAS
INDUSTRIALES

UNIVERSIDAD DE VALLADOLID

ESCUELA DE INGENIERIAS INDUSTRIALES

Grado en Ingeniería en Electrónica Industrial y Automática

Automatización de pintura de piezas complejas utilizando una estación de pintura y un robot colaborativo

Autor:

Merino Fidalgo, Sergio

Tutor:

Candau Pérez, José
Departamento de Ingeniería de
Sistemas y Automática

Valladolid, julio de 2021.

Agradecimientos

A mi tutor, Pepe, por ayudarme y estar siempre pendiente cuando he necesitado ayuda en la realización de este trabajo.

A mis compañeros y amigos que me han acompañado durante mi camino en esta aventura universitaria.

Y especialmente a mi familia.

A mis padres por estar ahí y apoyarme de forma incondicional.

Y a mi hermano, que siempre consigue alegrarme el día con una de nuestras tonterías.

Gracias.





Resumen

Actualmente cada vez hay más empresas pequeñas y medianas que apuestan por la automatización de procesos que hasta ahora se estaban realizando de forma manual o utilizando métodos arcaicos, resultando en una producción reducida y poco eficiente y a veces una labor tediosa para los trabajadores involucrados.

En este proyecto nos encontramos en un caso similar, donde una empresa familiar se ha abierto a la innovación y a las nuevas tecnologías para seguir evolucionando y así adaptarse a los requisitos del mercado. Para ello, se pretende automatizar un proceso de pintura utilizando un robot colaborativo, facilitando así esta tarea y logrando una mayor eficacia.

El principal objetivo de este proyecto es la automatización de un proceso de pintura de piezas de silicona de geometría compleja destinadas a trabajar en entornos de alta tensión. Para conseguirlo, se utilizará una instalación ya en funcionamiento para pintar piezas cilíndricas. Además, se debe encontrar una solución que involucre en mayor medida el brazo robótico para pintar de forma eficaz las piezas, teniendo muy en cuenta la homogeneidad del acabado, puesto que de ello depende su rendimiento eléctrico.

Palabras clave:

Robot, PLC, PolyScope, TIA Portal, pintura.



Abstract

Nowadays, there are more and more small and medium-sized companies that are opting for the automation of processes that until now have been carried out manually or using archaic methods, resulting in reduced and inefficient production and sometimes tedious work for the workers involved.

In this project, we found ourselves in a similar case, where a family business has opened up to innovation and new technologies in order to continue evolving and thus adapt to market demands. Therefore, it is intended to automate a painting process using a collaborative robot, thus facilitating this task and achieving greater efficiency.

The main objective of this project is the automation of a painting process for complex geometry silicone parts intended to work in high voltage environments. To achieve this purpose, an installation already in operation for painting cylindrical parts will be used. Moreover, a solution must be found that involves the robotic arm to a greater extent in order to paint the parts efficiently, while being extremely careful about the homogeneity of the finish, since this is critical for its electrical performance.

Keywords:

Robot, PLC, PolyScope, TIA Portal, painting.





ÍNDICE

Agradecimientos	1
Resumen	3
Abstract	5
1. Introducción	13
1.1. Presentación	13
1.2. Contexto industrial	13
1.3. Descripción del proyecto	15
2. Retos y objetivos del proyecto	19
3. Descripción de la instalación	21
3.1. Introducción	21
3.2. Contenedores de transporte	25
3.3. Puestos de entrada y salida de contenedores	26
3.4. El robot	27
3.5. Armario de control	29
3.6. Sistema de pintura	39
4. El proceso de pintura de piezas cilíndricas	47
4.1. Introducción	47
4.2. Análisis del proceso	50
5. UR10e y PolyScope	55
5.1. Descripción del robot	55
5.2. El robot en la instalación	60
5.3. Interfaz gráfica	63
6. PLC S7 y TIA Portal	75
6.1. ¿Qué es un PLC?	75
6.2. PLC S7 1214C DC/DC/DC	78



6.3.	El PLC en la instalación	80
6.4.	TIA Portal y el lenguaje de programación	83
7.	Desarrollo y soluciones del proyecto.....	91
7.1.	Inicio del proyecto en la empresa en el periodo de prácticas	91
7.2.	Primeras soluciones	92
7.3.	Obtención y análisis del software de la estación	96
7.4.	Desarrollo de nuevas soluciones y diseño de máscaras y carros	101
7.5.	Inicio del periodo de ampliación de prácticas.....	110
7.6.	Modificaciones y mejoras de las dos soluciones finales	113
7.7.	Pruebas con la funda del robot.....	123
8.	Resultados	129
9.	Líneas futuras de trabajo	131
10.	Conclusiones.....	133
11.	Bibliografía.....	135



ÍNDICE DE FIGURAS

Figura 1. Logo de Vulcanizados Alvarez	13
Figura 2. Representación virtual de las instalaciones	14
Figura 3. Estructura interna de una de las piezas a pintar.....	16
Figura 4. Pieza con forma de 'T' pintada.....	16
Figura 5. Pieza cilíndrica pintada.	17
Figura 6. Instalación de la estación de pintura.	21
Figura 7. Planos de la instalación de pintura.	22
Figura 8. Puerta de acceso y detalle del sensor.....	23
Figura 9. Sensor detector de piezas.....	24
Figura 10. Zona de control de la instalación.	24
Figura 11. Contenedor de transporte.	25
Figura 12. Puestos de entrada y salida de contenedores.	26
Figura 13. Sensores e imán en el puesto de entrada y salida.....	27
Figura 14. Representación en la pantalla HMI de imanes y sensores.	27
Figura 15. Robot UR10e.	28
Figura 16. Ubicación del robot en la instalación.....	29
Figura 17. Armario de control.....	30
Figura 18. PLC del armario de control.....	30
Figura 19. Pantalla HMI.....	31
Figura 20. Pestaña "Inicio" de la pantalla HMI.	32
Figura 21. Pestaña "Alarmas" de la pantalla HMI.....	32
Figura 22. Pestaña "Manual" de la pantalla HMI.....	33
Figura 23. Pestaña "Parámetros" de la pantalla HMI.	34
Figura 24. Pestaña "Robot" de la pantalla HMI.	35
Figura 25. Pestaña "Configuración Zona 1" de la pantalla HMI.....	36
Figura 26. Pestaña "Configuración Zona 2" de la pantalla HMI.....	36
Figura 27. Pestaña "Cotas" de la pantalla HMI.	37
Figura 28. Pestaña "Pintura" de la pantalla HMI.	38
Figura 29. Pestaña "Recetas" de la pantalla HMI.....	38
Figura 30. Cabina de pintura.....	40
Figura 31. Soporte rotativo de la cabina de pintura.	40

Figura 32. Reciprocador.	41
Figura 33. Pistolas de pintura.....	41
Figura 34. Sistema de extracción.	43
Figura 35. Sistema de suministro de pintura.	44
Figura 36. Panel de control de presión.	45
Figura 37. Sistema generador de presión.	45
Figura 38. Tipos de piezas cilíndricas.	47
Figura 39. Referencias para las piezas cilíndricas.	48
Figura 40. Pinzas y soportes para las piezas cilíndricas.	48
Figura 41. Detalle de la pinza.	49
Figura 42. Máscaras de las piezas cilíndricas.	50
Figura 43. Robot en posición inicial.	51
Figura 44. Robot cogiendo pieza de Zona 2.	51
Figura 45. Posición detección de pieza + detalle del sensor.....	52
Figura 46. Robot dejando la pieza en la cabina.	53
Figura 47. Robot UR10e.	55
Figura 48. Alcance del robot.	56
Figura 49. Relación entre carga en Kg y la compensación del centro de gravedad en mm.	57
Figura 50. Robot y su armario de control en la instalación.	60
Figura 51. Controlador del robot colaborativo.	61
Figura 52. Placa intermedia de la garra.	62
Figura 53. Garras y pinzas según la referencia.	62
Figura 54. Interfaz de usuario.	63
Figura 55. Pantalla inicial de "Programa".	67
Figura 56. Detalle de la barra de herramientas.	67
Figura 57. Detalle de las pestañas.	68
Figura 58. Comandos avanzados.....	69
Figura 59. Plantillas.	71
Figura 60. Control por valor numérico.....	73
Figura 61. Control por flechas.....	73
Figura 62. Control manual.	74
Figura 63. Constitución física en un PLC modular.....	75

Figura 64. Arquitectura funcional de un PLC.	75
Figura 65. Esquema del ciclo de trabajo.	77
Figura 66. CPU 1214C.....	78
Figura 67. Módulo de entradas y salidas digitales SM 1223 DC/DC.	79
Figura 68. El armario de control en la instalación.	80
Figura 69. Armario de control.....	81
Figura 70. Distribución de la interfaz HMI.	81
Figura 71. Interior del armario de control.	82
Figura 72. Pantalla inicial de TIA Portal.....	83
Figura 73. Pestaña para añadir elementos al proyecto.	84
Figura 74. Captura de TIA Portal.	84
Figura 75. Ejemplo de diagrama de contactos.....	85
Figura 76. Instrucciones de programación en TIA Portal.....	87
Figura 77. Ejemplo programación en SCL.	88
Figura 78. Tipos de bloques de programa.	89
Figura 79. Variables almacenadas en un bloque de datos.	90
Figura 80. Robot UR10e en la estación de pintura.	91
Figura 81. Captura del programa URSim.....	92
Figura 82. Carro con soporte de cambio de agarre y máscara y soporte modificados. ...	93
Figura 83. Las juntas, la base y la brida de herramienta del brazo robótico.	94
Figura 84. Pieza en el soporte rotativo.	95
Figura 85. Uniones de la ‘T’.....	95
Figura 86. Captura del programa del PLC.	99
Figura 87. Salidas para controlar la ejecución del robot.	99
Figura 88. Opción de control de la pinza en el menú “Manual”.....	100
Figura 89. Inclinação del robot para rotar la muñeca 3.....	101
Figura 90. Numeración de los orificios de la pieza.	103
Figura 91. Movimientos circulares con varios puntos.	104
Figura 92. Bloques de programa de Festo.	105
Figura 93. Pinza para la Referencia 1.	106
Figura 94. Chapa de la cabina frente al robot.....	107
Figura 95. Pantalla HMI del armario de control.....	109
Figura 96. Control de pistolas en modo manual de la pantalla HMI.	111

Figura 97. Software CODESYS para el control de las pistolas. 112

Figura 98. Configuración del TCP original. 113

Figura 99. Configuración del nuevo TCP_1. 114

Figura 100. Diferencia de TCP en la rotación en torno a un punto. 114

Figura 101. Problemas con el tubo de alimentación de presión de la pinza. 115

Figura 102. Valores configurables de cada receta. 117

Figura 103. Ejemplo de impresión 3D por parte de la entidad. 119

Figura 104. Diferencia entre el punto inicial del movimiento y el punto de cambio de posición intermedio. 120

Figura 105. Detalle de la instrucción ‘speedl’. 121

Figura 106. Detalle de escritura en registro desde el robot. 122

Figura 107. Imagen de un UR10e con funda protectora. 124



1. Introducción

1.1. Presentación

Este proyecto comenzó a forjarse durante la búsqueda de una entidad industrial para la realización de las prácticas en empresa acorde con mi titulación e intereses. Con la ayuda y recomendación de un amigo, contacté con una empresa, Vulcanizados Alvarez S.L., donde me recibieron con los brazos abiertos, me enseñaron su actividad y la fábrica y me propusieron distintos proyectos. Inicialmente seleccioné uno de los ellos para desarrollarlo durante el periodo de prácticas y otro para el TFG, pero finalmente me decanté por realizar un único proyecto para ambos.

1.2. Contexto industrial

Fundada en 1974, VULCANIZADOS INDUSTRIALES ALVAREZ S.L. tiene una larga experiencia en el proceso de transformación de toda la gama de polímeros elastoméricos [1].



Figura 1. Logo de Vulcanizados Alvarez

Anteriormente ubicada en Santovenia de Pisuegra, esta empresa nació como un negocio familiar, enfocada en la producción de **piezas de caucho** mediante un método mucho más artesanal. Con el paso del tiempo y el avance de la tecnología tanto el proceso de fabricación como las piezas producidas fueron evolucionando, ampliando así sus límites de mercado. Recientemente realizaron un cambio de sede, llevándose la fábrica al Parque Tecnológico de Boecillo, donde cuentan con un edificio de oficinas que comparten con otra empresa y una gran nave industrial donde realizan todos los procesos de fabricación, el control de calidad o el almacenamiento, entre otros.



Figura 2. Representación virtual de las instalaciones.

Su última apuesta ha sido el mercado de la alta tensión, donde suministran empalmes, juntas de unión y tomas de corriente para toda Europa. Además, tienen acuerdos comerciales para suministrar distintos tipos de productos a grandes multinacionales como SIEMENS o Schneider.

Debido a esto, han creado un laboratorio eléctrico de altas prestaciones donde pueden testear cada una de las piezas que producen, sometiéndolas a todo tipo de pruebas para garantizar que cumplen el riguroso estándar que reclaman sus clientes.

Con el fin de seguir evolucionando, se están adentrando rápidamente al mundo de la robótica industrial, donde ya cuentan con un número razonable de sistemas

robotizados para la producción de unas piezas concretas o de un robot colaborativo para la pintura de estas

Finalmente, la empresa produce desde caucho natural (NR), cauchos sintéticos comunes como cauchos de estireno (SBR), cauchos de etileno-propileno (EPDM), cauchos de nitrilo (NBR), hasta cauchos para aplicaciones específicas como cauchos de policloropreno (CR), cauchos butílicos (IIR), Cauchos de Poliuretano (PUR) Cauchos resistentes al aceite (ACM, ECO), hasta cauchos de silicona (MQ, PMQ, VMQ, PVMQ), adecuado a cada una de las necesidades del cliente [1].

Precisamente los requerimientos específicos y muy estrictos para una pieza determinada por parte de un cliente son el origen del proyecto del que el TFG forma parte, tal como se indica a continuación.

1.3. Descripción del proyecto

Este proyecto consiste en la implementación de una **estación de pintura automatizada** junto con un **robot colaborativo UR10e** para pintar piezas de **geometría compleja**.

Estas piezas con forma de 'T' y de 'X' están hechas de varios tipos de **silicona** especiales, cuya finalidad es ser utilizadas como conexiones, empalmes o tomas de corriente entre líneas de **alta tensión**. Debido a la composición y distribución de la silicona, se consiguen unas piezas que evitan las pérdidas de tensión entre conductores gracias a un compuesto semiconductor y proporcionan una o dos tomas de corriente para cualquier tipo de aplicación. Además, su estructura externa garantiza aislamiento y seguridad frente a cualquier tipo de descarga o pico de tensión.



Figura 3. Estructura interna de una de las piezas a pintar.

Hasta ahora estas piezas en Vulcanizados Alvarez están pendientes de comercialización, esperando a finalizar su fabricación con la capa de pintura exterior para comenzar con su distribución entre los distintos clientes internacionales.

Para pintar estas piezas se utiliza un compuesto especial, que confiere una serie de propiedades que completan las especificaciones de las piezas.



Figura 4. Pieza con forma de 'T' pintada.

Actualmente ya se está empleando la instalación mencionada anteriormente al inicio del apartado para pintar otro tipo de piezas con forma cilíndrica con un orificio pasante, cuya finalidad es similar a las piezas utilizadas en la realización de este proyecto. Estas piezas tiempo atrás se pintaban a mano, depositándolas en un elemento rotativo mientras un operario las pintaba con una pistola de pintura. Debido al aumento de la demanda por la constante expansión del mercado y de la empresa, se decidió realizar la instalación de la estación de pintura para aumentar la producción y mejorar las condiciones laborales de los empleados.



Figura 5. Pieza cilíndrica pintada.

Algo similar sucede con estas nuevas piezas, ya que tienen multitud de encargos por parte de grandes multinacionales y necesitan pintar de forma más rápida y efectiva garantizando un acabado excepcional en cada una de las piezas, por lo que pretenden emplear la nueva instalación para cumplir estos objetivos.

Aquí comienza el trabajo que se presenta como TFG: realizar esta integración para que se puedan pintar las piezas 'T' y 'X' utilizando la actual estación de pintura y teniendo en cuenta las dificultades que entraña.



2. Retos y objetivos del proyecto

El proyecto consiste en la automatización del proceso de pintura de piezas aislantes de alta tensión empleando un robot colaborativo UR10e.

En la empresa actualmente operan con este robot, con el que realizan un proceso similar en el que pintan piezas cilíndricas, donde utilizan un soporte regulado por un variador de frecuencia que hace que la pieza rote mientras las pistolas de la estación de pintura actúan sobre ella. En este caso, el robot se dedica a coger las piezas del carro, posicionarlas en la estación de pintura y posteriormente devolverlas al carro una vez pintadas. El control y la sincronización del proceso (el robot, las pistolas de la estación de pintura y el eje rotativo) se lleva a cabo mediante un **PLC** de control SIEMENS modelo S7-1200, que utiliza una conexión por **PROFINET** con el resto de los elementos.

Mi trabajo en este caso ha sido diseñar un nuevo proceso para pintar las piezas de geometría compleja (con forma de 'T' y de 'X'). La diferencia fundamental es la forma de pintarlas, puesto que no se puede utilizar la misma estrategia debido a la forma no cilíndrica. No hay que olvidar que un factor clave es garantizar la **homogeneidad** en el acabado final y que esa forma "compleja" de las nuevas piezas dificulta su obtención.

Asimismo, tanto la instalación de la estación como la programación del robot y el PLC la realizó una empresa externa. Hasta el momento, los operarios se limitaban a configurar parámetros prácticos y poner en marcha el proceso, pero no comprendían la programación ni el proceso de control, tareas que se reservaban al personal específico de la empresa instaladora.

Partiendo de esa base, surgen distintos aspectos que solucionar:

- Comprender al completo cuál es el objetivo de la empresa, qué se quiere hacer, cómo se quiere hacer y cuál es el objetivo final.
- Estudiar minuciosamente la instalación actual: conocer sus elementos, saber su función dentro del proceso y comprender el proceso completo de pintura.
- Aprender el lenguaje (**PolyScope**) y la forma de programar del robot, así como el funcionamiento y las características principales del mismo. Más adelante se profundizará en él.
- Analizar el programa actual para pintar piezas cilíndricas y tener un ejemplo del que tomar referencias.
- Encontrar una solución para la pintura de estas piezas de geometría compleja, donde el robot va a tener más protagonismo, ya sea sujetando él la pieza y moviéndose mientras se pinta, posicionándose de distintas formas y realizando varios agarres distintos para encontrar la mejor posición...
- Idear un esquema de programa donde se especifiquen los movimientos del robot adecuados a las dimensiones de las piezas.
- Diseñar las máscaras de protección de los orificios de las piezas (solo se debe pintar el exterior) y los carros donde se depositan las piezas (con sus soportes correspondientes).
- Recordar y ampliar los conocimientos de **TIA Portal** para la programación del PLC, donde se coordinan y comunican todas las partes del proyecto.
- Crear el programa definitivo para el robot y el PLC.

3. Descripción de la instalación

3.1. Introducción

La instalación fue diseñada para la pintura de las piezas cilíndricas anteriormente mencionadas. El robot coge las piezas de un carro y las deposita en la cabina de pintura, esperando a que se acabe de pintar, momento en el cual recoge la pieza y la coloca en el carro de dejada. Esta instalación la realizó una empresa externa, mientras que fue otra la encargada de realizar la comunicación y coordinación de la misma.



Figura 6. Instalación de la estación de pintura.

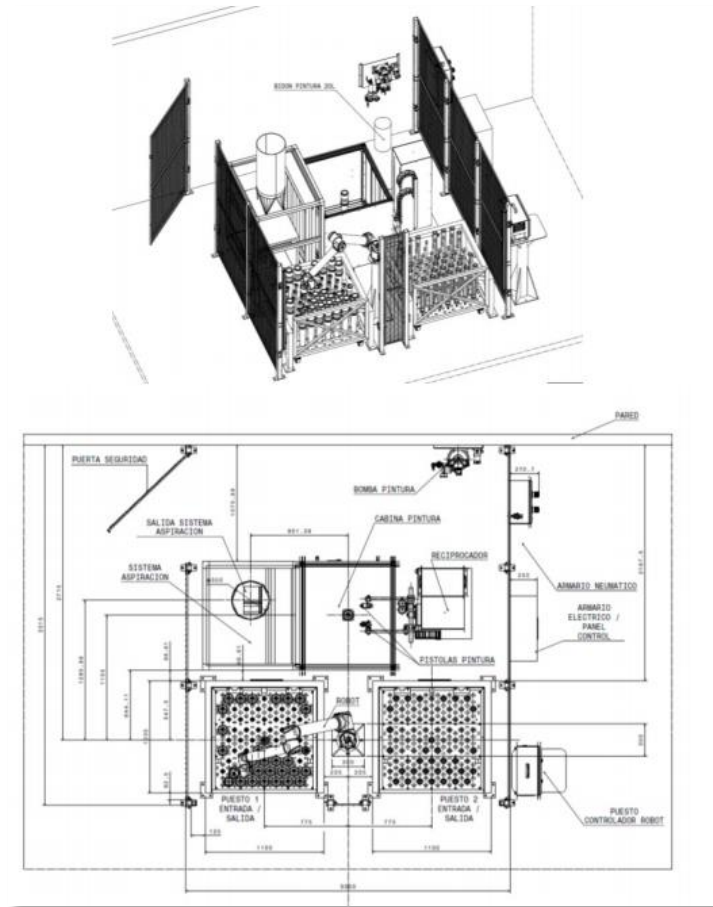


Figura 7. Planos de la instalación de pintura.

La instalación cuenta con un sistema de protección que consiste en una valla que rodea toda la estación de pintura. Las únicas zonas sin valla son los puestos de los carros, que se encuentran abiertos y que cuentan con un **sensor** que detecta la presencia de un agente externo y detiene la ejecución del proceso. Para acceder a la cabina de pintura o a parte de la instalación de bombeo de pintura, existe una puerta que cuenta con otro sensor que indica el estado de la puerta y bloquea el proceso en caso de que esté abierta.



Figura 8. Puerta de acceso y detalle del sensor.

Existe un tercer sensor ubicado en una de las vallas laterales a la altura de la entrada a la cabina de pintura, cuyo fin es detectar si el robot lleva o no pieza cuando va a dejarla en el soporte rotativo de la cabina y si al finalizar la pintura ha recogido la pieza correctamente para devolverla al carro.



Figura 9. Sensor detector de piezas.

Por último, destacar que el control del robot UR10e se encuentra en un pilar por fuera de la valla, mientras que tanto el armario de control como el panel de control de la presión del aire se encuentran colgados de la valla en la cara externa de esta.



Figura 10. Zona de control de la instalación.

A continuación, vamos a ver cada uno de los elementos principales que componen la instalación y la función que desempeñan en ella, tanto en el proceso anterior de las piezas cilíndricas como en el nuevo para las piezas en 'T' y 'X'.

3.2. Contenedores de transporte

Se trata de unos carros metálicos de acero con cuatro ruedas que cuentan con 64 soportes para colocar las piezas cilíndricas.

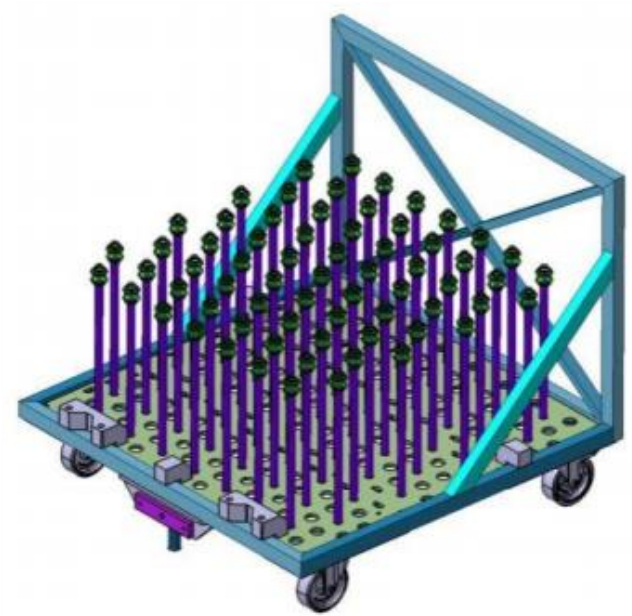


Figura 11. Contenedor de transporte.

Son los encargados de suministrar las piezas a la estación de pintura. Un operario coloca las piezas, lleva el carro hasta la estación de pintura y lo coloca en los puestos de entrada y salida de contenedores. Para este proyecto, se deben **rediseñar** con el fin de albergar las distintas piezas. Se variarán tanto las dimensiones y posiciones de los soportes como el número de piezas por carro, ya que se debe mantener la misma estructura exterior para que se puedan colocar en los puestos de la estación.

3.3. Puestos de entrada y salida de contenedores

Es una estructura metálica diseñada para colocar los carros en la estación en una posición fija para facilitar el acceso del robot a las piezas.



Figura 12. Puestos de entrada y salida de contenedores.

Se trata de dos zonas para colocar un carro en cada una, pudiendo utilizar un único carro para coger o dejar piezas o usar un carro distinto para cada tarea. Consta de dos sensores por puesto que detectan la presencia de un carro para verificar que se puede iniciar el proceso y de un par de imanes que se activan manualmente para fijar los carros a la instalación y que no se muevan durante la ejecución.

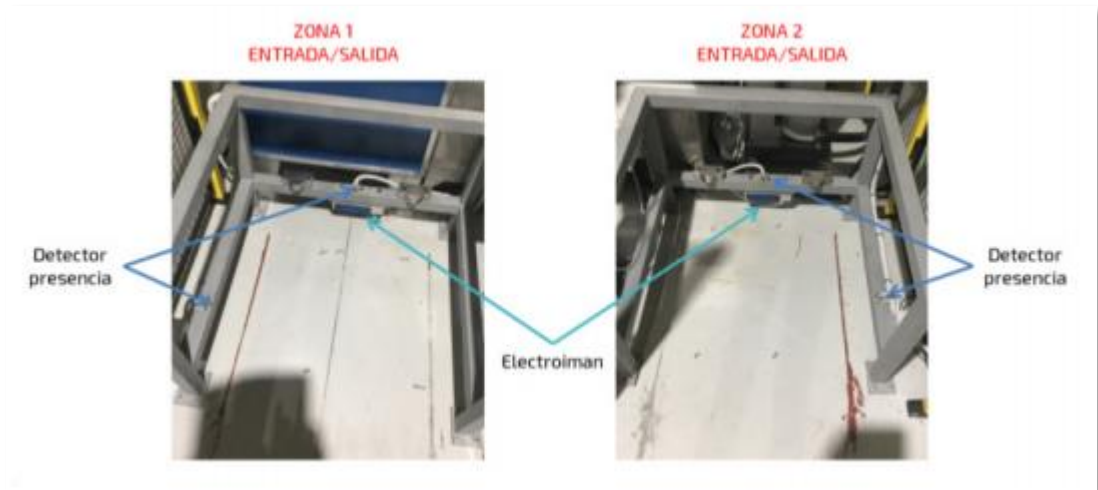


Figura 13. Sensores e imán en el puesto de entrada y salida.

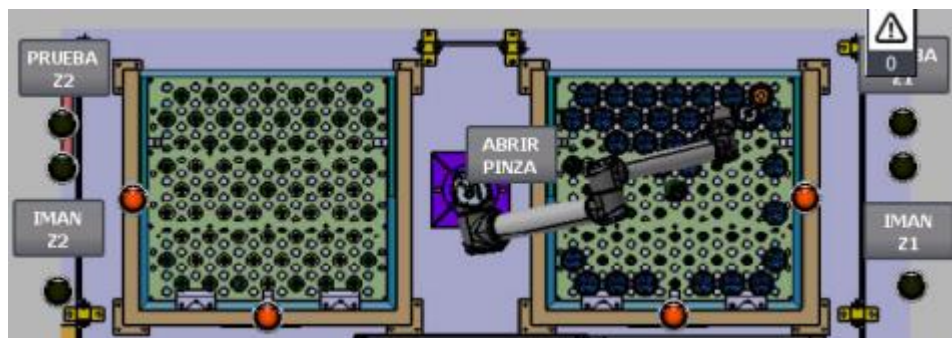


Figura 14. Representación en la pantalla HMI de imanes y sensores.

3.4. El robot

El UR10e de Universal Robots es un brazo robótico industrial colaborativo con una capacidad de carga útil (10 kg). Su alcance de 1300mm abarca amplios espacios de

trabajo sin comprometer la precisión o el rendimiento de la carga útil. Este robot se emplea en distintas aplicaciones, como carga y descarga, paletizado y empaquetado [2], [3].



Figura 15. Robot UR10e.

En la actual instalación está ubicado entre los puestos de los carros, sobre un pilar para poder acceder tanto a las piezas de los carros como al interior de la cabina de pintura.



Figura 16. Ubicación del robot en la instalación.

En la estación de pintura se encarga del transporte de las piezas desde los carros donde llegan a la instalación hasta la cabina de pintura, donde las coloca en el soporte rotativo para que sean pintadas y se retira. Una vez finalizado el proceso de pintura, el robot recoge la pieza y la deposita en el carro destino, ya sea el carro de origen u otro que haya sido asignado.

Por motivos económicos se ha optado por mantenerlo en el nuevo proyecto y será el gran protagonista, ya que tendrá mucha más relevancia.

En apartados posteriores se profundizará más en los detalles del robot.

3.5. Armario de control

Contiene los elementos principales de control de la instalación, así como las conexiones de entradas y salidas necesarias para el proceso de pintura.

Como ya se ha explicado, se ubica en uno de los laterales de la instalación, entre el control del robot y el regulador de la presión.



Figura 17. Armario de control.

Consta de dos partes principales:

3.5.1. PLC o autómeta

Controla todo el proceso y coordina todos los elementos de la instalación para que esta funcione correctamente.



Figura 18. PLC del armario de control.

Se trata de un PLC SIEMENS S7 1214 DC/DC/DC, junto con un módulo de entradas y salidas digitales SM 1223 [4], [5].

Con el programa creado en TIA Portal se encarga de sincronizar los elementos de la estación para ejecutar el proceso.

En este proyecto se ha modificado el programa para poder realizar la solución para las nuevas piezas.

Como con el robot, más adelante se analizará de forma más detallada el PLC y los programas de TIA Portal.

3.5.2. Pantalla HMI

Una pantalla HMI SIEMENS KTP700 Basic que sirve como interfaz de control para el operario. Desde ella, se puede controlar y configurar de forma superficial la instalación y el proceso que se lleva a cabo en ella [6].

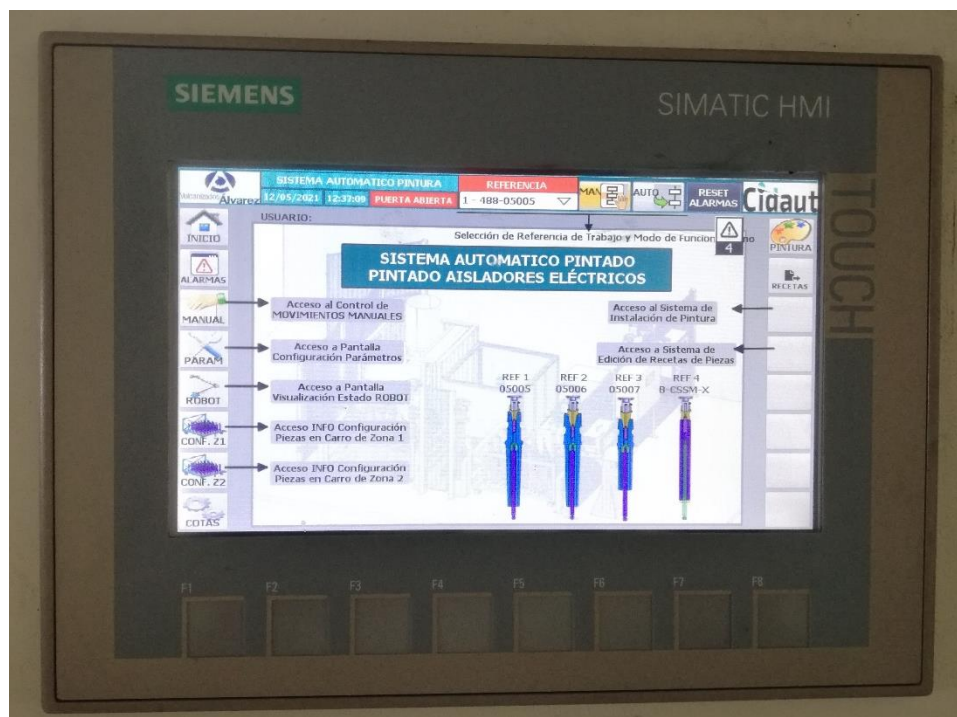


Figura 19. Pantalla HMI.

La pantalla cuenta con un formato de imagen con varios elementos fijos que se encuentran en los marcos, como pueden ser los logos de Vulcanizados Alvarez y de la empresa que lo realizó, la hora actual o si la puerta de acceso a la cabina está abierta o cerrada. A su vez, contiene varias pantallas con las que interactuar:

- Inicio



Figura 20. Pestaña "Inicio" de la pantalla HMI.

Se trata de una pantalla meramente informativa, donde se describe por encima qué son el resto de pestañas.

- Alarmas

Avisos no acusados			
N.º	Hora	Fecha	Texto
5	8:08:25	31/07/2020	CICLO AUTOMATICO NO AUTORIZADO, EXTRACTOR CABINA NO ESTA EN MARCHA
10	7:16:09	31/07/2020	ALARMA ZONA 2 SIN COTAS
7	7:16:09	31/07/2020	ALARMA ZONA 1 SIN COTAS
4	7:16:09	31/07/2020	DISPARO TERMICO MOTOR ASPIRACION
1	7:16:09	31/07/2020	DISPARO TERMICO MOTOR GIRO

Figura 21. Pestaña "Alarmas" de la pantalla HMI.

En ella se recogen las distintas alarmas que se generan en la instalación, incluidas las producidas durante la realización del proceso de pintura. Cuando se produce una alarma por la detección de intruso por parte del sensor o el fallo en el suministro de aire salta una pestaña en la que indica rápidamente el problema.

- Manual



Figura 22. Pestaña "Manual" de la pantalla HMI.

Esta pestaña permite controlar parcialmente algunos de los elementos de la instalación sin necesidad de ejecutar un programa. Gracias a ella, visualizamos el estado de la instalación y podemos activar o desactivar los imanes para los carros, la pinza del robot, los gatillos de las pistolas e incluso subir y bajar estas últimas.

Es importante destacar que se ha comprobado que varias de las funciones expuestas anteriormente no funcionan correctamente. Así por ejemplo, el control de la pinza o de la posición de las pistolas debido a problemas de conexión y programación durante el montaje de la instalación. Esto será un gran inconveniente de cara

a la realización del proyecto, puesto que limita seriamente la posibilidad de realizar ciertas pruebas de pintura. Como no es posible coger piezas o pintarlas en modo manual, se deberá realizar un programa completo teniendo en cuenta todas las entradas, salidas y variables para evaluar si la solución escogida es correcta.

- Parámetros

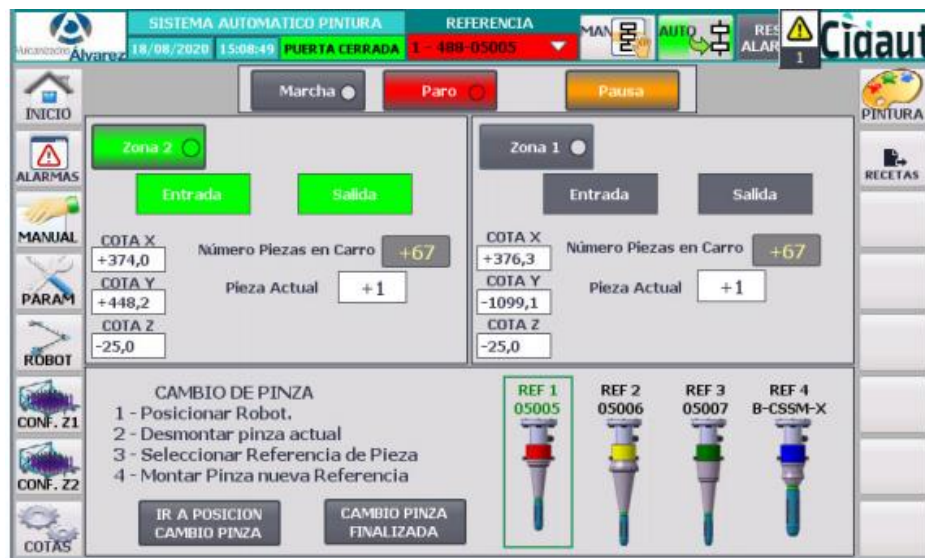


Figura 23. Pestaña "Parámetros" de la pantalla HMI.

Gracias a esta pantalla se pueden configurar distintas opciones, como la marcha o la parada del proceso automático, las cotas de las piezas de cada carro o el número de pieza con el que está trabajando. A su vez, cuenta con varios indicadores para conocer el estado actual del proceso o si el robot ha cogido o no la pieza.

- Robot

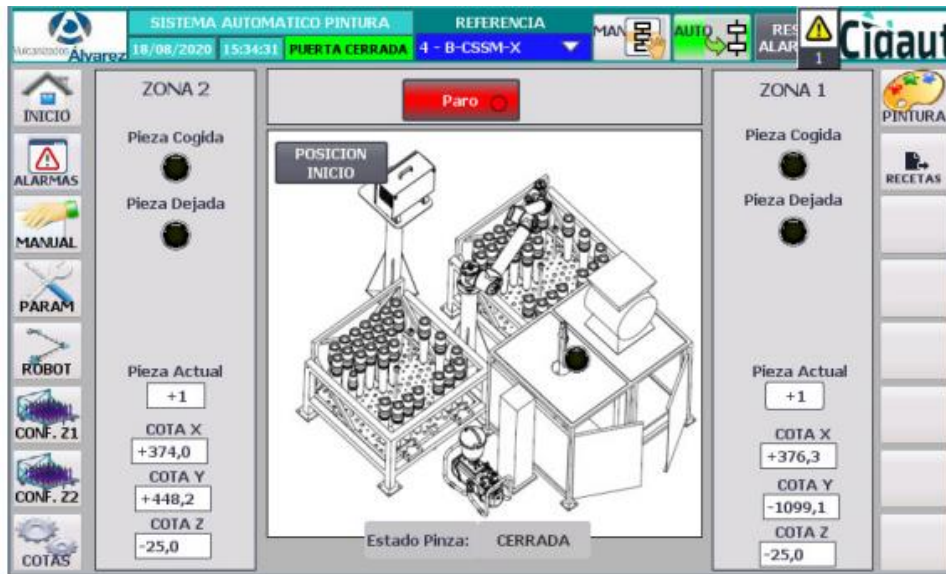


Figura 24. Pestaña "Robot" de la pantalla HMI.

Pantalla para la visualización del estado del Robot, en ella se informa al operador de la acción que está efectuando en cada instante el robot ya esté trabajando en la Zona 1, Zona 2 o en la zona de pintura.

Mediante el botón "POSICION INICIO" el operador puede desplazar al robot a la posición de reposo ubicada entre ambas zonas de contenedor y la cabina de pintura. En la parte inferior se puede ver el estado de la Pinza (cerrada o abierta) en la que se encuentra en cada instante.

- Configuración Zona 1



Figura 25. Pestaña "Configuración Zona 1" de la pantalla HMI.

Esta pantalla es informativa y permite al usuario conocer con cuantos aisladores eléctricos va a trabajar el sistema y donde deben ser colocados según el número de piezas totales en contenedor que se tengan configurados para la zona 1.

- Configuración Zona 2



Figura 26. Pestaña "Configuración Zona 2" de la pantalla HMI.

Es el mismo concepto de la pestaña anterior pero referida a la zona 2.

- Cotas



Figura 27. Pestaña "Cotas" de la pantalla HMI.

Esta pantalla permite el ajuste de las cotas de cogida de piezas en cada una de las zonas de ubicación de contenedores. Esta pantalla siempre debe ser utilizada por un usuario especializado por lo que el acceso a la misma está protegido con contraseña.

Para el ajuste de las cotas de ubicación de las piezas en cada uno de los contenedores el usuario debe introducir las posiciones de cogida/dejada de cada una de las coordenadas de orientación del robot (X, Y, Z) para los cuatro puntos de las esquinas de los contenedores (P11, P17, P17 y P97). Una vez introducidas todas las cotas correctamente se debe presionar el botón "Calcular COTAS". El sistema calculará de forma automática todas las cotas del resto de puntos de ubicación de las piezas.

- Pintura

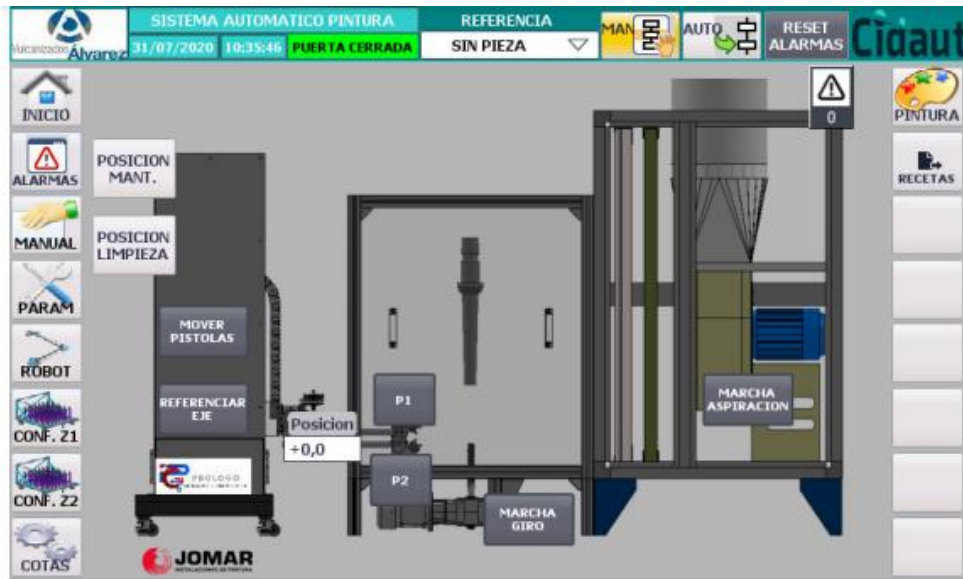


Figura 28. Pestaña "Pintura" de la pantalla HMI.

Pantalla específica para la visualización del estado de la instalación de pintura y el control de los diferentes actuadores existentes en la misma.

- Recetas

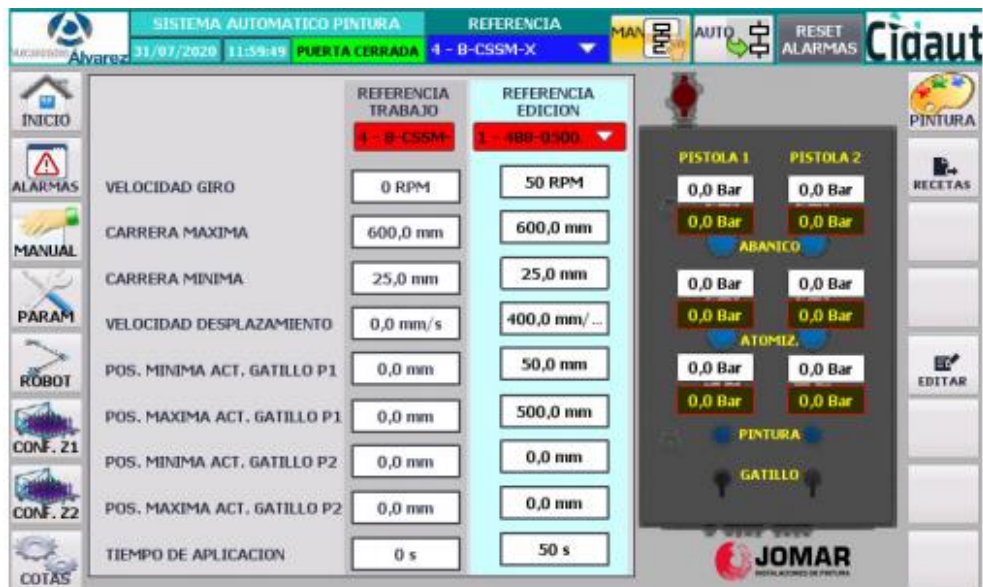


Figura 29. Pestaña "Recetas" de la pantalla HMI.

Finalmente, esta pestaña nos muestra las 4 referencias existentes para los distintos tipos de piezas que se pintan en la estación. En cada una de ellas se nos indican los distintos valores del tiempo de pintura, la carrera de las pistolas o la velocidad de las mismas, que es información que se envía al PLC para que la interprete. También encontramos en la zona de la derecha una representación del panel de control de presión, donde se indican los valores estipulados para cada una de las referencias.

En esta pantalla además aparece un botón a la derecha que nos permite editar estas recetas, cambiando los valores que consideremos oportunos para optimizar el proceso de pintura.

Desde esta pantalla se introducen las recetas de pintura para cada una de las cuatro referencias con las que se puede trabajar en la instalación. Se pueden observar los valores que tiene actualmente la referencia con la que se está trabajando y los valores de la referencia que se haya seleccionado para su edición.

3.6. Sistema de pintura

Este conjunto de elementos es el encargado de la tarea principal de la instalación, la pintura de las piezas, junto con el extractor. A continuación, describiremos sus componentes principales:

3.6.1. Cabina de pintura

Una estructura formada por tres chapas metálicas, una de ellas extraíble, que, junto con el sistema de extracción, forman un cubo abierto en la parte superior para que acceda el robot.



Figura 30. Cabina de pintura.

En su interior se encuentra el soporte rotativo, donde el robot coloca las piezas para que se pinten. Se trata de un soporte metálico conectado a un variador de frecuencia que, al recibir la instrucción correspondiente, activa el giro del mismo.



Figura 31. Soporte rotativo de la cabina de pintura.

3.6.2. Reciprocador y pistolas

Conjunto encargado de la pintura de la pieza. Consta de un reciprocador Festo, una estructura basada en un servomotor que controla la posición y la velocidad de las pistolas. Estas constan de un gatillo, de forma que, al abrirse este, pulverizan la pintura sobre la pieza.



Figura 32. Reciprocador.



Figura 33. Pistolas de pintura.

En el proceso de pintura de piezas cilíndricas, está programado para que, cuando el robot coloque la pieza en el soporte rotativo y se retire, se active el proceso de pintura, realizándose dos acciones simultáneas:

- El reciprocador hace subir y bajar las pistolas, dentro del rango de carrera máxima y mínima que se le ha estipulado, a una velocidad y durante un tiempo programados.
- Las pistolas accionan el gatillo y comienzan a pulverizar la pintura sobre la pieza, también dentro de un rango de posiciones predefinido.

3.6.3. Sistema de extracción

Es un sistema de aspiración, destinado a absorber la pintura que no se adhiere a la pieza y las partículas en suspensión que podrían interferir en la homogeneidad del acabado y, por tanto, en las características eléctricas y el comportamiento futuro de la pieza.

Para realizar este cometido, se dispone de una célula que aspira con gran potencia de succión y una chimenea que comparte con un horno (que se utiliza para **curar** las piezas tras pintarlas. En el proceso de curado se fija definitivamente la pintura a la pieza, ya que hasta entonces la pintura pulverizada se encuentra en un estado sólido, similar a polvo).



Figura 34. Sistema de extracción.

3.6.4. Sistema de suministro de pintura

La pintura, tras ser preparada correctamente por el operario, se deposita en un contenedor conectado a un sistema que bombea la pintura hasta las pistolas de la cabina de pintura. Cuenta con un sistema de conexiones y válvulas regulables para seleccionar el caudal y la presión adecuadas dependiendo de la pieza a pintar o del tipo de pintura utilizado.



Figura 35. Sistema de suministro de pintura.

3.6.5. Sistema de suministro de aire

Para el accionamiento de la pinza del robot y el bombeo y pulverización de pintura se necesita un sistema que genere aire a presión. Este sistema consta de un panel situado al lado del armario de control en el que se pueden regular manualmente los valores de presión de:

- Abanico
- Atomización
- Pintura

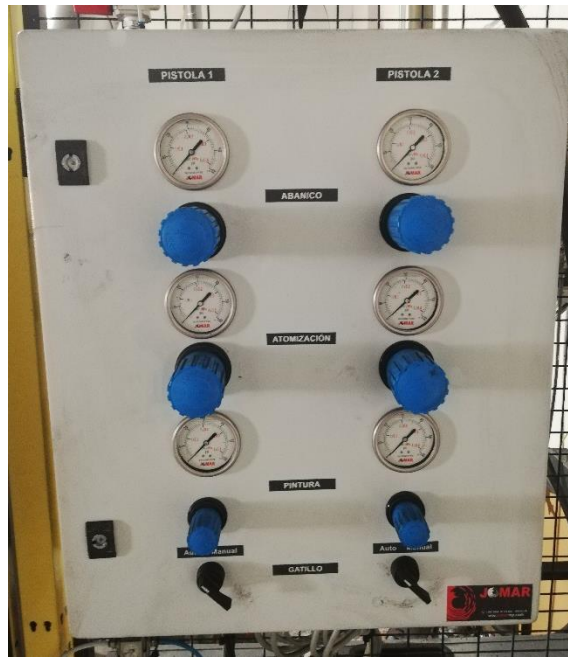


Figura 36. Panel de control de presión.

Dentro del recinto vallado de la estación y encima del recipiente de pintura, se sitúa otra parte del sistema, la encargada de generar la presión que llega a la pinza y las pistolas.

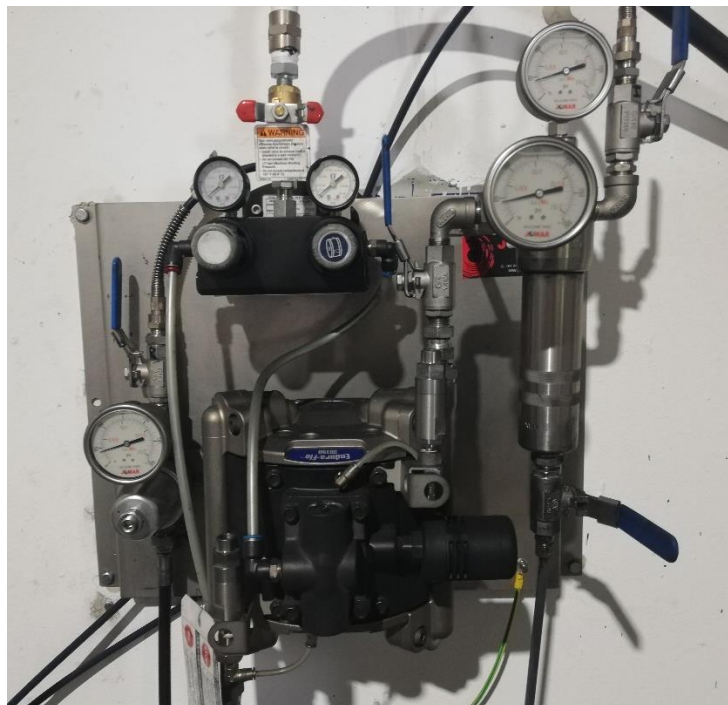


Figura 37. Sistema generador de presión.



4. El proceso de pintura de piezas cilíndricas

4.1. Introducción

Una vez definida la instalación en la que se desarrolla el proyecto podemos comenzar a analizar el proceso que se estaba llevando a cabo en la empresa hasta la actualidad: pintado las piezas cilíndricas que sirven como empalmes del tendido eléctrico de alta tensión. Tras describirlo, posteriormente se analizarán las modificaciones en el mismo encaminadas a la pintura de piezas con una geometría más compleja.

En primer lugar, vamos a ver los diferentes tipos de piezas cilíndricas que se pintan con este proceso:



Figura 38. Tipos de piezas cilíndricas.

Asociado a estas piezas, se tienen definidas cuatro referencias dentro del programa del PLC y la pantalla HMI en las que se especifican distintos valores necesarios para pintar adecuadamente cada una de las piezas.

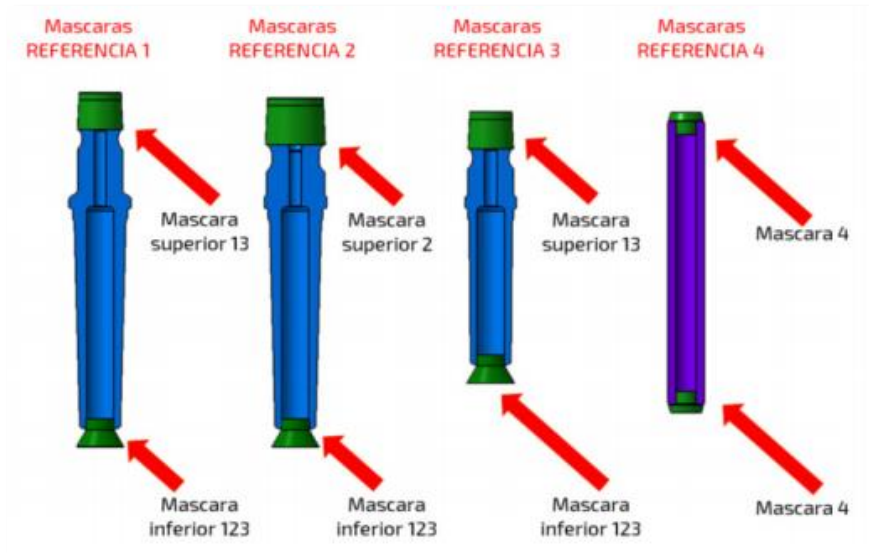


Figura 39. Referencias para las piezas cilíndricas.

Por cada una de esas referencias también se tienen una serie de pinzas y soportes.



Figura 40. Pinzas y soportes para las piezas cilíndricas.

Estas pinzas se colocan en el robot mediante un adaptador, que ya tiene incorporado, y se trata de las herramientas encargadas de coger las piezas del carro y llevarlas a la cabina de pintura. Estas pinzas las proporcionó la empresa responsable de la instalación, y están diseñadas para adaptarse a las dimensiones de las diferentes piezas. Su funcionamiento se basa en la introducción de la pinza en el orificio de la pieza para posteriormente insuflar aire a presión que expande tres membranas que funcionan a modo de dedos que sujetan la pieza.



Figura 41. Detalle de la pinza.

Los soportes anteriormente mencionados se ubican en la cabina de pintura, y son los encargados de sujetar la pieza mientras se pinta. Estos soportes rotan durante el proceso de pintura gracias a un variador de frecuencia para que la pieza se pinte de forma homogénea. Se dispone de varios modelos para los distintos tipos de piezas, que se pueden intercambiar tras retirar un tornillo y desenroscar el soporte.

Por último, se deben colocar las máscaras adecuadas a cada tipo de pieza. Se tratan de piezas mecanizadas en aluminio cuyas dimensiones coinciden de forma

que se encajan por presión manual en los orificios de las piezas a pintar para impedir que entre pintura en el interior y modifique sus propiedades eléctricas.



Figura 42. Máscaras de las piezas cilíndricas.

4.2. Análisis del proceso

El operario coloca un carro lleno de piezas en uno de los puertos de la instalación, comprueba que las pistolas están bien colocadas y coloca el soporte rotativo preparado para el tipo de pieza que se va a pintar. Tras asegurarse de cerrar la puerta de acceso a la cabina, activa la ejecución del programa para realizar la tarea “Cambio_Pinza”, para poder colocar la pinza adecuada. Posteriormente introduce en la pantalla HMI el número de piezas que contiene el carro, el puerto del carro de cogida (que contiene las piezas), el carro de dejada (puede ser el mismo o del otro puerto) y ajusta valores de configuración respecto a las dimensiones de la pieza y la presión de las pistolas de pintura.

A continuación, estando la estación en modo automático, se inicia la ejecución del proceso de pintura.

El primer movimiento que realiza el robot es la reubicación, moviéndose desde la posición en la que se encuentre a la posición inicial.



Figura 43. Robot en posición inicial.

Una vez alcanza esta posición el robot se aproxima a uno de los contenedores (según el que se haya programado) para coger una pieza.



Figura 44. Robot cogiendo pieza de Zona 2.

El robot con la pieza se desplaza hasta la posición 'Detección_Pieza', donde por medio del sensor se detecta si efectivamente el robot ha cogido la pieza de forma correcta. Si se enciende el led naranja en el sensor, significa que este ha detectado la presencia de la pieza.

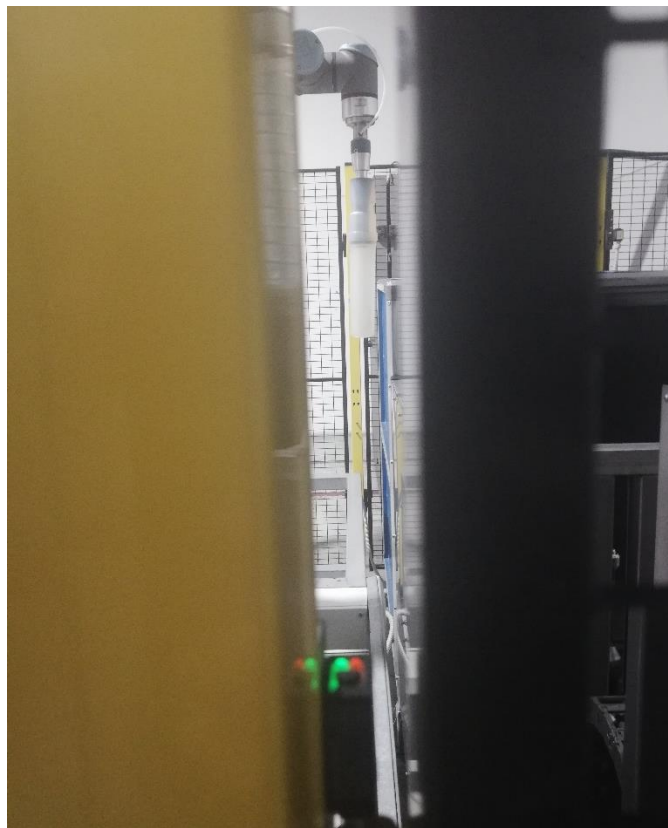


Figura 45. Posición detección de pieza + detalle del sensor.

Tras corroborar que el robot transporta una pieza, este accede a la cabina de pintura para depositarla en el soporte rotativo y se retira para que comience la pulverización de pintura.



Figura 46. Robot dejando la pieza en la cabina.

Una vez se ha retirado completamente el robot, las pistolas comienzan un ciclo que se repite dos o cuatro veces (según la referencia), donde estas ascienden y se inicia la pulverización de pintura según la referencia seleccionada hasta que llegan al límite superior y comienzan a descender hasta el punto inicial.

Al finalizar el ciclo de pintura, se da la orden al robot para que este recoja la pieza y, tras revisar mediante el sensor que se ha cogido correctamente, la deposita en el soporte del carro de dejada.



5. UR10e y PolyScope

5.1. Descripción del robot

Los robots de Universal Robots e-Series están equipados con una gama de funciones de seguridad incorporadas así como E/S de seguridad, señales de control digital o análogo hacia o desde la interfaz eléctrica, para conectar con otras máquinas y dispositivo de protección adicionales.

El UR10e es un robot industrial colaborativo extraordinariamente versátil que ofrece una gran carga útil (12,5 kg) de elevación y un largo alcance (1300 mm), por lo que es ideal para una amplia gama de aplicaciones, como el mantenimiento de máquinas, el paletizado y el embalaje [2], [3].

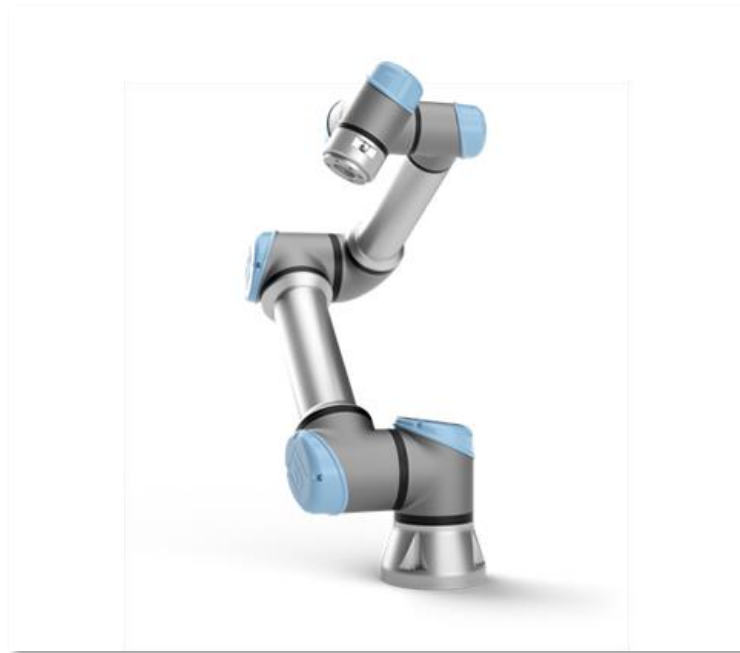


Figura 47. Robot UR10e.

Sus propiedades más interesantes son las siguientes:

- Se trata de un robot colaborativo, por lo que permite el trabajo de forma conjunta con un operario sin que este corra ningún riesgo. En caso de colisión o aprisionamiento, el robot detecta una resistencia inesperada

(incluso si choca con un obstáculo fijo) y realiza una parada de emergencia, finalizando su ejecución y mostrando una alarma por pantalla.

- Tiene un alcance de 1300 mm, siendo el brazo con mayor rango del fabricante. Esto permite acceder y trabajar en zonas relativamente alejadas de la base del robot.

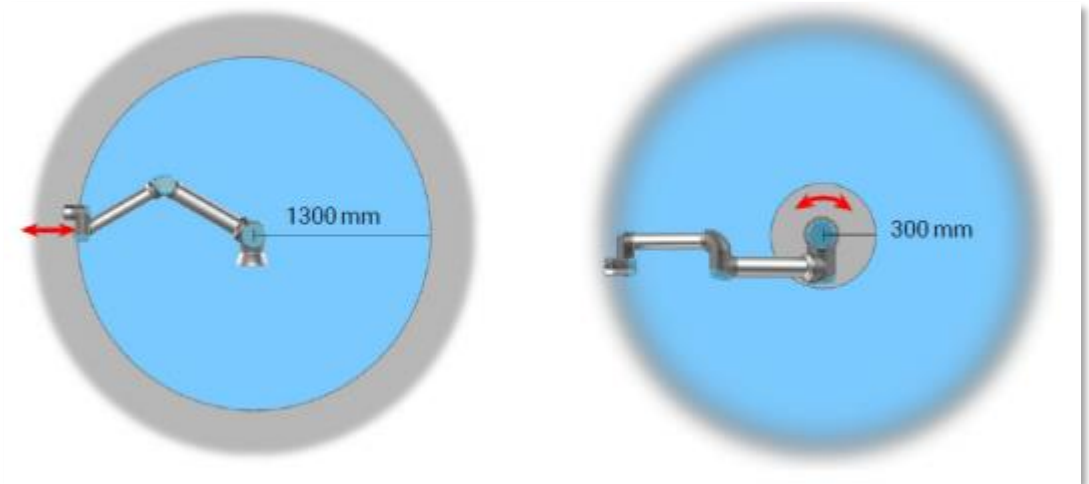


Figura 48. Alcance del robot.

- Tiene una carga útil de 10 Kg, por lo que podremos utilizarlo para mover objetos de un peso considerable.
- Permite una precisión de ± 0.05 mm, la cual definimos como la distancia entre el valor medio de los puntos que recorre el robot para alcanzar un punto programado y el punto programado. Cuanto más alejado este el brazo del robot de la base, menor será su precisión.
- El brazo pesa 33.5 Kg, por lo que su instalación y transporte resulta sumamente sencillos.

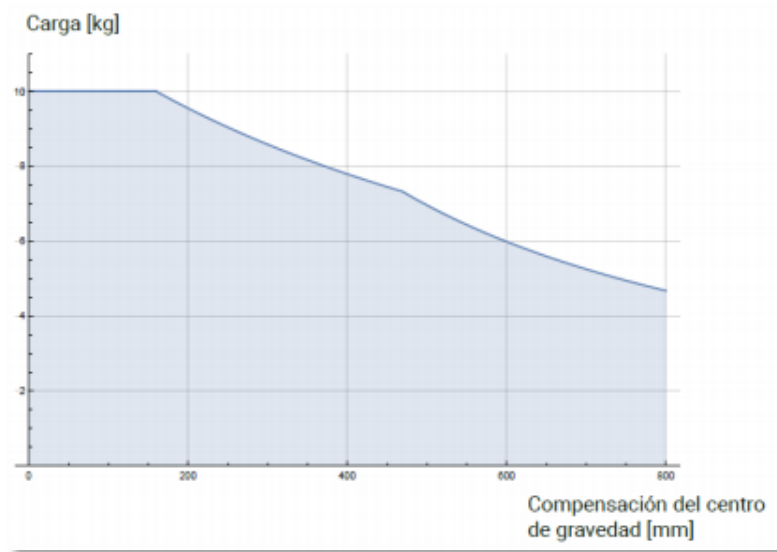


Figura 49. Relación entre carga en Kg y la compensación del centro de gravedad en mm.

- El robot posee 6 grados de libertad que son esenciales para obtener una capacidad de orientación de la herramienta que lleve conectada a su extremo en todo su alcance. Cada grado de libertad viene definido por el número de articulaciones que posee.

A continuación se muestran otras características principales del robot, según se especifica en su ficha técnica [7]:

Rendimiento

Consumo de energía	Aprox. 350W para un programa típico		
Operación de colaboración	17 funciones de seguridad ajustables avanzadas, incluyendo supervisión del codo Control remoto de acuerdo con ISO 10218		
Certificaciones	EN ISO 13849-1, Cat.3, PL d, y EN ISO 10218-1		
Sensor F/P - Fuerza, X-Y-Z	Sensor F/P - Par, X-Y-Z		
Rango	100 N	Rango	10 Nm
Resolución	2,0 N	Resolución	0,02 Nm
Precisión	5,5 N	Precisión	0,60 Nm
Rango de temperatura ambiente	0–50°C		
Humedad	90%RH (sin condensación)		

Especificación

Carga útil	10kg / 22lbs
Alcance	1300mm / 51,2in
Grados de libertad	6 articulaciones giratorias
Programación	Interfaz gráfica del usuario PolyScope con pantalla táctil de 12" con soporte

Movimiento

Repetibilidad de posición	+/-0,05mm con carga, según ISO 9283	
Movim. del eje del brazo robot.	Radio de acción	Velocidad máxima
Base	± 360°	± 120°/s
Hombro	± 360°	± 120°/s
Codo	± 360°	± 180°/s
Muñeca 1	± 360°	± 180°/s
Muñeca 2	± 360°	± 180°/s
Muñeca 3	± 360°	± 180°/s
Velocidad típica de TCP	1 m/s / 39,4 in/s	

Funciones

Clasificación IP	IP54	
Clase ISO Sala limpia	5	
Ruido	Menos de 65dB(A)	
Montaje del robot	Cualquier orientación	
Puertos de E/S en herramienta	Entrada digital	2
	Salida digital	2
	Entrada analógica	2
	Salida analógica	0
	Interface UART (9.6k-5Mbps)	
E/S de fuente de alimentación en herramienta	12V/24V 600mA continuos, 2A por periodos cortos	

Características físicas

Huella	Ø 190 mm
Materiales	Aluminio, Plásticos de PP, Acero
Tipo de conector para herramienta del robot	M8 M8 8-pin
Long. cable del brazo robótico	6m / 236in
Peso incluyendo cable	33,5 kg / 73,9lbs



Caja de control

Funciones

Clasificación IP	IP44
Clase ISO Sala limpia	6
Rango de temperatura ambiente	0–50°C
Puertos de E/S	Entrada digital 16 Salida digital 16 Entrada analógica 2 Salida analógica 2 Control a 500 Hz, 4 entradas digitales en cuadratura de alta velocidad dedicadas
E/S de fuente de alimentación	24 V 2A
Comunicación	Frecuencia de control: 500 Hz ModbusTCP: Frecuencia de señal de 500 Hz ProfiNet y EthernetIP: Frecuencia de señal de 500 Hz Puertos USB: 1 USB 2.0, 1 USB 3.0
Fuente de alimentación	100-240VAC, 47-440Hz
Humedad	90%RH (sin condensación)

Características físicas

Tamaño de la caja de control (anch. x alt. x prof.)	475mm × 423mm × 268mm / 18,7in × 16,7in × 10,6in
Peso	Máx. 13,6kg / 30,0lbs
Materiales	Acero

Consola de programación

Funciones

Clasificación IP	IP54
Humedad	90%RH (sin condensación)
Resolución de pantalla	1280 x 800 píxeles

Características físicas

Materiales	Plástico
Peso incluyendo 1 m de cable de TP	1,6kg / 3,5lbs
Longitud del cable	4,5m / 177,17in

5.2. El robot en la instalación

Ahora que ya conocemos el robot, vamos a centrarnos en su ubicación y funcionamiento dentro de la instalación.

En primer lugar, ubicaremos el robot y el armario de control con la 'Teach Pendant' en el esquema de la estación de pintura.

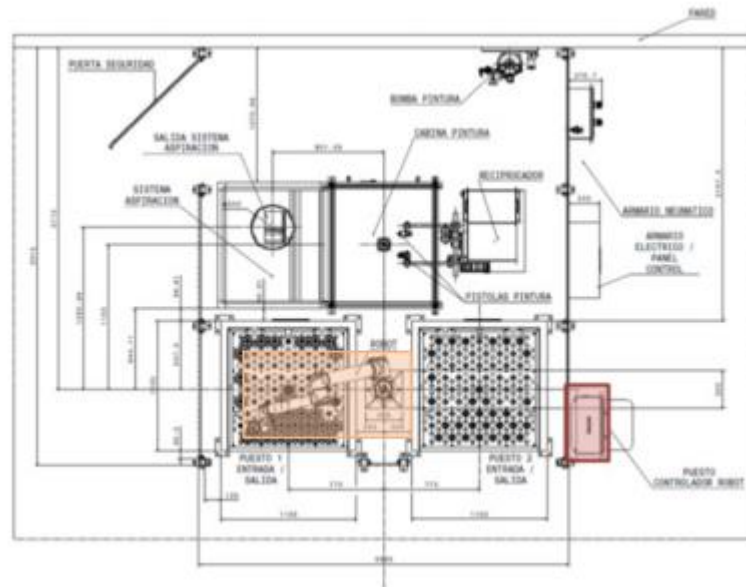


Figura 50. Robot y su armario de control en la instalación.

Cada uno de los elementos tiene una función específica dentro de la instalación. En la caja de control se sitúan los grupos de interfaz E/S con los puertos que permiten las conexiones adicionales. En la parte inferior nos encontramos con la interface Ethernet que nos permite conectarnos al PLC y al HMI, desde donde recibirá mediante PROFINET las instrucciones para realizar los movimientos programados. Además, esta interfaz nos permite tener control y acceso remoto al robot.

También nos encontramos con un puerto específico donde irá conectada la 'Teach Pendant', un slot para una tarjeta SD Card que contiene el sistema operativo del robot (PolyScope), dos puertos USB, un mini DisplayPort y un fusible de 10A



Figura 51. Controlador del robot colaborativo.

Todas las conexiones del robot, la entrada y salidas de cables están realizadas desde la parte interior del controlador del robot, a través de unos orificios diseñados para ello. El controlador del robot está conectado eléctricamente al cuadro general, por lo que el encendido y el apagado estarán condicionados a ello.

Para la manipulación de piezas desde los contenedores de transporte hasta la cabina de pintura y su posterior retorno a los contenedores, la instalación posee un robot colaborativo (UR10e) con garras neumáticas que permiten coger los aisladores eléctricos. En función de la referencia que se esté pintando en la instalación, la garra debe ser una u otra.

Al robot colaborativo se le ha añadido una placa intermedia para posibilitar un intercambio sencillo y rápido de cada una de las garras de amarre. No es necesario montar/desmontar este componente cada vez que se realiza el cambio de garra de amarre.

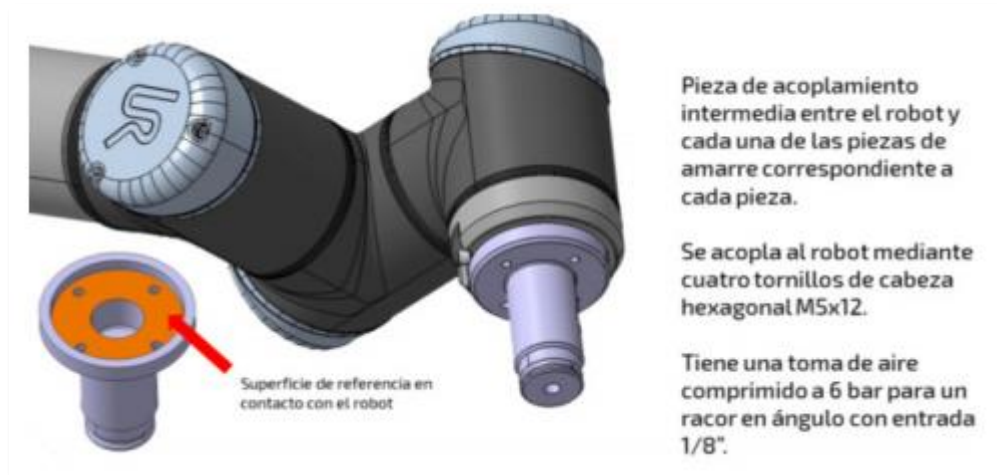


Figura 52. Placa intermedia de la garra.

Se ha establecido el siguiente código de colores para identificar/montar la garra adecuada en el robot colaborativo en función de la referencia que se vaya a pintar.



Figura 53. Garras y pinzas según la referencia.

Las pinzas 123 son intercambiables para las garras de amarre de las referencias 1, 2 y 3. La pinza 4 es única para la referencia 4.

5.3. Interfaz gráfica

La consola portátil nos permite configurar y ejecutar el programa que maneja robótico y la caja de control. Está basado en la herramienta grafica llamada PolyScope desarrollada por Universal Robots [2].

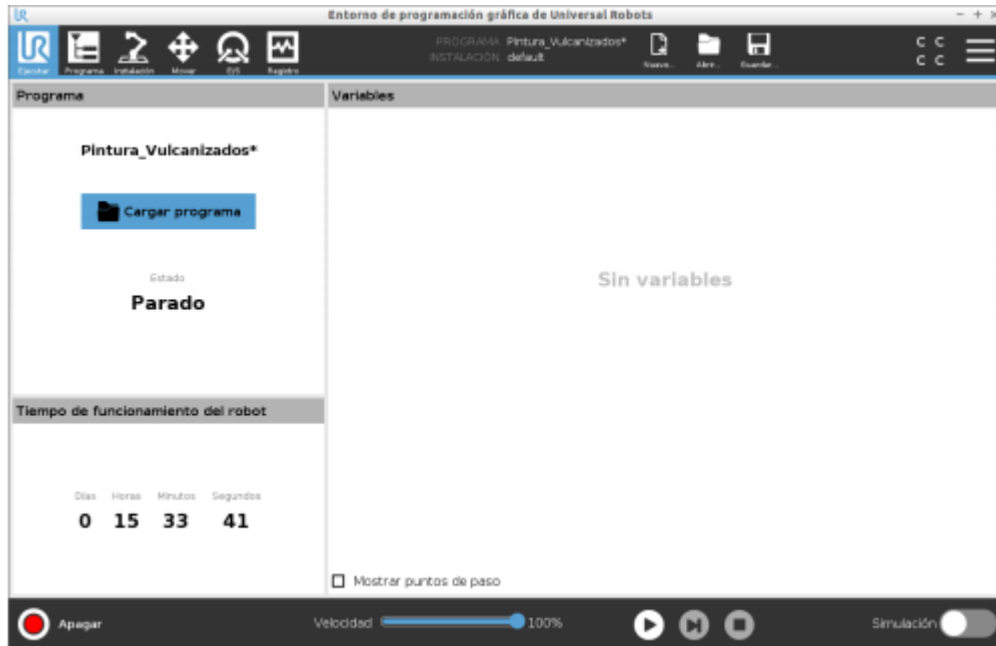


Figura 54. Interfaz de usuario.

En esta interfaz nos encontramos con los siguientes iconos / pestañas programa ya cargado.



Ejecutar permite la puesta en marcha del programa ya cargado.



Programa crea o modifica el programa del robot.



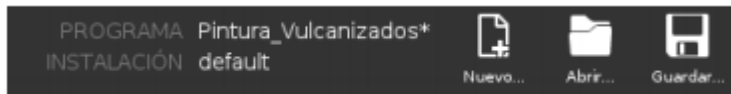
Instalación configura los ajustes del brazo robótico, montaje y seguridad.



Mover controla o regula el movimiento del manual del robot



Registro indica el estado del robot y mensajes de advertencia o error.



Gestor de programas y

de la instalación. Muestra el programa activo, la ruta del archivo y permite crear un nuevo programa, abrir uno ya existente o guardar el programa actual.



Modo Automático indica que el robot está funcionando en un entorno automático. Si se pulsa pasaremos a un control manual. El modo de funcionamiento del robot será siempre en automático ya que está controlado directamente por el PLC, desde donde serán enviadas las órdenes para arrancar, parar o mover el robot.



Modo Manual indica que el robot tiene cargado un entorno manual. Al ser pulsado pasará a control automático.



Modo Remoto indica que el robot está controlado por el PLC. Las funciones del resto de iconos se encuentran deshabilitadas en este modo de funcionamiento. Si se pulsa este botón se pasará a modo local.



Modo Local indica que el robot se puede controlar de forma local. Al pulsar el botón cambiaremos a Modo Remoto. Para cambiar a modo local, se deberá introducir por pantalla la contraseña asignada a esta función.



Seguridad muestra la configuración de seguridad activa.



Menú general permite el acceso a la ayuda Inicializar gestiona el estado del robot.



Inicializar gestiona el estado del robot. Cuando el robot es alimentado eléctricamente desde el cuadro general este botón estará en rojo por seguridad y el robot estará bloqueado con los frenos internos. Debemos de pulsarlo para poner el robot en estado operativo.



Control de velocidad muestra en tiempo real la velocidad a la que se mueve el brazo robótico, teniendo en cuenta los ajustes de seguridad y el modo de funcionamiento en el que nos encontremos. Igualmente, la velocidad se verá reducida cuando las barreras de seguridad de la instalación sean atravesadas.



Simulación permite ejecutar el programa en modo simulación, lo que nos permite visualizar los movimientos del robot en la pantalla sin que el robot real se mueva. Este modo es el que se utiliza en el software del ordenador URSim, donde podemos crear y ejecutar programas visualizando los movimientos que realizaría el robot físico. Posteriormente, se reproduce el programa en la instalación para comprobar que su funcionamiento es correcto al 100%.



Reproducir en modo manual inicia el programa del robot cargado actualmente. En condiciones normales de funcionamiento, la función de este botón esta desactivada ya que se encuentra controlado de manera remota.



Paso a paso permite ejecutar el programa en única etapa. Al igual que en el caso anterior, esta función esta desactivada al controlarse de manera remota.



Parar detiene el programa que tiene cargado el robot. Esta función esta desactivada en modo remoto

El apartado más interesante es el de **Programa**, desde donde se realiza la programación que va a llevar a cabo el robot.

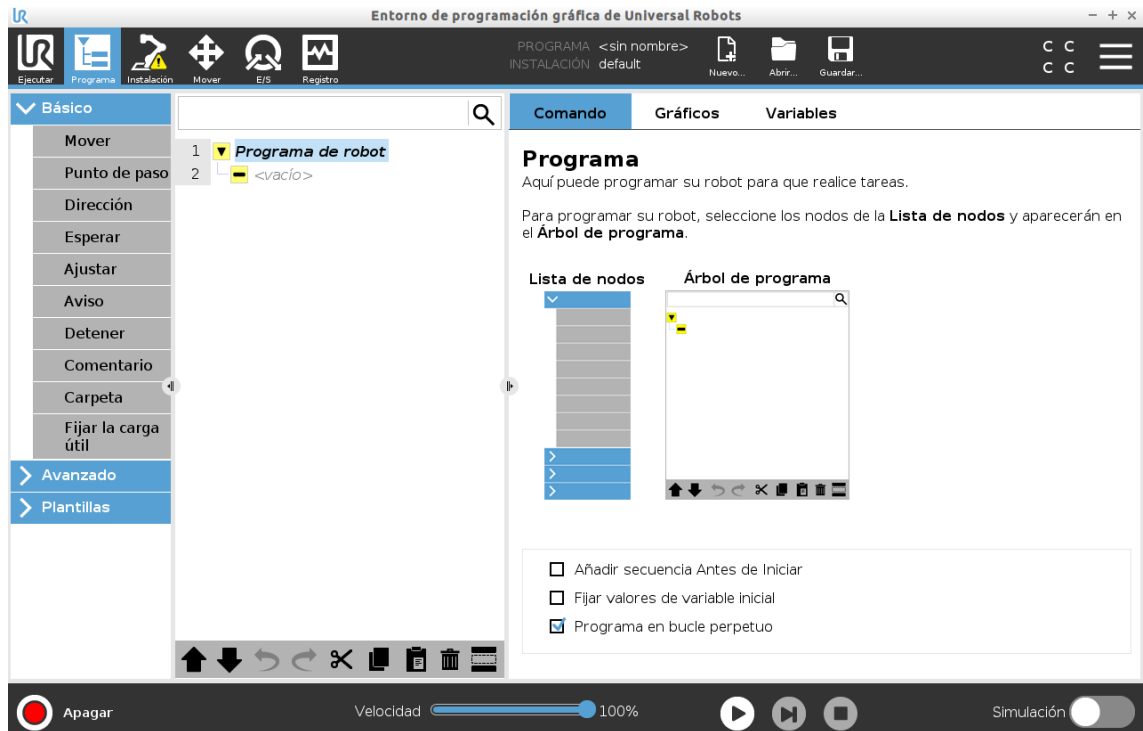


Figura 55. Pantalla inicial de "Programa".

El árbol de programa muestra el programa como una lista de comandos llamados nodos de programa. El nombre del programa aparece directamente encima de esta lista de comandos. A la derecha del árbol de programa, aparece información relacionada con el comando seleccionado. Se añaden comandos de la lista de comandos en **Básico** haciendo click en el tipo de comando deseado. Se puede acceder a comandos avanzados en **Avanzado** o puede utilizar **Plantillas**. Las plantillas ofrecen un árbol de programa ya listo para configurar.

En la parte inferior de la imagen podemos ver a su vez una serie de comandos rápidos o herramientas que facilitan la programación, como pueden ser la posibilidad de subir o bajar un comando, deshacer y rehacer, copiar, pegar, eliminar o suprimir (el equivalente a comentar en otros lenguajes de programación).



Figura 56. Detalle de la barra de herramientas.

Arriba a la derecha tenemos tres pestañas, **Comando**, que nos muestra información del comando seleccionado, **Gráficos**, donde vemos el robot simulado y podemos visualizar la ejecución de las instrucciones y **Variables**, donde vemos las variables que intervienen.

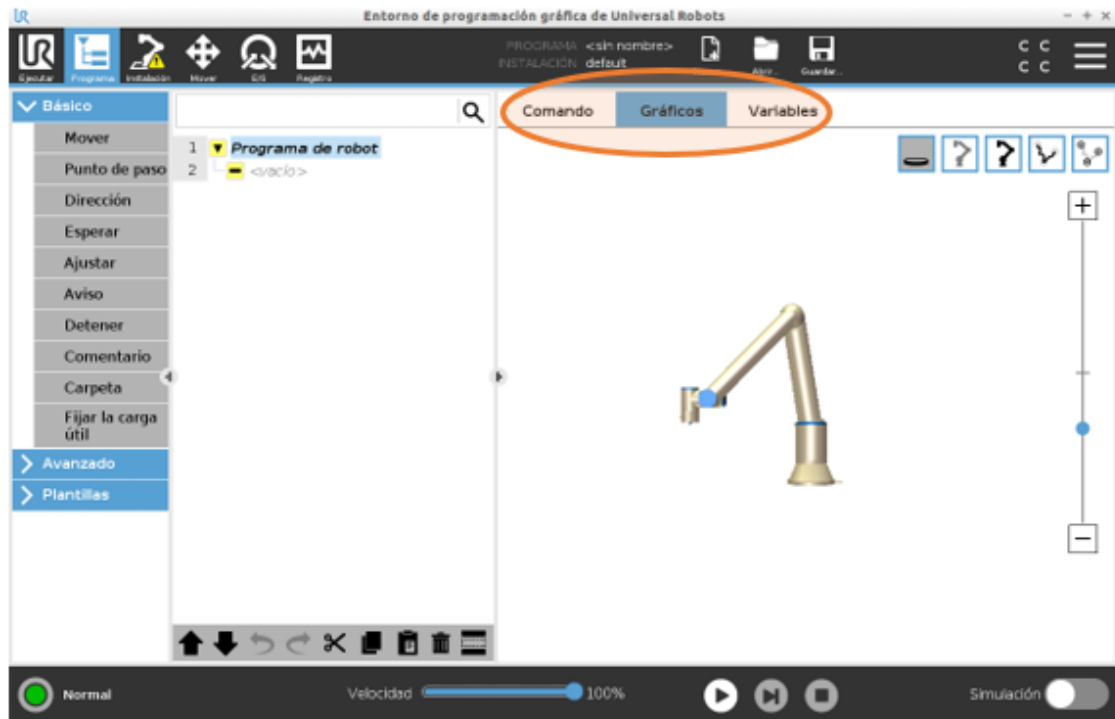


Figura 57. Detalle de las pestañas.

Dentro de Básico y como se puede ver en la figura anterior, podemos añadir un gran número de instrucciones relativamente sencillas:

- **Movimiento** permite crear un tipo de movimiento, ya sea rectilíneo (Mover L), cómodo, donde el robot realiza la trayectoria más rápida y fácil para ir de un punto a otro (Move J) o Move P, que se puede utilizar para realizar movimientos circulares. Dentro de cada comando movimiento debemos definir puntos de paso.
- Un **punto de paso** se trata de una posición en el espacio de trabajo del robot y que este alcanza con una configuración. Se introducen dentro de una instrucción Mover. Estos pueden ser fijos (con coordenadas

absolutas), relativos (indican una distancia a recorrer en una dirección desde el punto en el que se encuentra el TCP) o variables (relacionados con una variable que indica la posición).

- **Esperar** genera una espera, ya sea durante un tiempo predefinido o hasta que suceda cierto evento.
- **Ajustar** sirve para ajustar salidas digitales o analógicas para un valor dado.
- **Aviso** es un mensaje emergente que aparece en la pantalla cuando el programa llega a este comando.
- **Detener** para la ejecución del programa.
- **Comentario** da al programador la opción de añadir una línea de texto al programa. Esta línea de texto no hace nada mientras se ejecuta el programa.
- Una **Carpeta** sirve para organizar y designar partes específicas de un programa, para despejar el árbol de programa y para facilitar la lectura y navegación por el programa.

En el apartado de Avanzado nos encontramos con otra serie de instrucciones más específicas y que requieren ciertos conceptos de programación:

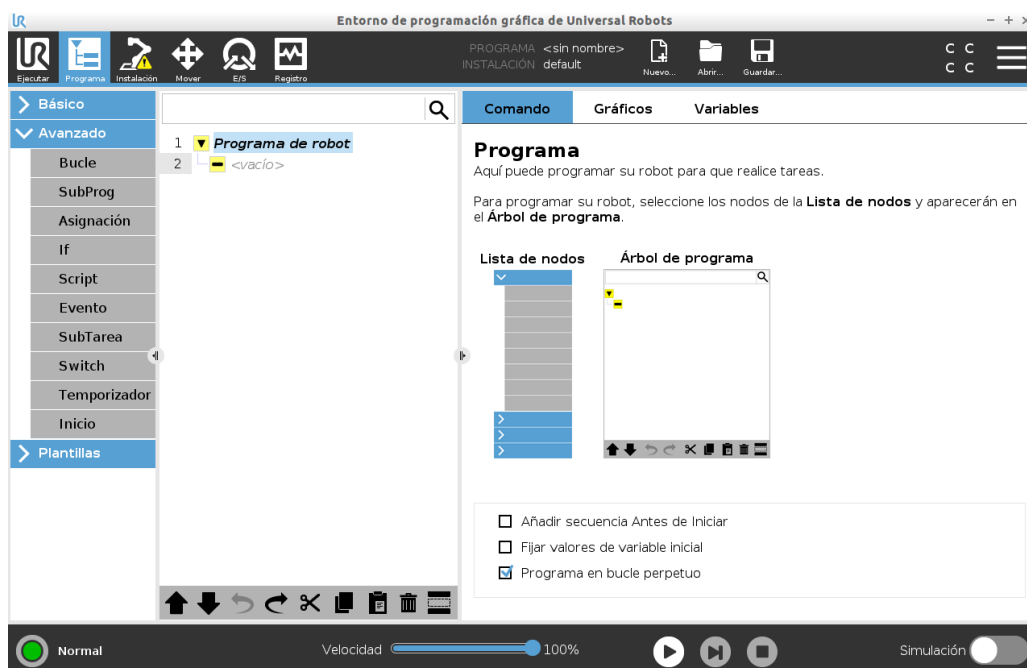


Figura 58. Comandos avanzados.

- **Bucle** repite los comandos del programa que se encuentran en su interior. Sería el equivalente al bucle 'for' de otros lenguajes de programación. En este caso este bucle se puede repetir siempre, las veces que nosotros queramos o hasta que se cumpla una condición.
- Un **Subprograma** puede albergar partes de un programa que se necesiten en varios lugares. Corresponde a las funciones de otros lenguajes de programación. Al invocar un subprograma, se ejecutarán las líneas del programa en el subprograma y luego se regresará a la línea siguiente.
- **Asignación** sirve para asignar valores a variables.
- **IF** es una estructura de comando *si... entonces*, que cambia el comportamiento del robot basándose en valores variables y entradas de sensores. Si la condición se evalúa como True (verdadera), se ejecutan las líneas incluidas en este comando IF. Un comando IF puede tener varios argumentos ElseIf que pueden añadirse o eliminarse. Sin embargo, un comando IF solo puede tener un argumento Else. Si una expresión dentro del comando IF se evalúa como False (falsa), se siguen las cláusulas de ElseIf o Else.
- **Script** da acceso al lenguaje de script subyacente que está ejecutando en tiempo real el controlador del robot. Es solo para usuarios avanzados, y puede encontrar las instrucciones para utilizarlo en el manual de scripts [8].
- Un **Evento** puede servir para supervisar una señal de entrada, realizar alguna acción o ajustar una variable cuando dicha señal de entrada se vuelva alta.
- Un **SubTarea** es un proceso paralelo al programa del robot. Puede usarse para controlar una máquina externa con independencia del brazo robótico. Un subproceso puede comunicarse con el programa del robot con variables y señales de salida.
- Un **Switch** se utiliza para evitar el uso de demasiados IF, y puede cambiar el comportamiento del robot en función de valores variables y entradas de

sensores. En él se tiene una condición y distintos casos con posibles valores de esa condición. Si al evaluar la condición coincide con uno de los casos, se ejecutan las líneas dentro del caso.

- **Temporizador** comienza a contar el tiempo desde que se ejecuta la instrucción. Este tiempo se guarda en una variable.
- **Inicio** ordena al robot volver a la posición Origen, en el caso de que se haya definido una.

Finalmente tenemos las Plantillas, donde encontramos trozos de código prediseñados especializados en la realización de tareas concretas:

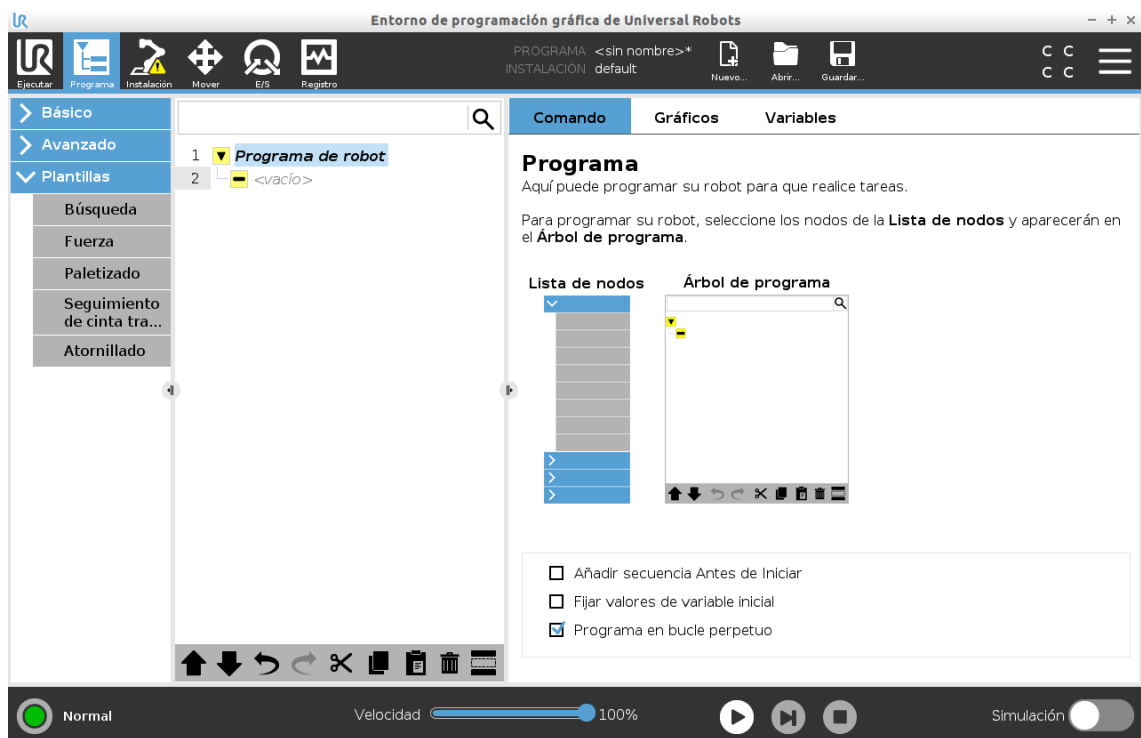


Figura 59. Plantillas.

- **Búsqueda** es una operación de apilado o desapilado, donde se indican las direcciones de inicio y final, unas coordenadas de origen y el grosor de la pieza a desplazar. Utiliza un sensor para determinar cuándo se alcanza la posición correcta para agarrar o soltar un artículo.
- El modo de **Fuerza** permite la conformidad y la fuerza en ejes seleccionables. Esto significa que podemos determinar la fuerza que hace

el brazo robótico al realizar un movimiento o sobre un objeto sobre uno o varios ejes. Esto es idóneo para aplicaciones en las que la posición real del TCP a lo largo de un eje predefinido no sea importante, pero que, en cambio, se requiera una fuerza deseada a lo largo de dicho eje.

- El **Paletizado** se utiliza a la hora de colocar objetos de forma ordenada sobre un palé o para retirarlos. Para ello, se definen las dimensiones del palé, el patrón de posicionamiento de cada uno de los objetos en él y el número de capas.
- El robot se puede configurar para realizar un **seguimiento del movimiento de la cinta transportadora**. Se permiten todos los movimientos en este nodo mientras se realiza un seguimiento de la cinta transportadora, pero están relacionados con el movimiento de la cinta.
- **Atornillado** permite mediante el uso de una herramienta específica apretar o aflojar tornillos, tras definir con precisión el punto TCP y los límites de fuerza y velocidad de atornillado.

Cabe destacar que, al ser un robot colaborativo, nos permite seleccionar puntos en el espacio de trabajo de 3 maneras distintas:

1. Ajustando las coordenadas de forma numérica manualmente. Podemos variar introduciendo valores tanto la posición de la herramienta o TCP (x , y , z y 3 valores de inclinación respecto a estos ejes) como de la posición de cada una de las juntas o articulaciones que componen al robot. Estas últimas también cuentan con la posibilidad de aumentar o decrementar el valor mediante flechas de forma progresiva, sin tener que introducir un valor numérico.

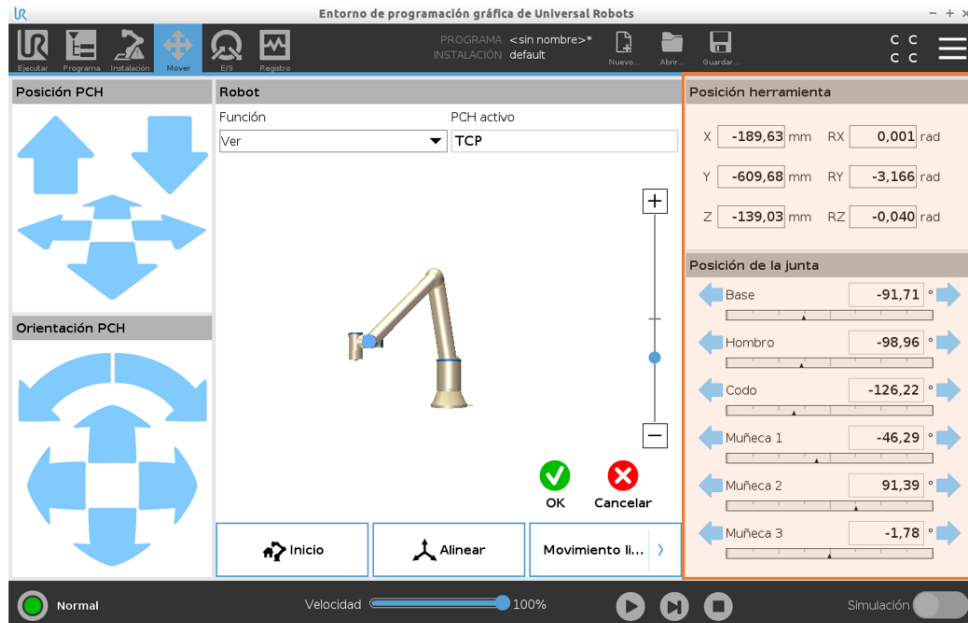


Figura 60. Control por valor numérico.

- Moviendo y controlando el TCP directamente utilizando las flechas de la izquierda en la figura inferior. Pulsando en las flechas de arriba movemos en los ejes especificados la herramienta en la dirección que queramos, mientras que con las flechas inferiores podemos modificar su orientación. Resulta más más cómodo cuando no se requiere una gran precisión.

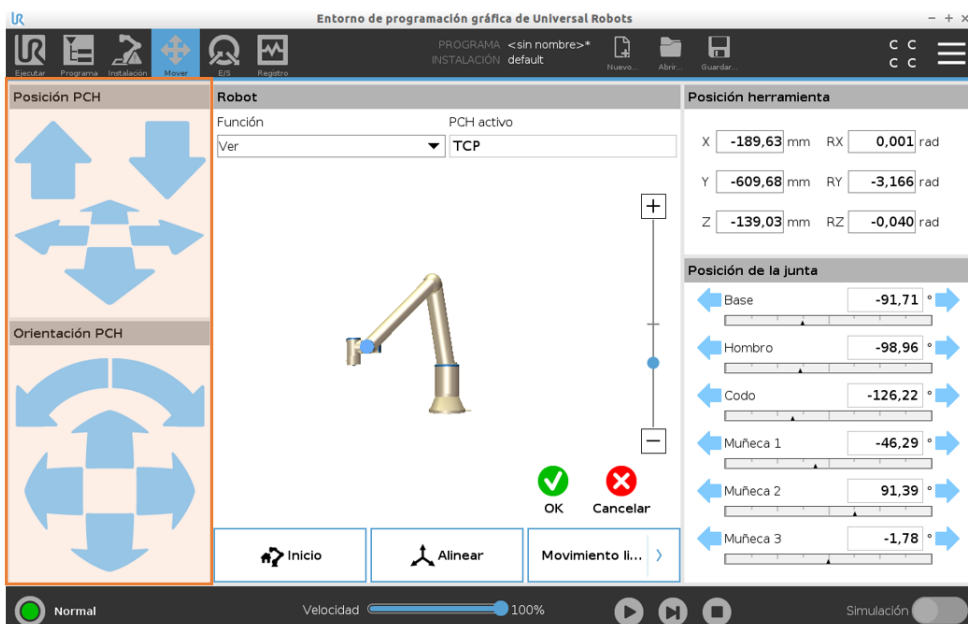


Figura 61. Control por flechas.

- Utilizando las características del robot colaborativo. Este permite mover el brazo de forma totalmente manual cuando seleccionamos la opción indicada en la figura inferior y se pulsa un botón específico en la parte trasera de la consola de programación.

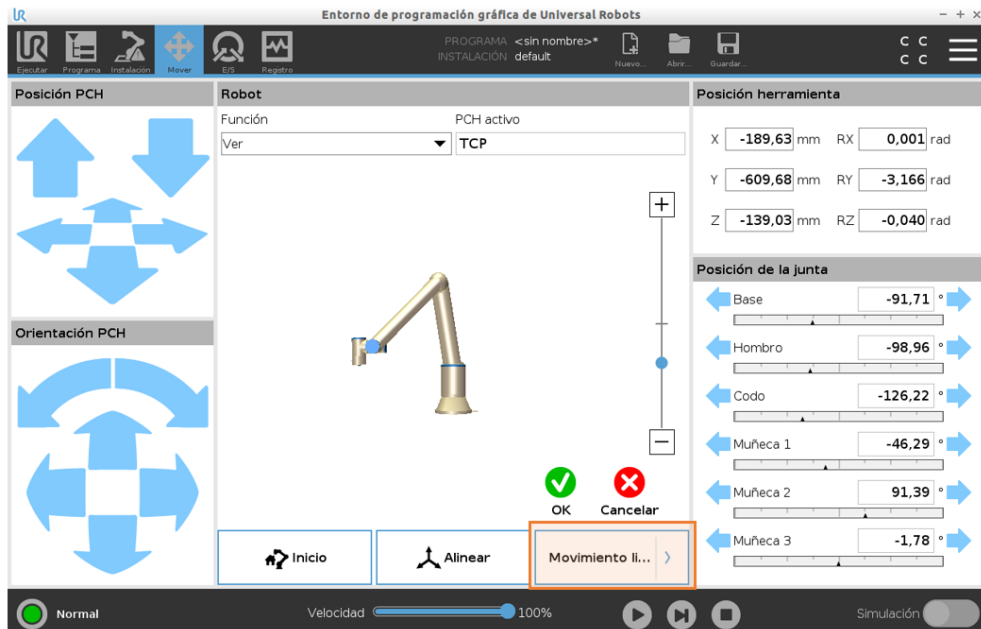


Figura 62. Control manual.

6. PLC S7 y TIA Portal

6.1. ¿Qué es un PLC?

Se trata de un equipo electrónico basado en un microprocesador que utiliza instrucciones almacenadas en memoria programable para implementar la lógica, secuenciación, temporización, conteo y funciones aritméticas mediante módulos analógicos o digitales, y está diseñado para controlar en tiempo real y en ambiente industrial procesos de evolución secuencial [9].

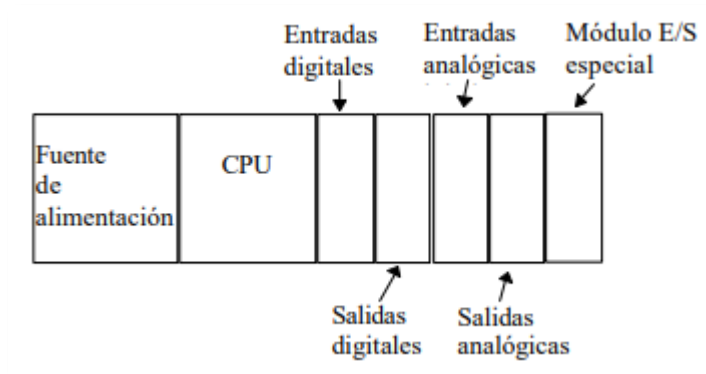


Figura 63. Constitución física en un PLC modular.

Su arquitectura funcional tiene un esquema como el siguiente:

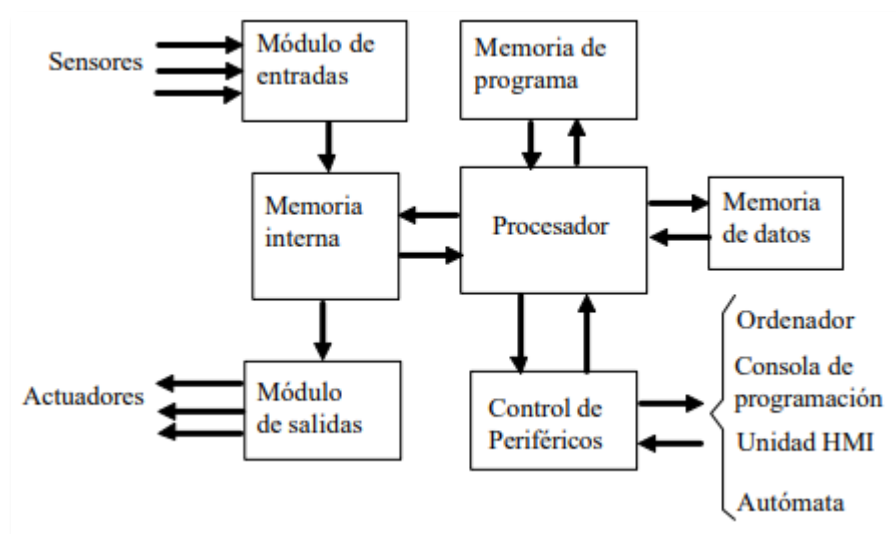


Figura 64. Arquitectura funcional de un PLC.

Si analizamos la memoria del autómata, encontramos 3 tipos distintos:

- **Memoria de programa.** Contiene el programa (instrucciones) que se ejecutan en el procesador. Se puede dividir en una parte ROM que contiene el programa monitor que es fijo y una parte RAM en la que se almacena el programa del usuario. Este programa se mantiene aunque se desconecte el autómata.
- **Memoria interna.** Almacena los valores de entradas y salidas, además de otras variables internas del autómata. En ella se almacenan variables de 1 solo.
- **Memoria de datos.** Contiene datos de configuración o parámetros de funcionamiento del autómata y del proceso, o datos de propósito general. En ella se almacenan variables tipo byte (8 bits) o word (16 bits).

El PLC es un computador y, por lo tanto, su funcionamiento consiste en la ejecución de un determinado programa situado en la memoria de programa. Existen dos modos de funcionamiento: modo de programación, y modo de ejecución.

- **Modo de programación.** En este modo de funcionamiento el programa monitor comunica el PLC con el elemento de programación (normalmente un PC), para que desde este se le envíe el programa de control que se desea ejecutar.
- **Modo de ejecución (modo RUN).** En este modo se ejecuta el programa de control del usuario, por lo que el autómata controla el proceso. Cuando se inicializa el Autómata, el programa monitor salta a la dirección donde está el programa de control para que se ejecute. La ejecución del programa de control se lleva a cabo de forma cíclica, ejecutando ciclos de 'scan' de forma indefinida.

Se llama ciclo de trabajo (o de "scan") al conjunto de tareas que el autómeta lleva a cabo de forma cíclica cuando está controlando un proceso. El tiempo de 'scan' es el tiempo que tarda el autómeta en realizar un ciclo completo, y depende de la velocidad de la CPU. El ciclo de trabajo más habitual tiene la siguiente estructura:

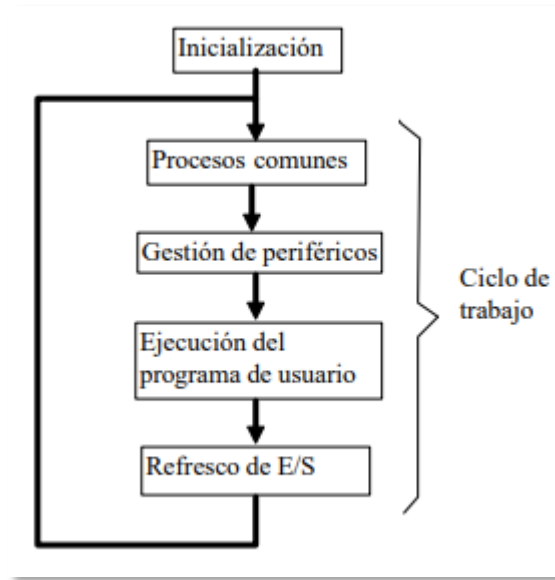


Figura 65. Esquema del ciclo de trabajo.

- La inicialización se lleva a cabo una sola vez cada vez al poner en marcha el autómeta.
- Los procesos comunes son funciones de autodiagnóstico y de programación del Watch Dog.
- La gestión de periféricos gestiona los puertos de comunicaciones (comunicación con un ordenador, con un terminal HMI, con otro autómeta, etc.).
- El programa de usuario se ejecuta instrucción a instrucción, de forma secuencial. Las instrucciones (líneas) del programa de usuario son las que implementan la lógica de control del proceso. Todas estas instrucciones se ejecutan utilizando los valores almacenados en la memoria interna de

entradas, y dan como resultado un cambio en la memoria interna de salidas.

- El refresco de E/S significa la actualización de la memoria de entradas con los valores actuales de las entradas (de los sensores), y la actualización de las salidas (actuadores) con los valores almacenados en la memoria de salidas.

6.2. PLC S7 1214C DC/DC/DC

La CPU compacta SIMATIC S7-1200 1214C DC/DC/DC - 6ES7214-1AG40-0XB0 de Siemens se trata de un PLC de alto rendimiento que dispone de 14 entradas digitales, 10 salidas digitales y 2 entradas analógicas [4].

La tensión de alimentación es de 20,4 a 28,8V DC. Dispone de un puerto PROFINET para la comunicación y programación, y gracias a sus 100 kB de memoria para guardado de datos y programa lo convierte en una solución óptima en todas las industrias en múltiples tareas de control.



Figura 66. CPU 1214C.

Algunas de las características más interesantes de cara a la programación que proporciona este dispositivo son:

Tiempos de ejecución de la CPU	
para operaciones de bits, típ.	0,08 µs; /instrucción
para operaciones a palabras, típ.	1,7 µs; /instrucción
para aritmética de coma flotante, típ.	2,3 µs; /instrucción
CPU-bloques	
Nº de bloques (total)	DBs, FCs, FBs, contadore y temporizadores. El número máximo de bloques direccionables es de 1 a 65535. No hay ninguna restricción, uso de toda la memoria de trabajo
OB	
• Número, máx.	Limitada únicamente por la memoria de trabajo para código
Áreas de datos y su remanencia	
Área de datos remanentes (incl. temporizadores, contadores, marcas), máx.	14 kbyte
Marcas	
• Tamaño, máx.	8 kbyte; Tamaño del área de marcas
Datos locales	
• por cada prioridad, máx.	16 kbyte; Clase de prioridad 1 (ciclo de programa): 16 kbyte, clase de prioridad 2 a 26: 6 kbytes
Área de direcciones	
Imagen del proceso	
• Entradas, configurables	1 kbyte
• Salidas, configurables	1 kbyte
Configuración del hardware	
Nº de módulos por sistema, máx.	3 Communication Module, 1 Signal Board, 8 Signal Module

A su vez, la CPU llevaba incorporado un módulo de entradas y salidas digitales Siemens SM 1223 DC/DC [5].



Figura 67. Módulo de entradas y salidas digitales SM 1223 DC/DC.

El módulo tiene las siguientes características técnicas:

- Módulo de 8 entradas digitales 24 VDC con 8 salidas.
- Tensión de alimentación entre 20,4 V y 28,8 V.
- Intensidad de salida 0,5 A por canal.

6.3. El PLC en la instalación

El PLC se encuentra ubicado en el armario de control principal, en una de las vallas laterales de la instalación.

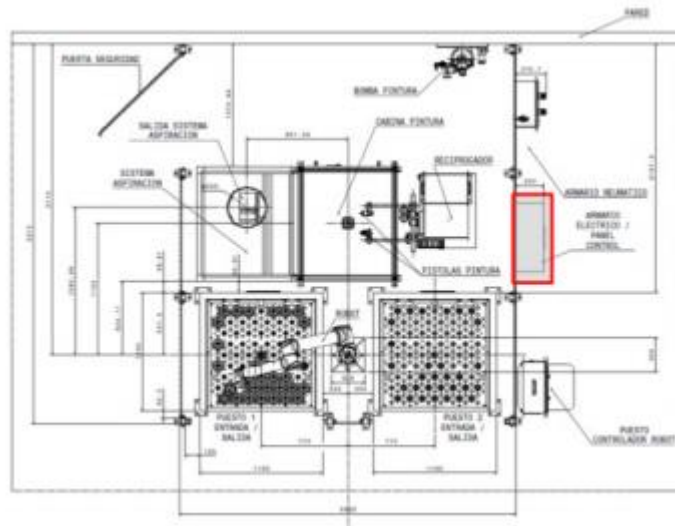


Figura 68. El armario de control en la instalación.

En el frontal del cuadro existe un panel HMI KTP700 Basic que ofrece al operario las herramientas necesarias para el control de la instalación y la visualización del estado de todos los elementos de la misma [6].



Figura 69. Armario de control.

La aplicación HMI diseñada para la Supervisión y Control de la Instalación de Pintura de Aislantes Eléctricos, implementada en el Panel de Control del armario, presenta las funcionalidades necesarias para el control de todos los elementos existentes en la instalación. La interfaz está basada en un conjunto de pantallas de operación y visualización, donde el operador tiene la posibilidad de parametrizar y configurar todos los elementos de la instalación, efectuar el control de todos los elementos en modo manual y supervisar el funcionamiento en modo automático.



Figura 70. Distribución de la interfaz HMI.

En el interior del armario encontramos un complejo entramado de elementos y conexiones con el PLC como eje principal y cerebro de toda la instalación.

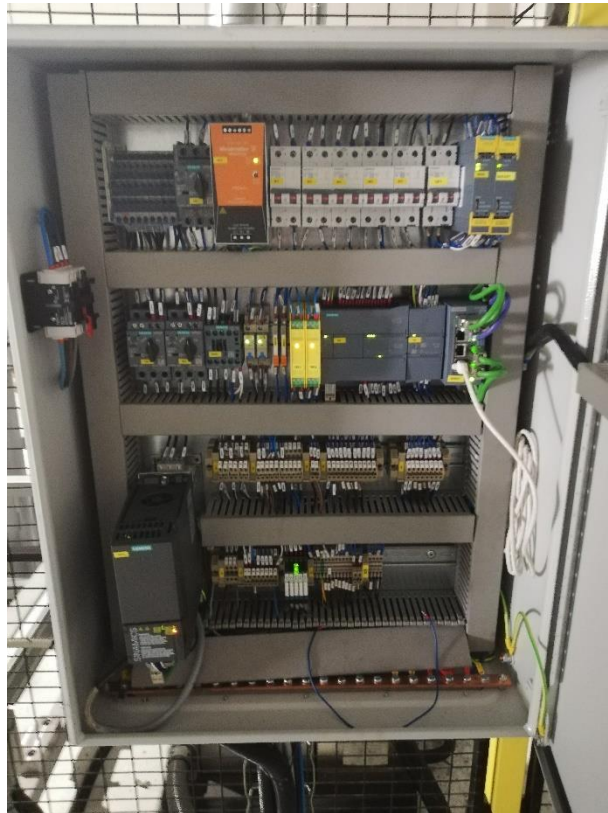


Figura 71. Interior del armario de control.

Si analizamos el proceso de pintura de piezas cilíndricas llevado a cabo en la instalación, podemos enumerar cada una de las funciones que lleva a cabo el autómeta:

- En primer lugar, controla toda la seguridad de la instalación, revisando mediante sensores el estado de válvulas, puertas y zona de trabajo para que la estación de pintura desarrolle su actividad correctamente. Este apartado está directamente relacionado con las alarmas que se generan y saltan en la pantalla HMI cuando se detecta un problema. Las más críticas son el disparo del térmico del motor de giro, el disparo del térmico de la cabina de aspiración, la alarma de presión general de aire o la intrusión en la cabina de pintura.

- Contiene todo el código del programa salvo el específico para el robot UR10e, aunque se comunica con él y lo controla en una relación similar a maestro-esclavo. El resto de elementos sí que son totalmente controlados por el autómatas (motor de giro, pistolas de pintura, reciprocador...), que se supervisan y se activan según se lo indica.
- Antes de iniciar el programa el operario debe configurar todos los parámetros necesarios, como la referencia de la pieza a pintar, los valores de presión y otros valores referentes a las pistolas de pintura.
- Para la ejecución del programa, el PLC coordina todos los elementos para que cada uno realice la acción que tiene asignada en el momento apropiado, según se ha reflejado en el código.

6.4. TIA Portal y el lenguaje de programación

TIA portal es el software de SIEMENS para la programación de PLCs. La principal ventaja que ofrece es la posibilidad de integrar distintas aplicaciones de software industrial para procesos de producción en una misma interfaz lo que facilita enormemente el aprendizaje, la interconexión y la operación.

Para realizar un programa primero debemos crear un proyecto o abrir uno ya existente para modificarlo.



Figura 72. Pantalla inicial de TIA Portal.

Si creamos un proyecto desde cero, primero debemos agregar el controlador con el que vamos a trabajar. TIA Portal recoge todos los modelos de SIEMENS disponibles. También permite añadir otros elementos, como pantallas HMI, dispositivos de entradas y salidas o incluso otro PLC.

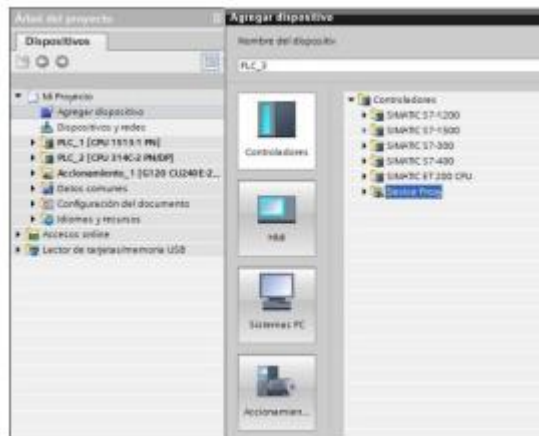


Figura 73. Pestaña para añadir elementos al proyecto.

A continuación, pasamos a la pestaña de programación, donde nos encontramos segmentos que contienen contactos y bobinas, como podemos ver en la figura inferior.

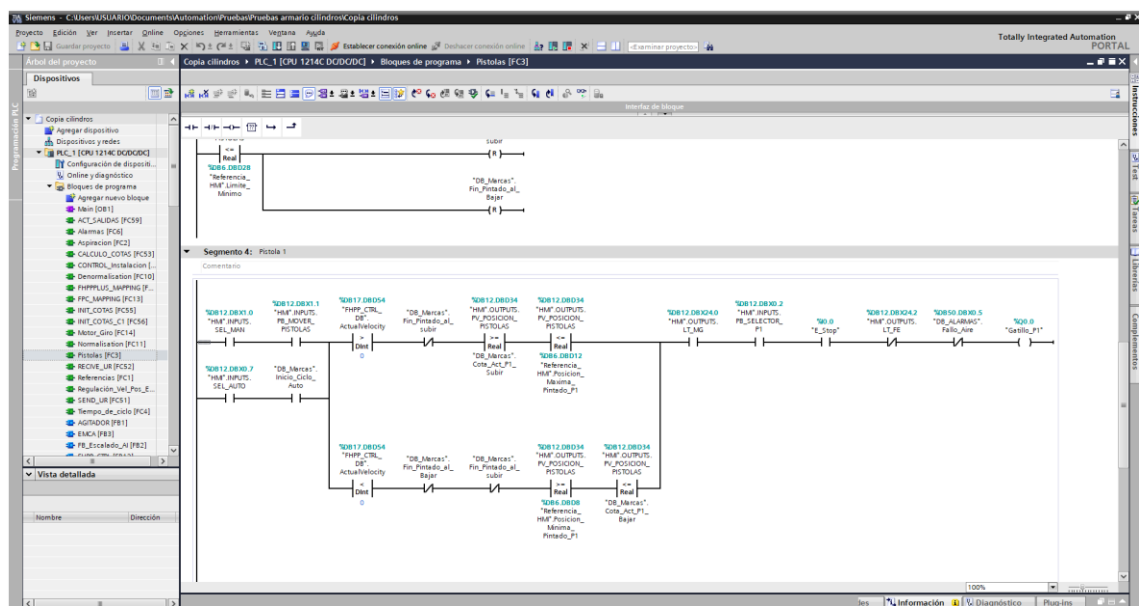


Figura 74. Captura de TIA Portal.

El lenguaje de programación más utilizado en los autómatas programables es el diagrama de contactos (*Ladder Diagram* o diagrama de escalera). Está basado en los automatismos cableados por medio de contactores, los primeros en usarse. Representa las ecuaciones lógicas por medio de contactos, existiendo 3 tipos de elementos fundamentales [9].

- **Contacto normalmente abierto (NA).** Representa un contacto que está abierto si la variable asociada vale 0, y que se cierra si la variable asociada vale 1.
- **Contacto normalmente cerrado (NC).** Representa un contacto que está cerrado si la variable asociada vale 0, y que se abre si la variable asociada vale 1.
- **Bobina.** Representada por un círculo o dos paréntesis, simboliza el valor de una variable. Cada salida tiene asociada una bobina. Si esta bobina tiene corriente (el diagrama de contactos la activa), la salida está a 1, y si no tiene corriente la salida está a cero. Además puede haber bobinas asociadas a variables internas (por ejemplo las que representan cada etapa del proceso).

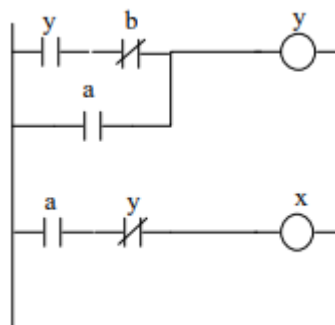


Figura 75. Ejemplo de diagrama de contactos.

El autómata ejecuta las líneas de arriba abajo. En este caso, calcularía $y \cdot \bar{b} + a$ y lo asignaría a y . Después calcularía $a \cdot \bar{y}$ y lo asignaría a x . Esta sería la fase de ejecución del programa de usuario, que se repetiría de forma cíclica según se ha visto.

La función lógica implementada por los contactos de un segmento se suele denominar condición de activación, ya que cuando esta condición es 1 la bobina de salida asociada se activa.

Hay una serie de normas que se deben seguir para realizar correctamente un diagrama de contactos.

- A la derecha de la bobina no puede haber contactos.
- En un segmento puede haber varias bobinas en paralelo.
- La corriente a través de los contactos solo circula de izquierda a derecha.

Además de la bobina simple puede haber otros 2 tipos de bobinas:

- **Bobina normalmente cerrada.** Se representa con un círculo o paréntesis con una barra transversal.
- **Bobina de enclavamiento.** En algunos autómatas se representa mediante un círculo o paréntesis que en el interior tiene una letra "S" (Set) para una bobina que se mantiene activa al hacerla conducir y tras poner a cero la condición de activación, o una letra "R" (Reset) para desactivar la variable anterior.

Esos elementos básicos solo permiten programar relaciones lógicas simples y, evidentemente, esto no es suficiente para controlar un proceso complejo.

Para ello contamos con un gran número de instrucciones y elementos para recrear y completar las funciones lógicas que necesitemos programar. Entre las más destacadas se encuentran:

- Instrucción END. Final del programa.
- Temporizadores.
- Instrucciones de salto.
- Contadores.
- Operaciones binarias (sumar, multiplicar, comparar...).

- Refresco de E/S. Permite refrescar las E/S cuando se desee, independientemente del refresco automático que se produce cada ciclo de scan.
- Funciones matemáticas complejas (senos, cosenos, exponenciales).
- Funciones especiales de comunicaciones (transmisión o recepción de datos).
- Control de interrupciones.

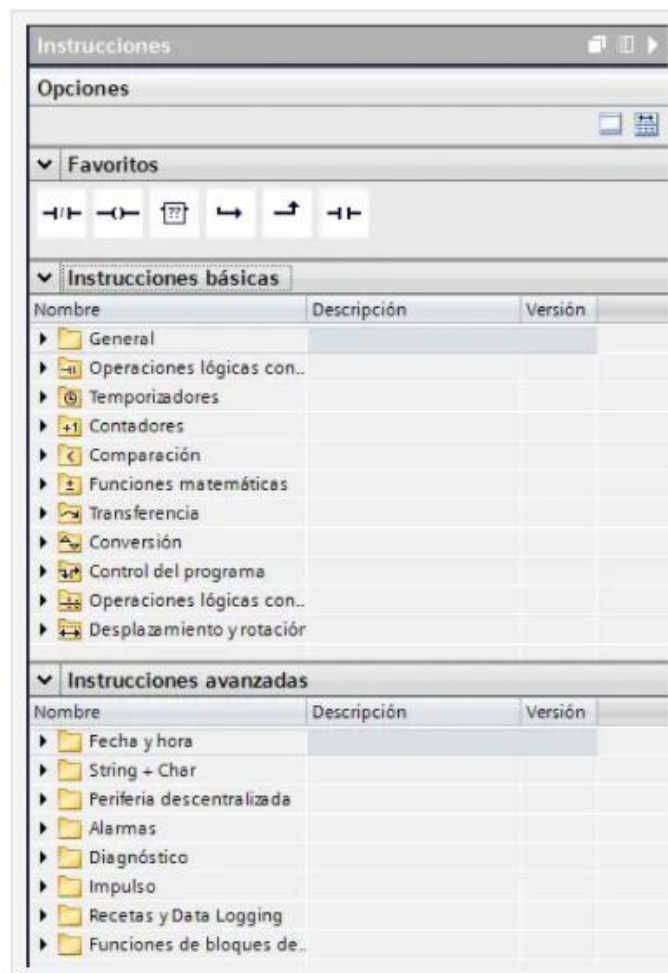
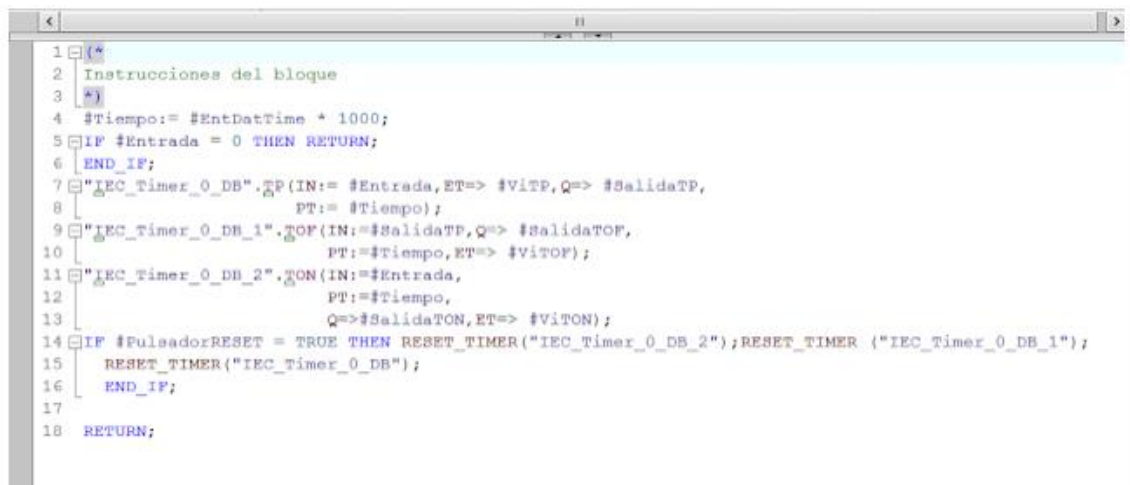


Figura 76. Instrucciones de programación en TIA Portal.

Existe también otro lenguaje de programación llamado SCL, similar a lenguajes de alto nivel como pueden ser C++ o Python. Es un lenguaje de programación de alto nivel orientado a Pascal que posibilita una programación estructurada La

programación se realiza mediante líneas de código escrito, donde se pueden definir variables, asignar valores o ejecutar acciones concretas [10].



```
1 (*
2 Instrucciones del bloque
3 *)
4 #Tiempo:= #EntDatTime * 1000;
5 IF #Entrada = 0 THEN RETURN;
6 END_IF;
7 "IEC_Timer_0_DB".TP(IN:= #Entrada,ET=> #ViTP,Q=> #SalidaTP,
8     PT:= #Tiempo);
9 "IEC_Timer_0_DB_1".TOP(IN:=#SalidaTP,Q=> #SalidaTOP,
10     PT:=#Tiempo,ET=> #ViTOP);
11 "IEC_Timer_0_DB_2".TON(IN:=#Entrada,
12     PT:=#Tiempo,
13     Q=>#SalidaTON,ET=> #ViTON);
14 IF #PulsadorRESET = TRUE THEN RESET_TIMER("IEC_Timer_0_DB_2");RESET_TIMER ("IEC_Timer_0_DB_1");
15     RESET_TIMER("IEC_Timer_0_DB");
16 END_IF;
17
18 RETURN;
```

Figura 77. Ejemplo programación en SCL.

Un uso típico sería el guardado y almacenamiento de datos en zonas de memoria, ya sea en un array de datos o en una matriz. Con sus funcionalidades de lectura cíclica, este lenguaje estructurado nos permite operar repetitivamente con estos datos un simple código de operación.

También es posible combinar este lenguaje de programación con los diagramas de contactos, pudiendo insertar segmentos SCL en bloques de Ladder.

Programar en TIA Portal tiene la peculiaridad de que te permite separa la totalidad del programa en bloques del programa de usuario [11]. Estos bloques pueden ser:

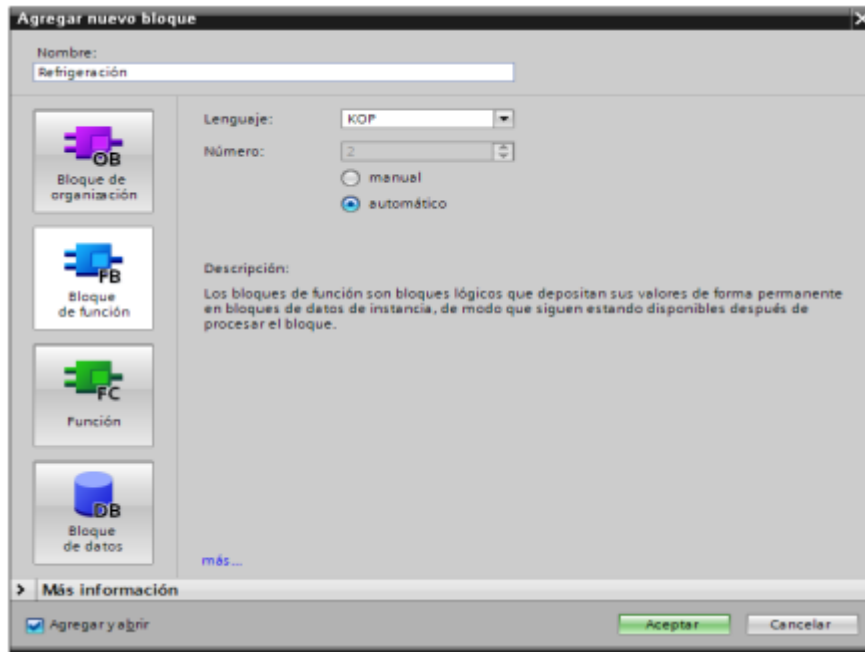
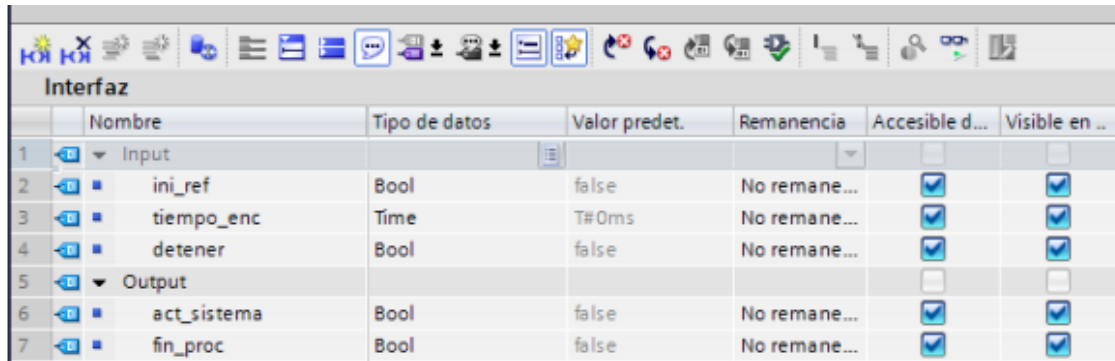


Figura 78. Tipos de bloques de programa.

- **Bloque de organización (OB).** Se trata del bloque principal donde se ejecutan el resto de funciones. Sería el equivalente al 'main' de lenguajes como C++.
- **Bloque de función (FB).** Los bloques de función son bloques programables con memoria, por lo que pueden almacenar valores en un bloque de datos asignado (DB). Los datos memorizados en el DB de instancia no se pierden al concluir el tratamiento del FB.
- **Funciones (FC).** Las funciones son bloques programables sin memoria, por lo que las variables locales (temporales) y sus valores se almacenan en la pila de datos locales. En el caso de memorizar datos de forma permanente, se recurre a bloques de datos globales. La FC contiene un programa que se ejecuta cada vez que la FC es llamada por otro bloque lógico, y se pueden utilizar para devolver un valor o para ejecutar una función tecnológica. Si esta función necesita valores externos, estos se añaden como entradas al bloque y se asignan a variables locales a la función.
- **Bloque de datos (DB).** Las variables declaradas en los bloques de funciones se almacenan en su respectivo bloque de datos de instancia.



	Nombre	Tipo de datos	Valor predet.	Remanencia	Accesible d...	Visible en ..
1	Input					
2	ini_ref	Bool	false	No remane...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
3	tiempo_enc	Time	T#0ms	No remane...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
4	detener	Bool	false	No remane...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
5	Output					
6	act_sistema	Bool	false	No remane...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
7	fin_proc	Bool	false	No remane...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Figura 79. Variables almacenadas en un bloque de datos.

Una vez se tiene el programa esquematizado, se debe proceder siguiendo los siguientes pasos para introducir el programa en el autómata:

1. Configurar el equipo (tipo de PLC, comunicaciones...).
2. Identificar el programa (nombre, fecha, versión, etc.)
3. Crear la estructura básica del programa (bloques y grupos dentro de los bloques).
4. Hacer la lista de nombres de las variables a utilizar (entradas, salidas, variables internas, datos, temporizadores, contadores), asignándoles su dirección en memoria.
5. Introducir el programa (diagrama de contactos).
6. Comprobar que el programa no da errores y depurarlo para mejorar su funcionamiento.
7. Transferir el programa al autómata.
8. Ejecutar en modo monitor y comprobar el correcto funcionamiento.

7. Desarrollo y soluciones del proyecto

7.1. Inicio del proyecto en la empresa en el periodo de prácticas

Una vez planteado el proyecto y los obstáculos a sortear, se procede a realizar, de forma resumida, el desarrollo del proyecto llevado a cabo durante las prácticas de empresa en orden cronológico. Esta parte pertenece al trabajo realizado durante el periodo de prácticas casi en su totalidad, debido a que se trata del proyecto que realicé con esta empresa utilizando sus instalaciones.

Inicialmente se estudió con el ordenador de escritorio el manual del robot. Como se ha indicado, se trata de un robot colaborativo UR10e, del fabricante Universal Robots.



Figura 80. Robot UR10e en la estación de pintura.

Tras conocer los detalles más importantes a nivel de seguridad y funcionamiento, se instaló el software gratuito URSim de la página de Universal Robots para poder simular la “Teach Pendant” (desde donde se controla y programa el robot real) y así aprender el lenguaje de programación y realizar los primeros programas.

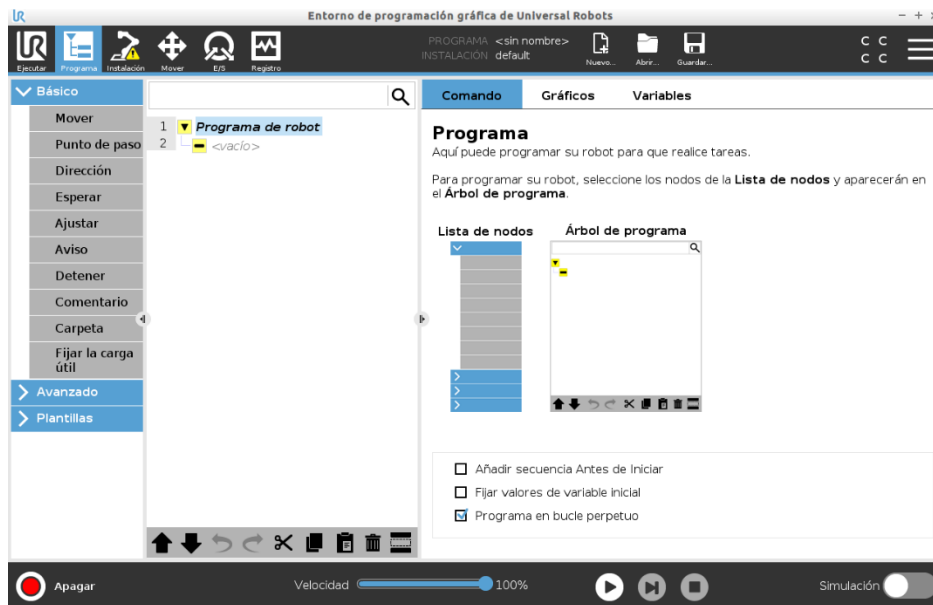


Figura 81. Captura del programa URSim.

Con la ayuda de los vídeos que proporciona la “UR Academy” y de diversos tutoriales en internet, se crearon distintas pruebas que conlucieron a posibles soluciones para pintar las piezas complejas [12].

Las primeras pruebas consisten en realizar un ‘pick and place’ sencillo para coger las piezas del carro, que posteriormente se sustituyó por la tarea de paletizado, la cual existe como plantilla predefinida en el entorno de programación.

7.2. Primeras soluciones

Posteriormente se planteó cómo llevar a cabo la pintura de las piezas con forma de ‘T’. La primera aproximación consistía en dividir la pieza en 5 zonas, que se pintarían en función del orificio de agarre por parte del robot. De esta forma, el robot necesitaría 3 agarres distintos para pintar la pieza por completo, lo que supone mucho tiempo perdido en el cambio de agarre, un número considerable

de instrucciones a programas y un aumento de la posibilidad de fallo al tener que coger la pieza 3 veces. Finalmente, se mejoró hasta necesitar un único cambio de agarre.

Para ello se diseñó un tipo de carro con un soporte específico para realizar esos cambios de agarre, adecuando a su vez las máscaras y los soportes para que el robot pudiera coger con la misma pinza por todos los orificios y que siempre estuvieran en la misma orientación en el carro de transporte.

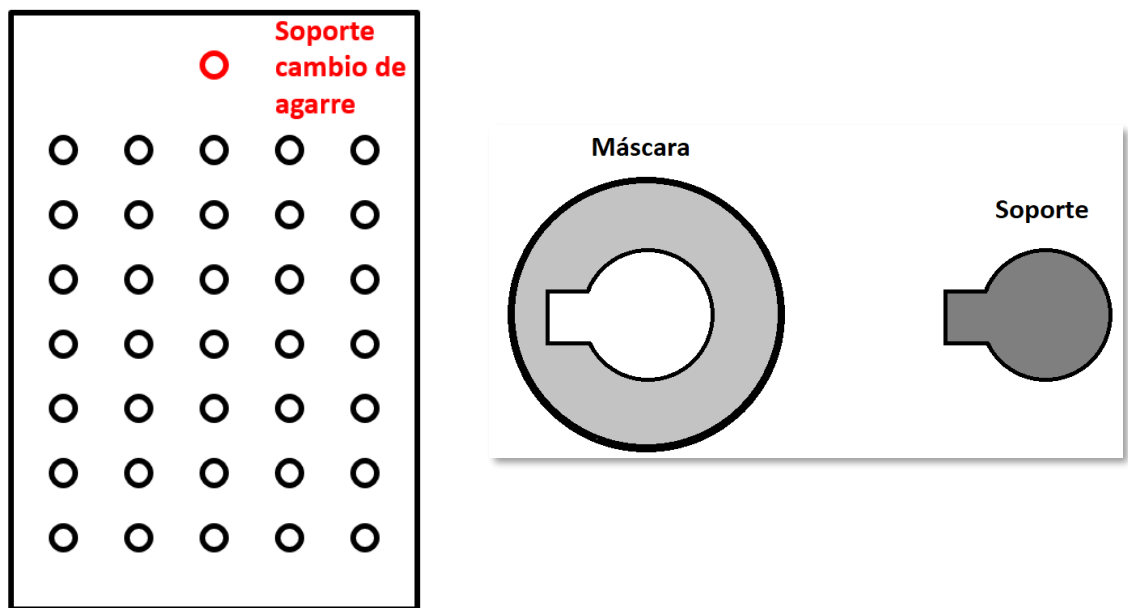


Figura 82. Carro con soporte de cambio de agarre y máscara y soporte modificados.

Otro problema a tener en cuenta es el hecho de que la muñeca 3 del robot, que sería la encargada de girar la pieza, tiene un rango de giro máximo de dos rotaciones completas (720°). Este impedimento descartó la posibilidad de rotar de forma ilimitada la pieza frente a las pistolas, teniendo que realizar una pequeña parada y el cambio de sentido en la rotación para para seguir trabajando en la pieza.

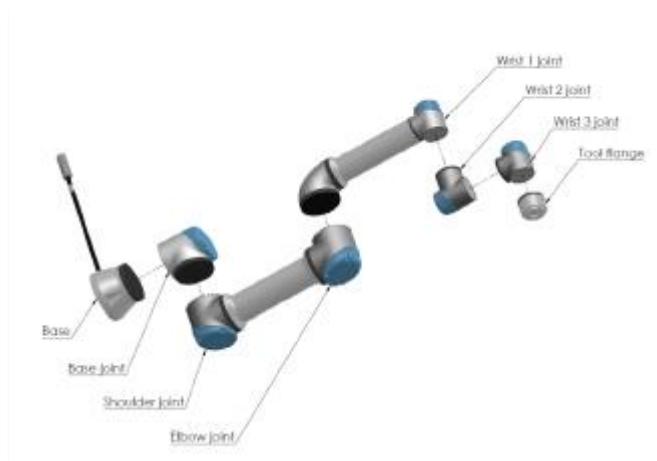


Figura 83. Las juntas, la base y la brida de herramienta del brazo robótico.

Otra solución planteada consistía en vez de realizar rotaciones con la pieza mientras se pinta, pintar realizando pasadas rectas e ir rotando para pintar las zonas que quedan por completar. En esta opción también se requería un cambio de agarre, pero se evitaba el problema de los límites de rotación de la muñeca 3.

Una tercera posibilidad era aprovechar el soporte rotativo del interior de la cabina de pintura que utilizan para pintar las piezas cilíndricas. De esta forma, se depositaba la pieza por uno de los orificios, se retiraba el robot y se activaban el motor del soporte y las pistolas, como ocurre en el proceso de las piezas cilíndricas. Una vez ha finalizado, se realiza un cambio de agarre con el robot y se coloca en el soporte de tal forma que se pinte correctamente el “brazo” de la ‘T’ restante.



Figura 84. Pieza en el soporte rotativo.

Los principales problemas de esta solución eran el hecho de que la rotación sería más lenta e inestable debido a la geometría de la pieza y la homogeneidad final de la pieza y la pintura de las zonas de difícil acceso como son las uniones entre ambos brazos.

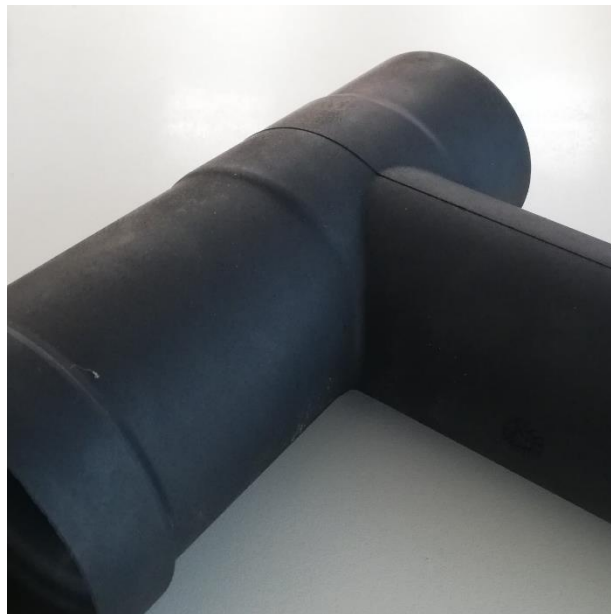


Figura 85. Uniones de la 'T'.

Las grandes dudas que generaban estas tres soluciones planteadas eran las siguientes:

- Los cambios de agarre ralentizarían considerablemente el proceso de pintura, sin olvidarnos del resto de problemas anteriormente mencionados que esta acción puede acarrear.
- La rotación limitada de la muñeca 3 y su velocidad reducida de giro en la primera solución propuesta.
- La protección necesaria del robot en las dos primeras soluciones, debido a que ahora este se introduce en la cabina de pintura mientras se están pintando las piezas. Esto obliga a la adquisición de una funda protectora para evitar que se Introduzca pintura en el brazo robótico.
- Los problemas de estabilidad y control durante la pintura de la pieza si se utiliza el soporte rotativo de la cabina en la tercera solución.
- Y por último, y siendo el principal reto del proyecto, si se puede garantizar la homogeneidad en todas las piezas, teniendo especialmente en cuenta las uniones mostradas en la *Figura 85*.

7.3. Obtención y análisis del software de la estación

Poco después se obtuvo la contraseña por parte de la empresa encargada de realizar la instalación para acceder al robot y al programa actual para pintar piezas cilíndricas. Esto permitía poder introducir los programas personales y probarlos en el robot real, viendo cómo se comportaba el brazo, la precisión de este y el posicionamiento respecto al resto de la instalación.

Así se estudió y analizó el programa de las piezas cilíndricas para tener una referencia para nuevos programas y ver cómo es la comunicación con el PLC.

Una vez analizado el programa, se descubrieron más posibilidades e instrucciones del robot y la orientación que tiene respecto a la estación. A partir de esto, se plantearon 3 listas para organizar el trabajo:

1. Contenido e instrucciones del programa que se puede copiar o reutilizar en el nuevo programa, como por ejemplo la función para iniciar el programa o la estructura para la introducción de cotas y otros datos de las piezas.
2. Registros (bool, integer o float) que se envían entre el robot y el PLC. Estos registros son los encargados de la comunicación entre estos dos elementos, enviando información relevante sobre el estado de ejecución del programa, la posición del robot, la detección de las piezas por parte del sensor o datos respecto a la referencia, el número de pieza que debe coger el robot o las medidas de las mismas.
3. Dudas para la empresa que realizó la instalación. Estas son relativas al funcionamiento de la estación y la interacción entre el robot y el PLC.

Tras elaborar estas listas, se procedió a adaptar el programa de las piezas cilíndricas al nuevo, copiando partes del programa y viendo qué modificar. Una cuestión interesante era la posibilidad de utilizar un programa que permitiera pintar todas las piezas nuevas ('T' y 'X') o por el contrario un programa para cada caso, teniendo en cuenta la programación que conlleva cada opción. Debido a que principalmente el esfuerzo se centró en la solución para las piezas con forma de 'T', finalmente se optó por intentar integrar todos los modelos con esta forma en un único programa para facilitar la tarea del operario responsable de la pintura. Esto se llevaría a cabo utilizando un programa con partes comunes como la inicialización del programa o el cambio de pinza, pero que utilizaría partes únicas para cada tipo de pieza. Las partes se seleccionarían en función de la referencia asignada a cada pieza mediante un 'IF', donde se ajustarían los valores de posición o tiempo de ejecución en función de las dimensiones de las piezas y su colocación en los carros.

Más adelante se valoraría el caso de las piezas con forma de 'X', cuya solución más sencilla sería implementarlas en otro programa distinto.

De forma paralela se inició el análisis del programa del PLC con el que se controla la estación de pintura al completo. Para ello, se instaló el software TIA Portal y se buscó información, manuales y tutoriales acerca de la programación en TIA Portal, utilizando contactos, viendo cómo funcionan las entradas y salidas, los bloques de programa y realizando simulaciones con PLCSIM [13], el software de SIEMENS para simular el funcionamiento de nuestros programas en el autómatas real.

Tras obtener el programa, se analizó en profundidad bloque por bloque hasta comprender parcialmente cómo funcionaba, cuál era el orden de funcionamiento y para qué servía cada uno de los bloques de programa. En su interior se pudo comprobar cómo estaban configuradas las salidas y entradas con el robot y comparar con la tabla que se había elaborado días atrás de los registros del programa del UR. A su vez, se encontraron unos bloques de programa escritos en lenguaje SCL desconocidos hasta el momento, descubriendo más tarde que se trataban de unos bloques propios del reciprocador Festo encargado de subir y bajar las pistolas [14].

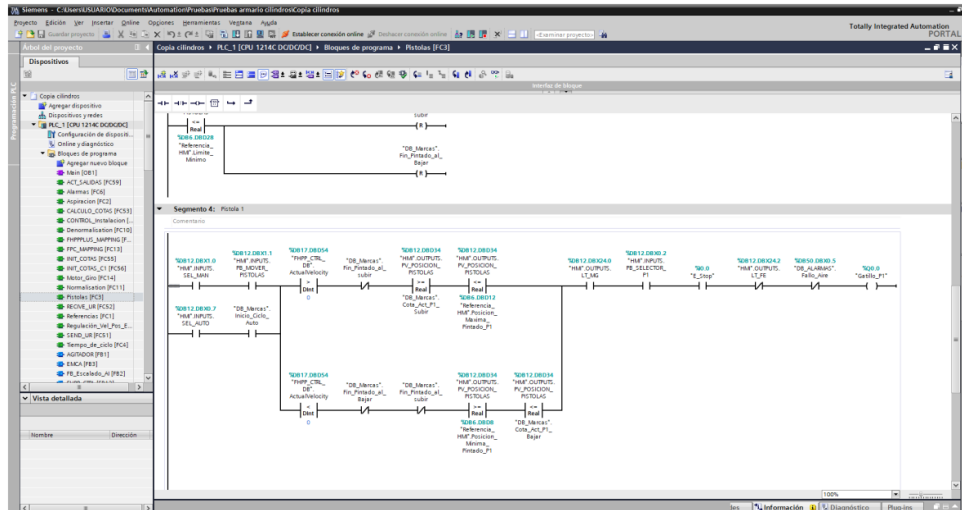


Figura 86. Captura del programa del PLC.

Debido al desconocimiento por parte de la empresa respecto al funcionamiento del PLC, se realizó un pequeño manual en el caso de que surgiera algún problema, especialmente cuando el autómatas se pusiera en modo “STOP”.

Continuando con las pruebas en el robot, se modificó el programa de cilindros de tal forma que se pudiera ejecutar de forma totalmente independiente del PLC, sin control remoto. Para ello se cambiaron todas las señales de entrada y salida (los registros que ya hemos visto) a unas variables que se pudieran modificar y configurar desde la ‘Teach Pendant’.

The screenshot shows the Teach Pendant interface for a robot. The interface is divided into several sections for configuring inputs and outputs. The top section includes a toolbar with icons for 'Ejecutar', 'Programa', 'Instalación', 'Mover', 'EPL', and 'Registro'. Below this, there are tabs for 'Interno' and 'Externo'. The main area is divided into four columns: 'Entrada configurable', 'Salida configurable', 'Entrada digital', and 'Salida digital'. Each column contains various input and output channels with checkboxes and numerical values. At the bottom, there are controls for 'Normal' mode, 'Velocidad' (set to 100%), and a 'Simulación' toggle.

Interno	Entrada configurable	Salida configurable	Entrada digital	Salida digital
Robot	E-Stop <input type="checkbox"/> 4	E-Stop <input type="checkbox"/> 4	Auto-Init <input type="checkbox"/> 4	0 <input type="checkbox"/> 4
Externo	E-Stop <input type="checkbox"/> 5	E-Stop <input type="checkbox"/> 5	1 <input type="checkbox"/> 5	1 <input type="checkbox"/> 5
	Reduced <input type="checkbox"/> 6	2 <input type="checkbox"/> 6	2 <input type="checkbox"/> 6	2 <input type="checkbox"/> 6
	Reduced <input type="checkbox"/> 7	3 <input type="checkbox"/> 7	3 <input type="checkbox"/> 7	3 <input type="checkbox"/> 7
Entrada analógica		Entrada digital de herramienta		Salida digital de herramienta
analog_in[0] 0.00 v Tensión		0 <input type="checkbox"/> 1		0 <input type="checkbox"/> 1
0V 10V				Corriente
analog_in[1] 0.00 v Tensión				000 mA
0V 10V				
Salida analógica		Entrada analógica de herramienta		
analog_out[0] 4,00 mA Corriente		analog_in[2] 0.00 v Tensión		
0V 10V		0V 10V		
analog_out[1] 4,00 mA Corriente		analog_in[3] 0.00 v Tensión		
0V 10V		0V 10V		

Figura 87. Salidas para controlar la ejecución del robot.

Al ir a probarlo en el robot, se comprobó que al utilizar las entradas de control solo funcionaban en el modo simulación y no en el robot real, por lo que hubo que cambiar todas esas entradas por salidas, las cuales sí eran modificables desde el controlador del robot. La ejecución fue tal como se esperaba, salvo por el hecho de que no se logró abrir ni cerrar la pinza, ni desde el robot ni de forma manual desde la pantalla del armario de control.



Figura 88. Opción de control de la pinza en el menú "Manual".

En una presentación con los responsables de la empresa y el tutor del TFG se expusieron las 3 posibles soluciones con las que se contaba hasta el momento, donde a mayores el tutor de empresa dio la posibilidad de una cuarta opción. Esta se basa en giros del robot con la pieza en posiciones oblicuas para que en una posición se pinten unas partes de la pieza (mientras la muñeca 3 del robot rota) y

en otra el resto que no se han pintado porque antes bloqueaba el acceso de la pintura por la geometría de esta. Una ventaja de esta solución es que no haría falta realizar un cambio de agarre, facilitando y acortando el proceso considerablemente.

7.4. Desarrollo de nuevas soluciones y diseño de máscaras y carros

Para esta nueva forma de enfocar el proyecto, se crearon nuevas pruebas para intentar reproducir la idea de la mejor manera posible, utilizando dos posiciones dentro de la cabina de pintura con el mismo punto de actuación, pero distinta inclinación y realizando una rotación de 720° (el máximo de la junta de la muñeca 3) en cada una de ellas. A este nuevo enfoque lo llamaremos *Solución 1*.

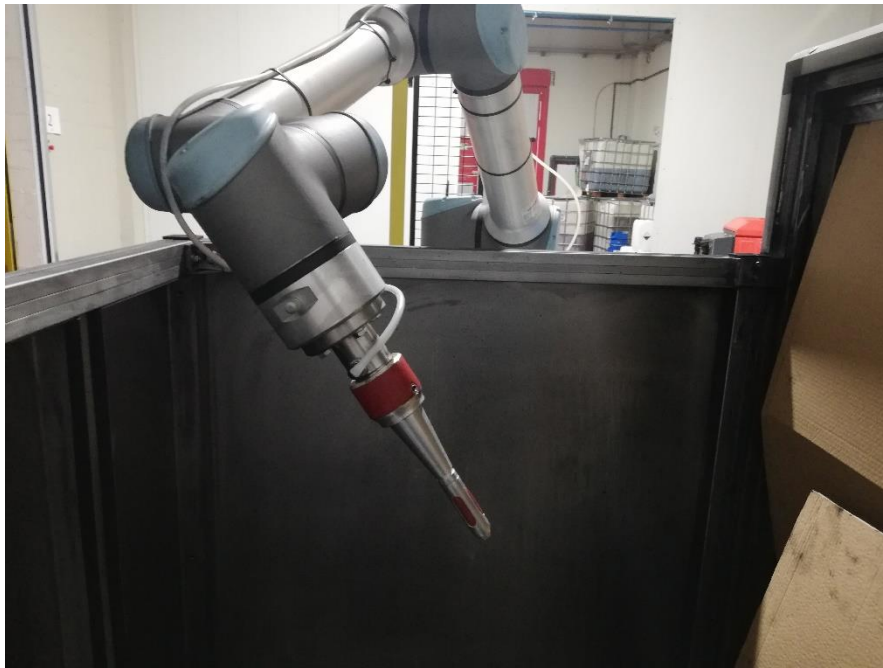


Figura 89. Inclinación del robot para rotar la muñeca 3.

A su vez, se investigó la forma de activar la pinza en el programa de TIA Portal para ver si había alguna forma de accionarla sin utilizar el control remoto (la única forma de hacerlo según el operario encargado de la pintura).

Otro nuevo apartado de trabajo fue el diseño de las máscaras y los carros para la nueva solución. Para ello, se consultó la opinión del operario encargado de la pintura y de preparar la instalación sobre cómo deberían ser las nuevas máscaras y carros, teniendo como referencia las utilizadas en las piezas cilíndricas, como se han mostrado en la *Figura 42*.

Posteriormente se acordó una reunión con el ingeniero mecánico encargado de diseñar las máscaras y los carros para ver cómo hacerlo y las características que debían tener. Para las máscaras, se sopesó la posibilidad anteriormente planteada de introducir una muesca para colocar las piezas siempre en la misma posición, pero acabó siendo descartada por el hecho de que los operarios tenían que colocar las máscaras en las piezas con mucha precisión para que luego la orientación de estas colocadas en el carro fuera la adecuada. Finalmente se determinó utilizar el mismo tipo de máscaras y soportes similares al de los de las piezas cilíndricas, con una circunferencia sin muesca y es el operario el que orienta visualmente y más fácilmente las piezas en el carro.

En cuanto a las primeras 'T' con las que se trabajaron, se necesitan 3 máscaras distintas, que coinciden con el orificio señalado en la *Figura 90*:

1. Correspondiente al orificio que se encaja en los soportes del carro, su forma está adaptada para que solo se pueda retirar la pieza verticalmente, empleando un tronco de cono para que se coloque fácilmente e impida que entre pintura al interior.
2. Se encuentra horizontalmente según la posición de la pieza en el carro, y se trata de una sencilla máscara ciega que protege las zonas internas de la estructura.
3. La más importante de todas y colocada en la parte superior de la pieza, se ha diseñado específicamente para que el robot pueda expandir la pinza y esta se adapte perfectamente, evitando el deslizamiento y que la pieza

pueda caerse durante la operación. Para mejorar tanto el alcance del robot como la efectividad de los movimientos de este respecto a las pistolas de pintura, esta máscara se diseñó prolongando su longitud, alejando así la pieza del brazo robótico.



Figura 90. Numeración de los orificios de la pieza.

Tras instalar TIA Portal en el portátil personal para que fuera más cómodo el desplazamiento hasta el robot y el PLC, se siguió investigando y realizando pruebas para intentar controlar la pinza, activando la presión del aire, modificando ligeramente el programa del PLC e incluso introduciendo un sencillo programa de pruebas, aunque sin hacer que funcione.

Realizando pruebas y visualizando el funcionamiento de la última solución, se planteó una nueva forma de llevarla a cabo, en la que, en vez de rotar en una posición fija, se toma un punto de referencia y es el robot entero el que realiza circunferencias en torno a él con la pieza inclinada, de tal forma que sucede algo parecido a la solución anterior, pintando algunas zonas en una posición y completando la pieza al ir girando o cambiando la orientación de la pieza respecto

a las pistolas. Esta nueva forma de abordar el problema será conocida como *Solución 2*.

El principal inconveniente es la realización de este movimiento ya que, a pesar de cambiar la orientación del robot, como siempre se realiza sobre el mismo punto del TCP, el robot detecta que ya se encuentra en la posición final y no se mueve.

Tras probar distintas opciones, se obtuvo algo que se parecía bastante a la idea planteada, cambiando el punto de aplicación y realizando movimientos circulares entre ellos.

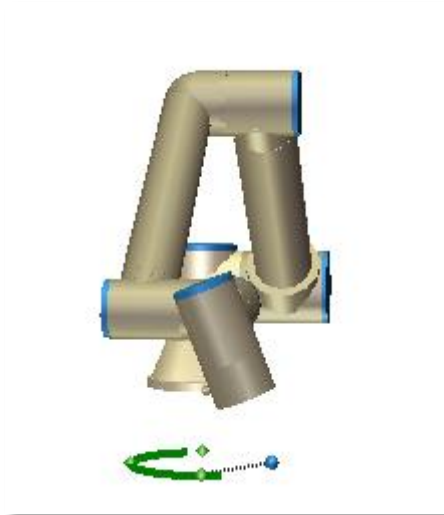


Figura 91. Movimientos circulares con varios puntos.

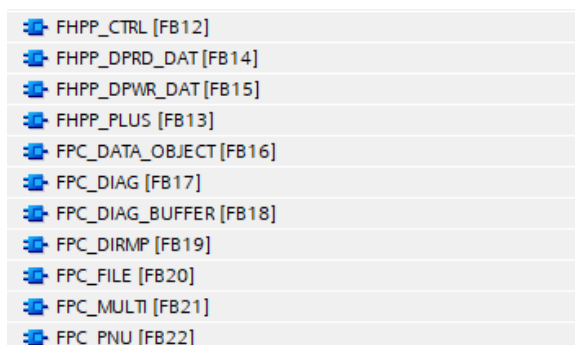
El siguiente paso era encontrar una forma de realizar pruebas reales de pintura con las nuevas soluciones. Para ello, había dos posibles soluciones:

- 1. Manual:** El programa del robot es totalmente independiente del PLC y se controla con las salidas manualmente desde la 'Teach Pendant' del UR. Esto significa que hay que coordinar la sincronización de las pistolas, la pinza y el sistema de extracción de forma totalmente manual, utilizando esperas para poder ir a la pantalla del armario de control de modo manual y activar la acción adecuada pulsando el botón correspondiente. De esta forma es el robot el que tiene el control, no haría falta reprogramar el autómatas y se puede hacer un programa más corto que solo sirva para realizar la prueba.

- 2. Automático:** Modificar esa prueba para adaptarla al control remoto del armario de control. Se toma de referencia el programa de los cilindros y se realiza uno basado en él realizando los movimientos adecuados y mandando y recibiendo señales cuando sea necesario. Para ello, se introducen todos los cambios que ya se habían pensado en su momento, desde el cambio en las cotas hasta las posiciones de pizas y sensores. El mayor problema reside en cambiar el programa del PLC, ya que su programación es más compleja y habría que cambiar muchos apartados, como la activación de las pistolas, la supresión del bloque del soporte rotativo o la recepción y envío de señales al robot. Además, supone la realización del programa íntegro por parte del robot, desde la toma de la pieza en el carro hasta su devolución tras el proceso de pintura.

Como la opción más sencilla era la primera, se investigó a fondo todo lo relacionado con el control manual del PLC, principalmente la activación de la pinza y las pistolas y el movimiento de estas. Para ello, se profundizó en el estudio del control del reciprocador Festo, que constituye la parte más compleja y desconocida de todo el proyecto. Se buscó toda la información posible tanto en los comentarios del programa como en internet, pero no se comprendía con seguridad cómo lidiar con ello, por lo que se mandó un correo con varias dudas a la empresa responsable de la instalación.

En su respuesta comentaron que esos bloques son propios de Festo, que no los han programado ellos y que para ellos son bloques opacos, una caja negra en la que introducen unos valores y obtienen unos valores y unas acciones como resultado. También apuntaron que el control manual de la instalación es bastante pobre, ya que varias de las acciones que presuntamente se pueden llevar a cabo realmente no es posible ejecutarlas.














 FHPP_CTRL [FB12]
 FHPP_DPRD_DAT [FB14]
 FHPP_DPWR_DAT [FB15]
 FHPP_PLUS [FB13]
 FPC_DATA_OBJECT [FB16]
 FPC_DIAG [FB17]
 FPC_DIAG_BUFFER [FB18]
 FPC_DIRMP [FB19]
 FPC_FILE [FB20]
 FPC_MULTI [FB21]
 FPC_PNU [FB22]

Figura 92. Bloques de programa de Festo.

Se realizó otra reunión con el ingeniero mecánico de la empresa para discutir algunos asuntos de las máscaras, el peso de la pieza con ellas puestas y otras cuestiones. En ella comentó que necesitaba tener más información acerca de las pinzas, como la expansión que tienen las membranas de los dedos para realizar las máscaras o el coeficiente de rozamiento de estas, para saber si sería posible que la pinza cogiera a la pieza por la máscara. Esto es necesario debido a que no es lo mismo coger la pieza directamente que hacerlo a través de la máscara, ya que la fricción de los dedos de la pinza con la goma no es la misma que con el aluminio.

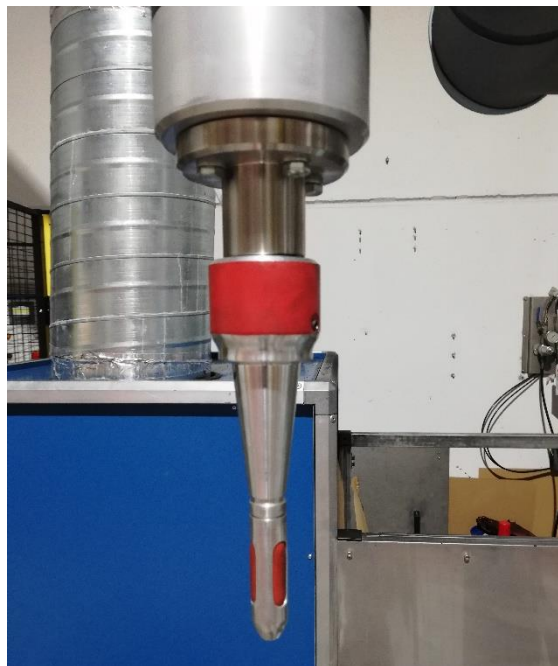


Figura 93. Pinza para la Referencia 1.

Al no encontrar ningún tipo de referencia en la pinza y nada parecido en la página de accesorios de Universal Robots ni en el resto de la red, el gerente comunicó a la empresa que realizó la instalación nuestras dudas, a las cuales respondieron en una visita a la fábrica de la siguiente:

- Las pinzas no son comerciales, las hicieron ellos a medida expresamente para esta instalación, por lo que no tiene una hoja de características de ellas y la mejor forma de ver si sujetan la pieza es probando.

- Como su activación en modo manual parece no funcionar, se puede cambiar e instalar una válvula conectada a las entradas y salidas del controlador del robot para que la podemos abrir y cerrar de la misma forma que se controlan las pruebas con el robot independiente.
- Para el correcto funcionamiento y programación del PLC es necesario descargar las librerías correspondientes de Universal Robots y Festo, además de un software específico para este último.

Tras descargar estos últimos archivos, se acordó una visita por parte de un trabajador de la empresa que suministra los productos de Universal Robots, en la que se le presentaron la instalación, mis ideas y mis dudas. Él solucionó el problema de giro respecto un punto, utilizando un comando de velocidad muy sencillo y fácil de controlar. También aconsejó acerca de las posibles mejoras del proceso, como por ejemplo cortar la chapa frontal de la cabina de pintura para que el robot pueda acceder por ahí y aumente su rango de acción, el cual es bastante justo respecto a la posición de la cabina en la que debe operar. Esto no supondría un problema debido a que la chapa se corta fácil y que el extractor tiene suficiente poder de succión como para que no se note la falta de este elemento de protección, pudiendo poner en caso de que sea necesario unas láminas de caucho para reemplazar la chapa que permitan la entrada del robot.



Figura 94. Chapa de la cabina frente al robot.

Cierto día se realizó una prueba junto a la empresa instaladora debido a que había que preparar una muestra de pintura para una empresa interesada en comprar estas piezas. Para ello, se puentearon algunos contactos para realizar una prueba de forma independiente y bastante manual en la que se colocó la pieza tipo 'T' en el soporte rotativo con unas máscaras improvisadas. Una vez colocada, se activaba el extractor, el giro del soporte y a continuación el ciclo de las pistolas de subida y bajada con el gatillo activo. El resultado, aparentemente, fue mejor de lo esperado, ya que el acabado parecía bastante homogéneo dadas las condiciones.

Aun así, surgieron varias complicaciones a mejorar:

- La velocidad de giro del soporte debe ser baja, para que la pintura tenga más opciones de acceder a las zonas más difíciles y a la vez reducir las vibraciones y problemas surgidos por la inercia añadida del otro brazo de la 'T'. Esto se puede solucionar fácilmente desde el menú "Recetas" de la pantalla del armario de control.
- El tiempo de pintura debe ser superior, para garantizar mayor homogeneidad en todas las partes de la pieza. Para cambiarlo, solo hay que introducir un valor superior en las recetas o modificarlo directamente desde el programa del PLC.
- Un factor a tener en cuenta es la cantidad de pintura que se pierde, ya que en este caso hay que orientarlas de tal forma que pinten, aparte de la zona cilíndrica, el brazo transversal. Esto produce la pérdida de pintura al ampliar el área de pulverización de las pistolas. En este caso no hay una solución clara, ya que es muy difícil limitar estas pérdidas si queremos que la pieza quede bien cubierta en su totalidad, salvo que orientemos de forma muy precisa las pistolas y regulemos su difusión con sumo cuidado.
- El mayor problema de este método de pintura es el acceso a las uniones y zonas más complejas de la pieza. Lo único que se puede hacer es, como en el caso anterior, orientar correctamente las pistolas y regular su difusión.

Una vez contemplada esta posibilidad, se siguió con el desarrollo de las otras soluciones aplicando el nuevo comando de giro en vez de los movimientos circulares entre varios puntos como se había hecho en la primera aproximación. Tras probar varias configuraciones y valores, se logró realizar el movimiento

deseado con una inclinación aceptable para que realizara aproximadamente 2 rotaciones completas.

Como ya se tenía a grandes rasgos los movimientos dentro de la cabina de pintura para dos posibles soluciones, se inició la búsqueda de información para realizar los cambios necesarios para las nuevas piezas y soluciones. Para ello, se recopiló toda la información acerca de las posiciones de las piezas en los carros en el programa de TIA Portal y del robot, incluyendo los valores estándar introducidos en la pantalla HMI del armario de control. También se midieron los cambios de posición necesarios para la detección de pieza a la entrada y salida de la cabina de pintura, junto con la posición para colocar la pieza en el soporte rotativo en el caso de utilizar esta otra solución.

Para ampliar los conocimientos respecto a los elementos del armario de control, se inició una lectura de información y visualización de tutoriales acerca de la programación de la pantalla HMI de SIEMENS KTP700 Basic. De esta forma, sería posible modificar la interfaz de la pantalla de control, añadir botones, introducir más valores o crear nuevas recetas [7].

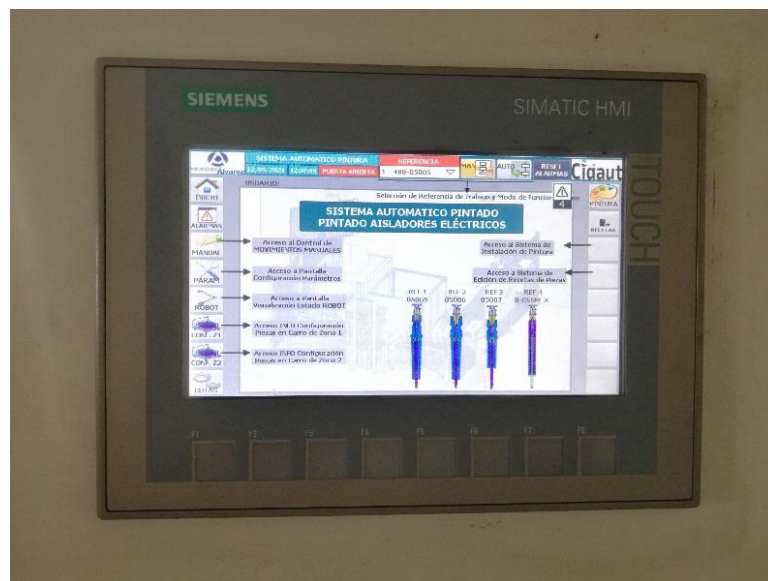


Figura 95. Pantalla HMI del armario de control.

Con esto acabé mi periodo de prácticas.

Aún quedaba mucho trabajo por hacer, siendo de primera necesidad empezar a realizar pruebas de pintura y testear las distintas soluciones para comprobar cuál de todas ellas era la más adecuada. Esto estaba planeado llevarlo a cabo en el periodo restante de la ampliación de prácticas en empresa, expuesto en las páginas siguientes. Destacar que cada dos semanas se realizaba un informe con los avances o las tareas realizadas que se enviaban al tutor de empresa para que este hiciera un seguimiento del progreso y poder corregir o aconsejar en el caso de que hubiera algo que no estuviera bien hecho o perjudicara a la empresa.

7.5. Inicio del periodo de ampliación de prácticas

Teniendo en cuenta lo último que se hizo durante las prácticas, teníamos la solución que se probó utilizando el soporte rotativo de la cabina de pintura y pintando la 'T' con el mismo programa que se utiliza para las piezas cilíndricas.

El problema es que, debido a la gran cantidad de inconvenientes, esta se trataba de una opción controvertida y que, a la postre, se demostró que no era válida, gracias a ciertas comprobaciones tanto visuales como eléctricas, llevadas a cabo por el equipo de investigación y control de calidad en el laboratorio eléctrico de la fábrica.

El mayor punto negativo se encuentra en la pintura de las juntas mostradas en la *Figura 85*, que al tener la pieza totalmente vertical no llegaban a cubrirse correctamente, dando lugar a problemas de rendimiento de conducción y aislamiento.

Debido a esto, se continuó con el desarrollo de las otras soluciones que involucraban al robot trabajando dentro de la cabina de pintura, sujetando la pieza.

Por ello, se realizaron modificaciones al programa correspondiente a la *Figura 91* para poder adaptarlo al control remoto del PLC y que su ejecución fuera automática desde el armario de control, realizando la comunicación mediante los registros mencionados en el análisis del programa en TIA Portal.

Teniendo en cuenta las soluciones a desarrollar, se modificó el programa del PLC para adaptarlo a las nuevas necesidades. El principal cambio consistía en subir las pistolas al inicio del proceso para dejarlas fijas en una posición en la que actuará el robot con la pieza, subiendo y bajando, para pintar uniformemente la totalidad de la pieza. Las pistolas permanecen ahí hasta que finaliza la pintura de todas las piezas del carro, momento en el cual descienden hasta la posición de reposo inicial.

Para tener una referencia de la posición de las pistolas, se utilizó el modo manual de la pantalla del armario de control, probado distintas alturas y colocando el robot de forma manual en la cabina para dejarlo completamente alineado.



Figura 96. Control de pistolas en modo manual de la pantalla HMI.

La programación de la posición de las pistolas se realiza mediante unos bloques de programa creados por el fabricante del reciprocador (Festo), los cuales son difíciles de entender y se necesitó pedir ayuda a los encargados de la instalación

sobre cómo hacer las modificaciones. Estos contestaron que simplemente había que introducir unos datos y que los bloques devolvían las salidas necesarias, y que era necesario descargarse un software específico para poder controlar adecuadamente el reciprocador.

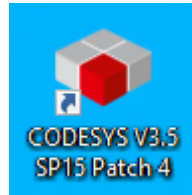


Figura 97. Software CODESYS para el control de las pistolas.

En una visita al robot, se realizó otra descarga de la programación del PLC y se instaló otro archivo “.gsdml”, consiguiendo así más información de la pantalla HMI y los elementos de la estación “emca” y “giro”, correspondientes al reciprocador y al motor que hace girar el soporte rotativo de la cabina. También se llevaron a cabo pruebas de rango y alcance con el movimiento de giro del robot en torno a un punto, conociendo así los límites y las mejores posiciones para que acceda y trabaje en la cabina de pintura.

A modo de experimento, se anclaron las piezas en la pinza del robot y se ejecutaron los programas con las dos posibles soluciones. El resultado fue satisfactorio, ya que el robot se desempeñaba con normalidad y realizaba los movimientos programados sin inconvenientes. Para empezar a realizar pruebas de pintura estaba todo listo, solo faltaba pulir algunos detalles relacionados con la programación, recibir la funda protectora del robot frente a la pintura y, a la hora de automatizarlo por completo, los carros y máscaras apropiados para las distintas piezas.

7.6. Modificaciones y mejoras de las dos soluciones finales

A la espera de conseguir los elementos necesarios anteriormente citados, se realizaron algunos ajustes y mejoras a los programas del robot.

En primer lugar, se definió un nuevo punto de actuación del robot para mejorar el movimiento de este alrededor de un punto, adecuándolo a las dimensiones de la cabina, el alcance del robot y la inclinación necesaria para la correcta pulverización de la pintura sobre la pieza.

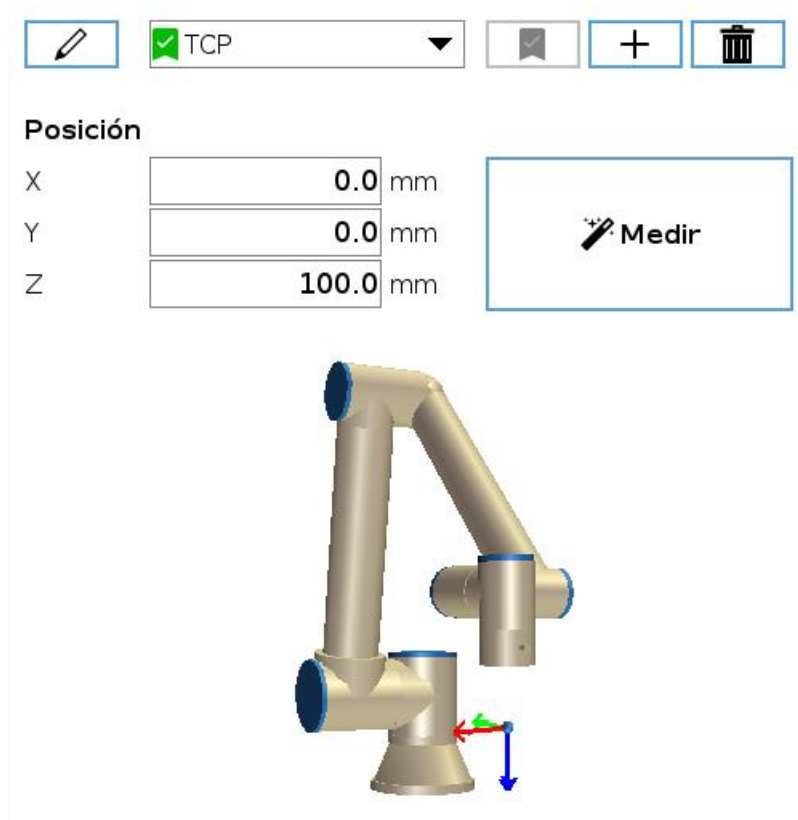


Figura 98. Configuración del TCP original.

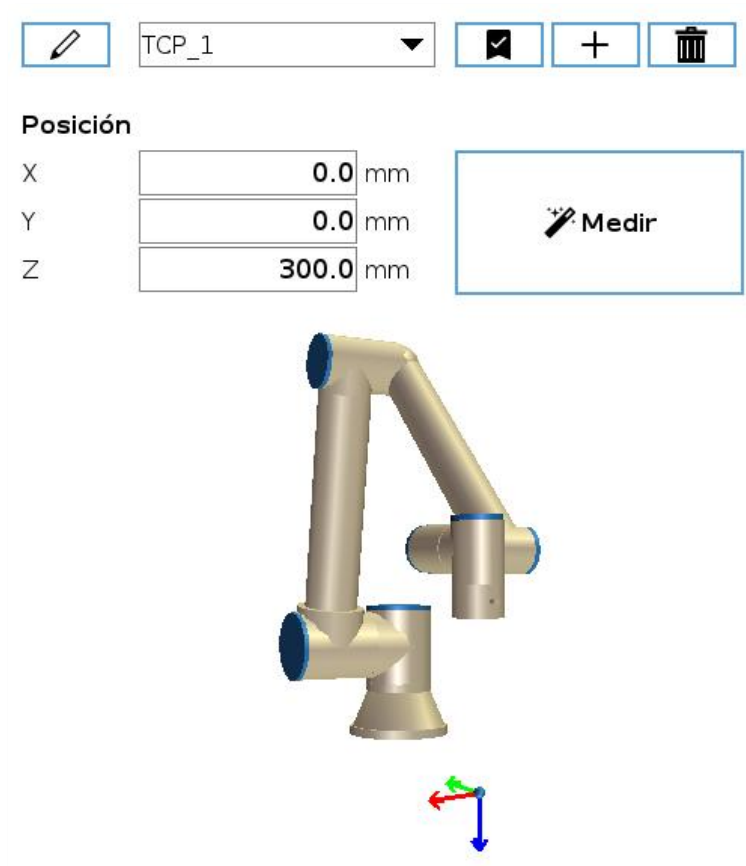


Figura 99. Configuración del nuevo TCP_1.

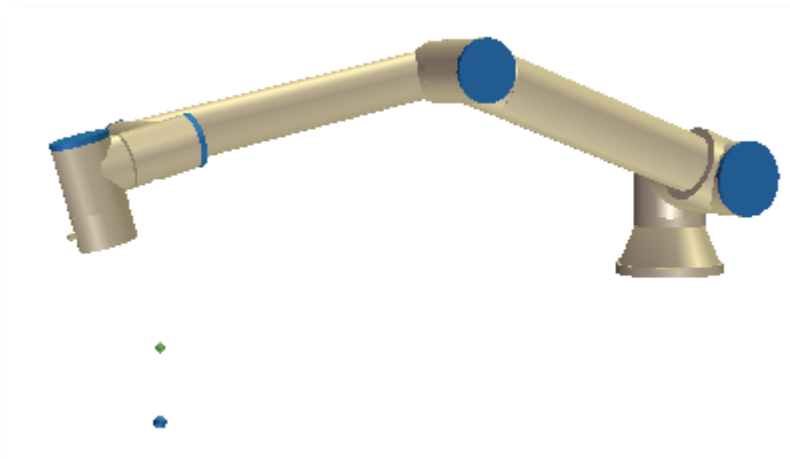


Figura 100. Diferencia de TCP en la rotación en torno a un punto.

Siendo el punto verde el TCP original y el punto azul el nuevo TCP_1.

Hablando con el gerente de la empresa, se llegaron a las siguientes conclusiones:

- La funda del robot estaba en camino, aunque no se sabe con certeza cuándo llegará.
- La conexión del tubo de aire con la pinza que hace que esta se expanda sufre mucho en algunos movimientos, especialmente en las soluciones que he propuesto, donde se utiliza fundamentalmente la rotación de la muñeca 3.



Figura 101. Problemas con el tubo de alimentación de presión de la pinza.

Una posible solución es cambiar la junta de unión por una rotativa, dando más libertad al conjunto y evitando que el conducto se enrosque en el robot.

- Para mejorar el rendimiento y la pintura de las piezas, en vez de dejar las pistolas fijas mientras sube y baja el robot con la pieza, se puede hacer que las pistolas también realicen un desplazamiento vertical (similar al que realizan con las piezas cilíndricas) mientras el robot se mueve en la cabina de pintura. La idea es que las pistolas realicen un ciclo de movimiento muy corto, siendo el robot el que realiza el mayor desplazamiento y consiguiendo así distintas posiciones entre las pistolas y la pieza, lo que, en principio, ayudaría a llegar a todas las partes de la pieza.
- Como ya propuso el trabajador que vino por parte de la empresa que suministra los componentes robóticos, la posibilidad de cortar la chapa frontal de la cabina de pintura está ahí en el caso de que se requiera

aumentar ligeramente el rango de acción del robot, tanto horizontalmente como alcanzar posiciones inferiores dentro de la cabina de pintura.

Se realizaron pruebas en el robot real del programa con la solución correspondiente a la rotación del robot en torno a un punto (*Solución 2*), cuyos resultados fueron correctos, a falta de seguir modificando y ajustando movimientos.

Posteriormente se debatió con el tutor de empresa cómo solucionar el hecho de que existen varios tipos distintos de piezas con geometrías significativamente dispares, requiriendo cada una de ellas una configuración de pintura que, aunque tengan la misma base en los movimientos y giros, debe ajustarse para cubrir las piezas en su totalidad. Tomando como referencia las soluciones llevadas a cabo para las distintas piezas cilíndricas, se llegaron a varias conclusiones y posibilidades.

- Lo mejor sería intentar utilizar los mínimos cambios dentro de la configuración y de la activación del programa para facilitar la labor del operario y no tenga que estar pendiente de cambiar de programas cada vez que quiera pintar una pieza distinta.
- Por ello, se debía añadir una nueva receta a mayores de las 4 actuales dedicadas a las piezas cilíndricas para poder pintar la pieza 'T' con la que se estaba trabajando en esos momentos. Esto se realizaría en TIA Portal, modificando tanto el programa del PLC como la configuración de la pantalla HMI para poder introducir y modificar fácilmente los valores de posición y velocidad de pistolas, tiempo de ciclo, etc.



Figura 102. Valores configurables de cada receta.

- En la programación del robot, se podría realizar un programa para cada una de las piezas, haciendo que el operario tuviera que cambiarlo en la pantalla de control del robot al ir a pintar una pieza distinta. La otra solución sería crear un programa único para este tipo de piezas 'T' y puede que 'X' (independiente de las cilíndricas) en la que se leyera la referencia introducida y se seleccionara una rama del programa mediante 'IFs' para cada tipo de pieza, con los ajustes de posiciones y movimientos que cada una conlleva.

Para introducir los valores adecuados en las recetas de la pantalla primero se deben saber las posiciones en las que actuará el robot, las de las pistolas y el tiempo de ciclo o el tiempo que van a pintar las pistolas mientras el robot esté en la cabina, estimando inicialmente posibles valores para cada variable. Realizando una primera medición de este último parámetro en el simulador, se obtuvo que el tiempo de pintura, comparado con el tiempo promedio utilizado para las piezas cilíndricas es bastante inferior y teniendo en cuenta que las piezas 'T' por su geometría requieren más tiempo de aplicación, se modifica el programa para que los movimientos en cabina se repitan hasta cuatro veces, dando lugar a un tiempo superior pero seguramente insuficiente, sirviendo aun así como referencia y base sobre la que partir y mejorar el programa.

Un inconveniente de vital importancia para el rendimiento del proceso era la pérdida significativa de tiempo en los cambios de posiciones a la hora de repetir el bucle, debido a que las rotaciones máximas de la muñeca 3 son 720° , pudiéndose valorar incluso la posibilidad de cerrar el gatillo de las pistolas durante estos cambios para no perder pintura. Debido a esto, se creó una nueva versión del programa en la que se eliminaban los cambios de posiciones intermedios y se añadía una espera simbólica (0.1 segundos) entre cada movimiento para que así no surgieran problemas con los límites de las juntas. De esta forma también existe la posibilidad de variar la inclinación en cada una de las repeticiones (muy limitado por el alcance del robot como se ha visto anteriormente) para que la orientación de las pistolas, respecto a las partes de la pieza, cambien constantemente.

Algo parecido se llevó a cabo con la opción de la rotación en una posición oblicua fija (*Solución 1*), donde se realizaban varias rotaciones de la muñeca 3 en la misma posición para alargar el proceso y que se distribuyera bien la pintura.

Para la *Solución 2* se retocaron ciertos valores para lograr una aceleración mayor al inicio del movimiento y una velocidad máxima reducida con el fin de homogeneizar los movimientos para que fueran lo más regulares y continuos posibles, a pesar de las paradas necesarias durante el cambio de dirección al alcanzar los límites de las juntas.

De forma similar se realizaron estos mismos ajustes en la *Solución 1*.

Estas nuevas pruebas se llevaron a cabo finalmente en el robot real para visualizar si su funcionamiento era el correcto. El resultado fue bueno, aunque existían distintas mejoras que sugirió el responsable de la instalación como la posibilidad de que el robot se mueva verticalmente junto con las pistolas a diferente ritmo para mejorar el acabado.

También se habló con la empresa con la que comparten edificio de oficinas para crear una nueva pinza adaptada a este nuevo tipo de piezas. Esto se debe a que no parece haber ningún tipo de junta rotativa que sirva para el suministro de presión de la pinza actual. Esta entidad, especializada en el diseño y fabricación mediante impresión 3D, emplea una amplísima gama de materiales (*Figura 103*), lo que les permite crear casi cualquier tipo de estructura usando el material que quieran para todo tipo de aplicaciones.

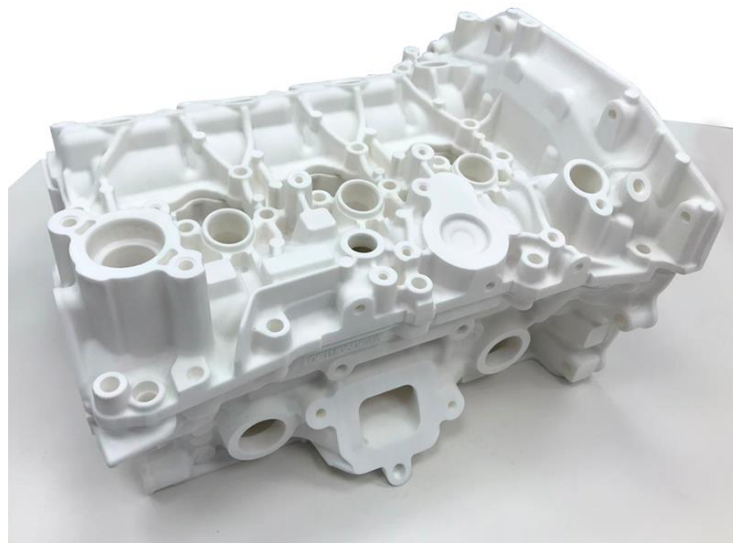


Figura 103. Ejemplo de impresión 3D por parte de la entidad.

Mientras tanto, se realizaron las modificaciones pertinentes para hacer subir y bajar el robot durante la ejecución en la cabina de pintura.

Para la *Solución 2* se realizaron 4 bajadas mediante movimientos relativos hasta un total de 12 cm en los momentos en los que se llegaba a completar las 2 rotaciones de la muñeca 3, para volver a subir siguiendo la misma filosofía tras realizar el cambio de posición intermedio.

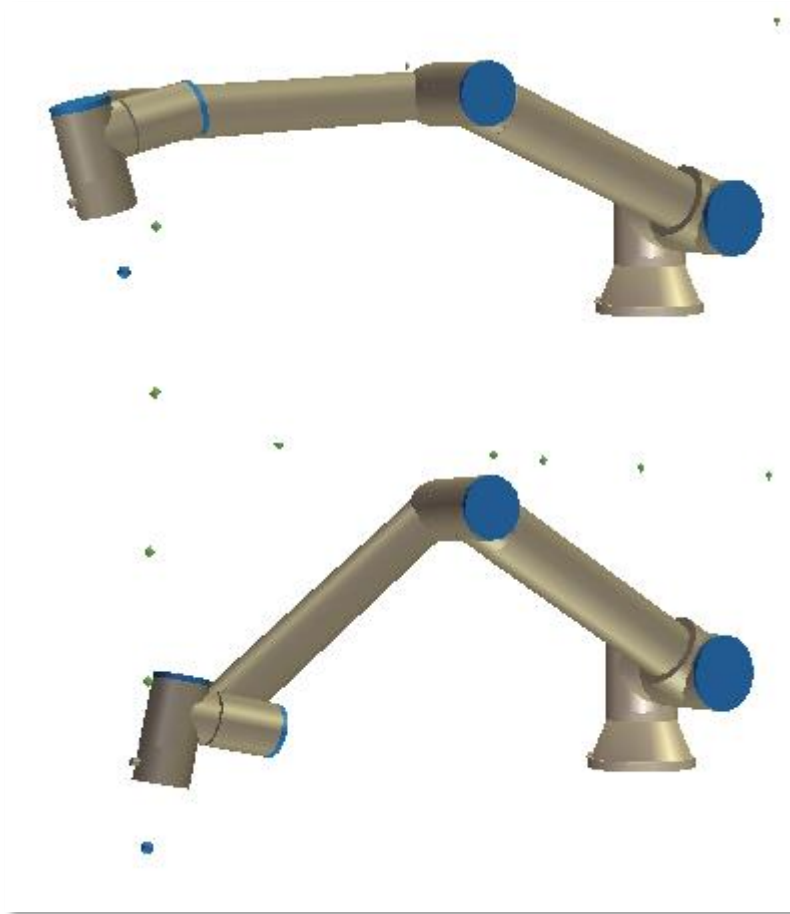


Figura 104. Diferencia entre el punto inicial del movimiento y el punto de cambio de posición intermedio.

La idea para la *Solución 1* era la misma, pero con movimientos relativos no funcionaba correctamente, ya que era necesario realizar muchos cambios y las rotaciones de muñeca no se detectaban de la manera esperada, al no realizar las dos rotaciones correspondientes sino una pequeña rotación que no servía para nada. Es por eso que hubo que hacerlo con múltiples posiciones fijas, una para descenso o subida y otra para rotar la muñeca.

Tras realizar estos últimos cambios se llevaron a cabo las pruebas correspondientes en el robot real para ver el resultado en conjunto, con el espacio de trabajo y el resto de elementos de la instalación.

La *Solución 2* tuvo un desempeño correcto, aunque surgió la duda de poder integrar el movimiento ascendente o descendente durante las rotaciones en torno al punto, y no al finalizar las dos rotaciones como se hacía anteriormente. De esta forma se evitaba ese tiempo que antes se “perdía” en que el robot subía o bajaba sin rotar, lo que exponía una parte de la pieza durante demasiado tiempo a las pistolas hasta el punto de plantearse la posibilidad de parar la pulverización de pintura durante este corto periodo de tiempo. Esta modificación se pudo realizar aprovechando el comando ‘speedl’ con el que se creó el movimiento entorno a un punto, simplemente añadiendo un valor en el eje z durante la ejecución de la instrucción.

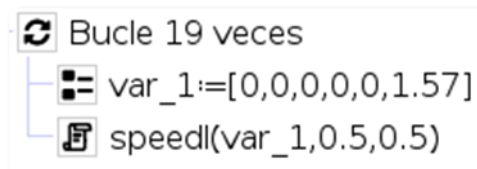


Figura 105. Detalle de la instrucción ‘speedl’.

Para la *Solución 1* también se pensó hacer algo parecido, eliminando así el problema de añadir muchas posiciones intermedias. Esto permitía utilizar un bucle en el que el robot comience desde una posición superior y se desplace a una inferior en vertical mientras gira la muñeca 3 hasta completar los 720 grados, momento en el cual comienza a ascender rotando en el sentido contrario hasta la posición inicial. En caso de no resultar una solución adecuada, se podrían realizar distintas configuraciones para que el robot subiera y bajara en determinados momentos, más lentamente o incluso con diferentes velocidades.

Durante la visita al robot también se midieron referencias y posiciones del sensor de presencia para cambiar el punto de detección de la pieza a la entrada y a la salida de la cabina de pintura, bajando este punto unos 20 cm aproximadamente.

Para continuar con las modificaciones necesarias para añadir una quinta receta al programa, se buscó cómo obtener la programación de la pantalla HMI para así ver cómo funciona y por si era necesario modificar algo. Para esto se envía un correo a la empresa responsable de la instalación para que faciliten información.

Mientras se esperaba la respuesta, se valoró la posibilidad de introducir un nuevo registro de intercambio de información entre el robot y el PLC para así desactivar los gatillos de las pistolas de pintura durante los movimientos de cambio de posición. Esto se lograría redefiniendo uno de los registros sin utilizar de la comunicación entre elementos, escribiendo en él desde el robot para indicar que se va a realizar ese cambio de posición y que se debe parar la pulverización de pintura.



```
write_output_boolean_register(17, False )
```

Figura 106. Detalle de escritura en registro desde el robot.

A su vez, el PLC debe recibir la lectura de ese registro y actuar sobre los gatillos de las pistolas, añadiendo un nuevo contacto asociado al nuevo registro en el segmento de activación de los gatillos.

Para las posibles pruebas automatizadas, se realizó una copia de los programas que realizan las dos soluciones principales y se modificaron introduciendo todos los registros de comunicación con el PLC para así lograr la ejecución del programa de forma remota.

Tanto estas dos nuevas versiones como las anteriores se ajustaron para cambiar las posiciones de detección de pieza anteriormente mencionado, a la espera de

tener las medidas y posiciones de las piezas en los carros para así ajustar los puntos de cogida y dejada. A mayores, se hicieron pruebas de optimización de movimientos y posiciones en los programas anteriormente mencionados, con el fin de evitar desplazamientos innecesarios del robot y reducir el tiempo empleado para la pintura de cada pieza.

7.7. Pruebas con la funda del robot

Finalmente llegó la funda del robot. Tras limpiar el robot de las partículas de pintura depositadas en él, se intentó poner la funda de varias formas, aunque sin conocer cuál era la correcta de todas ellas debido a la falta de instrucciones con ella. Se realizó una búsqueda de algún tipo de manual de instrucciones para saber cómo se debe colocar, qué parte corresponde a la base del robot o la posición que debe tener este.

Otro factor a tener en cuenta era la carga que lleva el robot en cada momento y el centro de gravedad de esta, teniendo en cuenta el peso y la longitud de la herramienta que la sujetaba. Esta información es interesante para el robot debido a que, a partir de estos valores, el autómata se autorregula en función de la posición para destinar más o menos potencia entre las articulaciones del brazo, logrando así que los movimientos y posiciones finales no se vean afectadas por la carga. Para ello se realizaron distintas pruebas tras medir y pesar las piezas con la ayuda del ingeniero de diseño e introducir estos valores en el programa del robot

Finalmente se comienzan a probar distintos movimientos e incluso programas en el robot real con la funda puesta para comprobar la movilidad y si podía alcanzar las posiciones más “retorcidas” o más alejadas de la base, donde el brazo debe extenderse casi en su totalidad.



Figura 107. Imagen de un UR10e con funda protectora.

También se llegó a probar a pintar una pieza cilíndrica con su programa automático correspondiente tras previa comprobación el programa adaptado que se realizó para poder ejecutarlo independientemente del PLC. El problema es que rápidamente advertimos que el robot se detenía antes de llegar a la cabina de pintura con la pieza. En un primer momento se pensó que podía deberse a un problema con la funda, que impedía la total extensión del brazo para alcanzar las posiciones más alejadas, pero fue descartado debido a que sí completaba el programa sin control del PLC. Se valoró la opción de que podía ser el sensor de detección piezas que, al ser la funda ligeramente holgada, detectaba la presencia de un objeto y bloqueaba la ejecución del programa.

Las siguientes jornadas de trabajo se dedicaron a investigar si esta podía ser la causa real, repasando el programa del PLC relacionado con el sensor y las alarmas, pero sin encontrar una respuesta concluyente.

En modo automático continuaba deteniendo su ejecución al ir a dejar la pieza, confirmando que se trataba del sensor de detección de pieza que se activaba con la funda. Para solucionar este problema, se barajaron tres posibles opciones:

1. Subir la posición de dejada de piezas, soltando así la pieza desde mayor altura en el soporte de la cabina de pintura.
2. Ajustar la funda al robot mediante bridas o algún tipo de cuerda para que no cuelgue.
3. Cambiar de posición el sensor, pudiéndolo bajar ligeramente para que no dé problemas.

La primera solución se descartó rápido, debido a que se podría solucionar el problema al dejar la pieza, pero a la hora de recogerla del soporte tras haber pintado el robot alcanza una posición a menor altura que no se puede modificar, detectando en esta situación la funda como sucedía anteriormente.

La segunda opción se realizó de forma provisional utilizando cinta adhesiva para intentar pegar la funda al cuerpo del robot para que no colgara y la detectara el sensor, buscando posteriormente una solución más adecuada como bridas o algún tipo de cuerda. Tras realizar varios ajustes añadiendo más cinta adhesiva, se consiguió pintar una pieza de forma automática sin detectar problemas. El problema es que al realizar una segunda prueba se obtuvo el mismo fallo del principio, debido a que la funda (ya sea porque estaba mal puesta o porque se había fijado mal la cinta) se había desplazado ligeramente debido a los movimientos del robot. Esto en principio se podría solucionar con una fijación de la funda a la base del robot para que no se moviera, evitando así la zona colgante que detecta el sensor. Al realizar esta acción, presenta la dificultad de que corresponde con una articulación del robot, dificultando la posibilidad de utilizar bridas o elementos similares.

Viendo esto, la tercera solución parecía la más posible, teniendo que bajar el sensor e incluso ponerlo más cercano al robot, cambiando evidentemente la posición de detección del robot.

Para esta solución podían surgir a su vez tres variaciones, dependiendo de cómo funcionara el sensor a la hora de detectar objetos durante la ejecución del programa.

1. Desde la posición actual de detección, realizar otro movimiento descendente hasta la nueva posición del sensor para que este lea la pieza para luego volver a subir y proseguir con el funcionamiento normal. La duda es si, antes o después de realizar la lectura intencionada del sensor, este puede detectar la pieza como una intrusión en la cabina y detener el programa.
2. Realizar el mismo movimiento que en la solución anterior pero teniendo en cuenta la longitud de cada tipo de pieza, modificando en función de la longitud la distancia que tiene que bajar el robot con la pieza para que el sensor la detecte, realice la lectura y vuelva a subir. Habría que modificar el programa con las distintas opciones en función de la referencia, siempre y cuando sirvan para distinguir la longitud de cada pieza.
3. La última posibilidad era emular la forma de detección que se lleva a cabo en el proceso actual, con un movimiento horizontal en el que no influya la longitud de la pieza. Para ello, el robot se movería a una posición de aproximación a la altura del sensor, se desplazaría hacia adelante hasta que este detectara la pieza y luego el robot retrocedería a la posición anterior horizontalmente para proseguir con el proceso.
El problema de esto puede ser que se pierda algo de tiempo al realizar esta serie de movimientos al entrar y al salir de la cabina de pintura.

Inicialmente se realizó un programa para cubrir la primera solución que se mencionó con anterioridad, realizando un movimiento vertical para que el sensor detectara si el robot lleva pieza o no. Se realizaron dos versiones, una para poder probar los movimientos con el robot en modo manual y otra para control remoto desde el PLC.

Y aquí finalizaron las horas correspondientes a la ampliación de prácticas y, por tanto, el trabajo llevado a cabo en la instalación de la empresa. La idea era seguir desarrollando el proyecto hasta poder completarlo, ya fuera mediante prácticas

extracurriculares o mediante un contrato de trabajo para poder permanecer en la empresa.

La realidad es que, según me hizo saber la directiva, finalmente habían descartado implementar mi trabajo en la fábrica, debido a la alta demanda tanto de piezas cilíndricas como de las nuevas con forma de 'T' y 'X'.

Esto dio lugar a dejar el proyecto a medias y principalmente a falta de probar a pintar piezas con los programas desarrollados para comprobar cuál de las dos soluciones era la más eficaz y poder realizar los ajustes necesarios para lograr la homogeneidad necesaria de la capa de pintura.





8. Resultados

En este apartado analizaremos los resultados finales obtenidos en este proyecto.

El objetivo final del mismo era la pintura de estas piezas de geometría compleja buscando un grado de homogeneidad óptimo. Debido a la finalización del tiempo en la empresa y a la decisión de esta de abandonar el proyecto, este no pudo concluirse como se esperaba. Es por eso que se presentan los resultados obtenidos hasta ese momento, mostrando así el grado de desarrollo alcanzado.

Por otra parte, debemos tener en cuenta que, debido a las características del mismo, los resultados no se pueden presentar como habitualmente se hace con tablas de valores o gráficas donde se visualiza la evolución o el funcionamiento. Partiendo de esta premisa, podemos encontrar los siguientes:

1. A nivel académico y para enriquecimiento personal, el punto más importante es el de la programación, ya que se han creado los programas tanto del robot como del PLC para llevar a cabo de forma funcional las acciones deseadas para pintar las piezas. Por parte de la programación del robot tenemos 2 programas por cada una de las dos soluciones finales vistas en el desarrollo del proyecto, uno para su ejecución independiente controlado por las salidas manuales y otro para ejecutarlo de forma remota controlado por el armario de control. Todos estos programas se encuentran tanto en .txt como en el formato preparado para la ejecución en el robot o el simulador del ordenador.

En el caso del programa del PLC, se puede visualizar y editar en TIA Portal.

2. Por parte de la empresa, el principal beneficio es el resultado de la programación, es decir, lo que hace el robot y el resto de la instalación gracias a los programas diseñados para poder pintar correctamente las piezas. La única forma de comprobar estos resultados es visualizar de primera mano en la fábrica el desempeño del robot y las pistolas de

pintura. Otra opción más fácil pero menos representativa es utilizar los simuladores de los respectivos programas, reproduciendo los programas del robot en la pestaña “Gráficos” o utilizando la herramienta PLCSIM para comprobar el funcionamiento de los contactos en el PLC.

3. Por último, cabe destacar el resto del trabajo realizado más allá de la programación y control de la estación de pintura. En este apartado se pueden englobar ejemplos como el diseño de máscaras y carros para las piezas, la compra de la funda protectora del brazo, las sugerencias de mejora respecto al estrangulamiento del conducto de activación de la pinza, la realización de manuales de ayuda para la entidad o la búsqueda de información respecto a los elementos que componen la instalación como las pinzas o el control de las pistolas.

De lo anterior se desprende que, si bien no se ha logrado implantar la solución propuesta, el trabajo realizado ha permitido conocer en profundidad las posibilidades de las instalaciones y elementos (PLC, robot, etc) actuales, sus limitaciones y las mejoras necesarias tanto en este proyecto concreto como en otros similares dentro de la empresa.

Para los dos primeros puntos, se ha generado un enlace que contiene los ficheros de los programas, tanto los del robot como del PLC y algunos vídeos correspondientes a la ejecución de los distintos programas del robot, capturados desde el simulador en el ordenador personal.

https://drive.google.com/drive/folders/1q1gQMo_ iyBmUt02rcNMj6nk72EIBl8DQ?usp=sharing



9. Líneas futuras de trabajo

Como ya se adelantó al final del apartado 7, la empresa iba a desestimar la puesta en marcha del proyecto.

Esto se debe a que de cara a los próximos meses y años está recibiendo multitud de pedidos de estos tipos de piezas, por lo que la instalación actual con el robot sería insuficiente para cubrir esa demanda. Del mismo modo ha aumentado la producción de las piezas cilíndricas, dando lugar a que necesita utilizar la estación actual a tiempo completo para poder pintarlas todas.

En su lugar, la entidad ha optado por la adquisición de una máquina prácticamente hecha por encargo especializada en la pintura de estas piezas con forma de 'T' y 'X'.

De esta forma consiguen liberar de carga de trabajo a la actual instalación de pintura para poder aumentar la producción de piezas cilíndricas, mientras que la nueva estación (que llegará a comienzos del año siguiente) permitirá garantizar la homogeneidad de la pintura de las piezas de geometría compleja y podrá pintar a un ritmo superior al que ofrecía el brazo robótico.

Si por el contrario nos centramos únicamente en el proyecto y tenemos como objetivo desarrollar las partes restantes hasta completarlo, habría que:

- Principalmente comenzar a realizar pruebas de pintura. Teniendo los programas completos solo faltaría poder comprobar su funcionamiento pintando piezas.
- Para llevar a cabo el punto anterior y tras conseguir finalmente la funda protectora del robot, faltaría crear las nuevas máscaras y carros. Ambos fueron diseñados de forma conjunta con el ingeniero mecánico y de

diseño de la empresa, pero debido a la adquisición de la nueva estación estos prototipos no se hicieron realidad.

- Del mismo modo se consultó la posibilidad de adquirir una nueva pinza para el robot diseñada por la empresa vecina, la cual serviría para coger las piezas complejas evitando el estrangulamiento del tubo de alimentación de la pinza anterior y el posible deslizamiento con las máscaras.
- Tras realizar las pruebas iniciales pertinentes, modificar los programas y la instalación si fuera necesario (el sensor de detección de piezas) para lograr el correcto funcionamiento del proceso y la homogeneidad deseada en las piezas.
- Una vez se tuviera el proceso en perfecto funcionamiento y produciendo un resultado satisfactorio, se podrían buscar mejoras para maximizar el rendimiento, reduciendo tiempos muertos u optimizando movimientos intermedios de posicionamiento.
- Por último, faltaría documentar todo el proceso de forma que la empresa tuviera un documento con información acerca del trabajo realizado, información acerca de la programación y un manual para poder ponerlo en marcha o actuar en caso de aparecer un problema.

10. Conclusiones

Una vez finalizado el periodo de prácticas curriculares, se analizaron los trabajos y las tareas expuestas anteriormente que se han conseguido realizar:

- Comprensión del objetivo e intereses de la empresa respecto a este proyecto.
- Instalación perfectamente estudiada, elemento por elemento para conocer el proceso de pintura de piezas cilíndricas y así diseñar un nuevo proceso para las piezas de geometrías complejas.
- Conocimiento amplio del lenguaje de programación 'PolyScope' del robot UR10e y del programa de pintura de piezas cilíndricas, realizando gran cantidad de programas para probar movimientos y posibles soluciones.
- Análisis del programa base del PLC, comprendiendo qué hace cada bloque de programa y realizando pruebas modificando el programa para adecuarlo al nuevo proceso de pintura. En este apartado también se incluyen bloques de programa particulares de algunos de los dispositivos conectados al autómeta (reciprocador de Festo).
- Iniciación en la programación de pantallas HMI, para actualizar la interfaz actual del armario de control a la nueva propuesta de trabajo, en el caso de que hubiera sido necesario.
- Se han propuesto múltiples soluciones para pintar estas piezas complejas, incluyendo una prueba puenteando el armario de control para pintar colocando la pieza en el soporte rotativo del interior de la cabina de pintura.

Asimismo, se han seleccionado dos cuyos resultados pueden ser óptimos:

- Realizando rotaciones de la muñeca 3 del robot en dos posiciones fijas oblicuas a la pistola de pintura para que esta se reparta de forma homogénea por la superficie de la pieza.
 - Realizando circunferencias ligeramente inclinadas respecto a un punto medio donde se encuentra fijo el TCP.
-
- Se ha trabajado conjuntamente con el departamento de diseño de la empresa para la creación de los carros y las máscaras de protección de las piezas para transportarlas y evitar que entre pintura por sus orificios, respectivamente.
 - De forma similar, se encargó una pinza especializada a una empresa externa para poder coger las piezas con forma de 'T' y 'X'.
 - Se han realizado auditorías con empresas externas para conocer mejor la estación de pintura y cómo abordar las dificultades y buscar soluciones.
 - Se ha buscado material necesario para el proceso (funda protectora para el robot) y se ha consultado con el gerente de la empresa para su obtención. Una vez recibida, se realizaron pruebas con ella en el robot.

11. Bibliografía

[1] Página principal de la empresa Vulcanizados Alvarez S.L.

<https://www.vulcanizadosalvarez.com/>

Consultado el 25 de mayo de 2021.

[2] Universal Robots e-Series. Manual de usuario del modelo UR10e. Versión 5.6.

<https://www.universal-robots.com/download/manuals-e-series/user/ur10e/56/user-manual-ur10e-e-series-sw-56-spanish-es/>

Consultado el 25 de mayo de 2021.

[3] Página de soporte del fabricante Universal Robots.

<https://www.universal-robots.com/support/>

Consultado el 25 de mayo de 2021.

[4] Página de información del PLC SIMATIC S7-1200, CPU 1214C, DC/DC/DC,

<https://mall.industry.siemens.com/mall/es/WW/Catalog/Product/6ES7214-1AG40-0XB0>

Consultado el 27 de mayo de 2021.

[5] Página de información del SIMATIC S7-1200, E/S digitales SM 1223, 8 DI/8 DO, 8 DI DC 24V.

<https://mall.industry.siemens.com/mall/es/WW/Catalog/Product/6ES7223-1BH32-0XB0>

Consultado el 27 de mayo de 2021.

[6] Página de información de la pantalla SIMATIC HMI, KTP700 Basic

<https://mall.industry.siemens.com/mall/es/WW/Catalog/Product/6AV2123-2GB03-0AX0>

Consultado el 30 de mayo de 2021.

[7] Ficha técnica del robot UR10e

<https://seiki-robotics.com/wp-content/uploads/2019/09/ur10e-tech-spec-es.pdf>

Consultado el 31 de mayo de 2021.

[8] Manual UR SCRIPT (comandos programación PolyScope).

<https://www.universal-robots.com/download/manuals-e-series/script/script-manual-e-series-sw-56/>

Consultado el 30 de mayo de 2021.

[9] “El Controlador Lógico Programable.” Documento del Campus Virtual de la asignatura AUTOMATIZACIÓN INDUSTRIAL.

https://cursoanterior4.campusvirtual.uva.es/2019_2020/pluginfile.php/1091965/mod_resource/content/1/PLC.pdf

Consultado el 2 de junio de 2021.

[10] “Programación en lenguajes de alto nivel con SCL y SIMATIC S7-1200” para TIA Portal.

<https://www.automation.siemens.com/sce-static/learning-training-documents/tia-portal/advanced-programming-s7-1200/sce-051-201-scl-s7-1200-r1709-es.pdf>

Consultado el 6 de junio de 2021.

[11] “Principios básicos de la programación de FC con SIMATIC S7-1200”. Documentación didáctica de TIA Portal.

<https://www.automation.siemens.com/sce-static/learning-training-documents/tia-portal/basics-programming-s7-1200/sce-031-100-fc-programming-s7-1200-r1709-es.pdf>

Consultado el 30 de mayo de 2021.

[12] Página de Universal Robots Academy.

<https://academy.universal-robots.com/es>

Consultado el 25 de mayo de 2021.



[13] Manual de S7-1500 PLCSIM Advanced

https://cache.industry.siemens.com/dl/files/153/109739153/att_895955/v1/s7-plcsim_advanced_function_manual_en-US_en-US.pdf

Consultado el 30 de mayo de 2021.

[14] Portal de soporte de Festo.

https://www.festo.com/net/en-il_il/SupportPortal/

Consultado el 25 de mayo de 2021.